ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 5, May 2013

Constrained Reliability Redundancy Optimization of Flow Networks using Genetic Algorithm

Pardeepkumar

Associate Professor, Dept. of Instrumentation, Kurukshetra University, Kurukshetra, Haryana, India

ABSTRACT:The paper defines a method based on genetic algorithm to solve constrained reliability redundancy optimization of flow networks. The algorithm is implemented using an advance dynamic adaptive penalty function approach considering a mix of feasible and a certain amount of infeasible solutions to guide the search from both feasible and infeasible region sides to optimal or near optimal solution of capacity related reliability of flow networks. The algorithm uses a weighted reliability dependent composite performance measure for the optimization. The method is efficient in terms of its optimality rate and efficiency.

Keywords: flow networks; capacity; telecommunication networks; GA; meta-heuristics.

ACRONYMS

CRR	capacity related reliability	**FN**	flow network
CPM	composite performance measure	GA	genetic algorithm

NOTATION

C_i	capacity of i^{th} subsystem
Cap_i	capacity function of i^{th} subsystem connected either in parallel or series
Cos_j	total amount of resource related to constraint j
$c_{ji}(x_i)$	cost of subsystem i for j^{th} constraint with x_i components, $i = 1, 2, \ldots, n$
$g_i^j(x_i)$	amount of resources consumed by j^{th} constraint in subsystem i with x_i components.
k	number of constraints, $j = 1, 2, \ldots, k$
$L(X)$	$(L_{x1}, L_{x2}, \ldots, L_{xn})$ lower bound of each subsystem i.
m	number of main minimal path sets, $l = 1, 2, \ldots, m$
n	number of subsystems, $i = 1, 2, \ldots, n$
N	number of solution in a set of population
θ	generation number
Off_θ	Offspring of generation θ
$P_l(x_i)$	l^{th} minimal arc/path set of the system, $l = 1,2, .., m$
P_θ	set of population θ
q_i	unreliability of a component at subsystem i
$Q_i(x_i)$	unreliability of subsystem i with $x_i$components.
r_i	reliability of a component at subsystem i.
$R_i(x_i)$	reliability of subsystem i with $x_i$components.
$R_s(X)$	system reliability
$S(x_i)$	set of variables that have been used as key-elements in a given decomposed expressions
$U(X)$	$(U_{x1}, U_{x2}, \ldots, U_{xn})$ upper bound of each subsystem i.
x^*	optimal solution
x_i	number of components in subsystem $i; i= 1,2,.., n$
X	a vector $(x_1, \ldots\ldots x_n)$
X^f	best feasible solution yet obtained
X^u	best infeasible solution yet obtained
Y	finite set of traffic paths
w	Total number of constraint violated
β	severity parameter (user defined)
λ	a positive constant
γ_j^{cos}	Magnitude of violation of constraint j, $\left\lvert \sum g_i^j(x_i) - Cos_j \right\rvert$

ASSUMPTIONS

Following are the assumptions for the rest of the sections:

1. The system and all its subsystems are coherent.
2. Subsystem structures (other than coherence) are not restricted.
3. All component states are mutually and statistically independent.
4. All constraints are separable and additive among components.
5. Each constraint is an increasing function of x_i for each subsystem.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 5, May 2013

I.INTRODUCTION

From the point of view of quality management and decision making, it is important to define the load carrying capacity of each node and arc of the network and develop some indexes in order to evaluate the system performance. Therefore, some researchers [1-4] have proposed models to represent such performance of flow networks and termed them as capacity related reliability (CRR) models. All these models are based on assumption that any node and arc can carry any amount of flow. However, it is not a realistic assumption. Many real life systems such as computer systems, telecommunication systems, transport systems, urban traffic systems, electrical power transmission systems, logistic supply chains, internet and flow networks etc. are mostly linked with performance. The performance of such networks, under the condition that each node and arc flow capacity is deterministic, is not only associated with network reliability but it is also the measure of maximum flow capacity per unit time between source to terminal [5].For instant, computer networks can always be modelled as flow network (FN) in which arcs/branches stands for transmission line and nodes stand for switches. All switches are assumed to be perfect i.e. any amount of flow can pass through them. Virtually each transmission line consists of several physical lines, and each physical line has several successful states hence, the capacity of each branch has several values. However, in modern flow networks or transport systems all the flow paths of a network are never active for transportation of flow from source to destination because the selection of flow paths to transport flow are decided by routing mechanism and logical links assigned in physical layer during the implementation stage. Further arcs and nodes may also be in failure or maintenance states. Hence the system capacity is not a fixed number [5]. Evaluating the probability that the system capacity is larger than or equal to the demand *d* (integer) with cost constraint in general is called the system reliability and is also a useful performance index to measure quality level of FN[6-11].

Many researchers [12-17] have successfully applied Genetic Algorithms (GA) to explore large, multimodal, complex problem spaces. The GA based approach can be effectively adapted for optimization of complex combinatorial problems without the knowledge of explicit mathematical treatment [13,14,16]. Hence, an advanced genetic algorithm based technique using a dynamic adaptive penalty function has been proposed. The dynamic adaptive penalty function approach considers a mix of feasible and a certain amount of infeasible solutions to guide the search from both feasible and infeasible region sides to optimal or near optimal solution of capacity related reliability of flow networks. The proposed GA to optimize CRR problems using composite performance measure (CPM) based on weighted reliability. The concept of weighted reliability was first introduced by Pahuja [18], requires that all the successful states of arcs and nodes qualifying the connectivity measure of the network be enumerated and its probability be evaluated and multiplied by the normalized weight to find out the composite performance of flow networks. The main GA steps are population reproduction, selection, crossover, and mutation. The selection, crossover, and mutation processes are repeated until the termination condition is satisfied. The proposed GA has also successfully addressed the ultrahigh reliability of modern flow networks. As only successful states are considered for computation hence higher temporal efficiency is achieved.

II.COMPOSITE PERFORMANCE MEASURE

A path is a sequence of arcs and nodes connecting a source to a destination. All the arcs and nodes of network have its own attributes like delay, reliability and capacity etc.. From the quality and performance point of view, measurement of the transmission ability of a network to meet the customers demand is very important [8]. When a given amount of flow is required to be transmitted through a flow network, it is desirable tooptimize the network reliability to carry the desired flow. The capacity of each arc (the maximum flow passing the arc per unit time) has two levels, 0 and/or a positive integer value. The system reliability is the probability that the maximum flow through the network between the source and the sink is not less than the demand [6-11]. The proposed algorithm utilizes weighted reliability to form composite performance measure (CPM) to optimize the capacity related reliability (CRR) of flow networks.

The weighted reliability measure [6,7,18] i.e. composite performance measure (CPM), integrating both capacity and reliability may be stated as:

$$\mathrm{CPM}_{P_l} = \frac{\min\limits_{i \in P_l(x_i)} \{Cap_i\}}{Cap_{max}} . R_i \tag{1}$$

where $\min_{i \in P_l(x_i)} \{Cap_i\} / Cap_{max}$ is the normalized weighti.e. the ratio of capacity in the i^{th} state to the maximum capacity of the system and $R_i$is probability of the system being in state $S(x_i)$ and computed as:

$$R_i = P_r\{S(x_i)\} = \prod_{\frac{u}{S(x_i)_u=1}} p_u \text{ x } \prod_{\frac{v}{S(x_i)_v=0}} q_v \tag{2}$$

The capacity function of different subsystems connected in parallel(Ramirez et al. 2005) [4] is:

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 5, May 2013

$$(Cap_i)_{par} = \sum_{i=1}^{x_i} C_i \tag{3}$$

and the capacity function of different subsystems connected in series is:

$$(Cap_i)_{ser} = \min_{i \in P_l(x_i)} \{C_i\} \tag{4}$$

The rules for connecting series and parallel path sets /arcs to integrate capacity and reliability to give composite performance measure are expressed as:

$$\text{CPM}_{par} = \sum_{i=1}^{n} Cap_i \,.\, \bigcup_{i=1}^{n} r_i \tag{5}$$

$$\text{CPM}_{ser} = Cap_i . \prod_{i=1}^{n} r_i \tag{6}$$

III.PROBLEM FORMULATION

The general constrained redundancy optimization problem in complex systems can be reduced to the following integer programmingproblem [19]:

Maximize$R_s(X)$ (7)

$$\text{subject to} \quad \sum_{i=1}^{n} g_i^j(x_i) \le Cos_j, \;\; j = 1, 2, \dots, k \tag{8}$$

and$x_i \;\ge 1,$ for all $i = 1, 2, \dots, n.$

$$L_{x_i} \le x_i \le U_{x_i} \text{for all } i = 1, 2, \dots, n.$$

The above problem model is based on assumption that each node and arc of the network is capable of transporting any amount of flow between source and terminal. However, in present days modern flow networks the situation discussed above isnot justifiable as all the arcs are not simultaneously connected to carry flow from source to terminal. Theperformance of flow networks is considered as the reliability of maximum flow capacity and should consider immediate states of communication that manifest as performance degradation [20] by considering a simple case illustrated in Fig.1

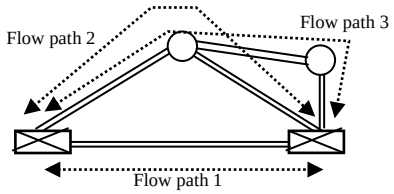

Fig. 1 A simple flow network.

The figure shows that there are three flow paths on the physical layer. Suppose that these flow paths, each having a capacity of 200, are used for ensuring transmission between source and terminal by load sharing. Then, the total flow capacity of the network is computed by summing the capacities of active flow paths. In this case, there are four failure modes. That is, the total capacity between two routers could be 0, 200, 400, or 600, which are recognized as being equivalent to the maximum flow in the graph. However, suppose flow paths 1 & 2 are primary flow paths with each having a capacity of 200, while flow path 3 with capacity 200 is a backup path only for flow path 1. In this case, failure modes are capacities 0, 200, or 400. These are quite different from maximum flow in graph theory.The selection of paths to transport flow is decided by routing mechanism and logical links assigned in physical layer. Thus in practical systems the entire pathsets are never utilized for transfer of information [20]. The flow is transmitted through the main path(s) only and in case of failure of main path(s), backup path(s)takes over the task of main path(s).Hence, considering the situation as discussed the general constraint reliability redundancy optimization problem expressed by equations (7) and (8) has been adapted by replacing $R_s(X)$with composite performance measure (CPM) discussed in section II (1) equations (5) and (6)above.

IV.GENETIC ALGORITHM

The development of GA can be termed as an adaptation of a probabilistic approach based on principles of natural evolution. Nature observes the principle of survival of fittest, weak and unfit species within their environment are faced with extinction by natural selection. The strong ones have greater opportunity to pass their genes to future generations

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 5, May 2013

via reproduction [21]. The genetic algorithms follow the process in which populations undergo continuous change through cross breeding, mutation and natural selection.

A. Steps of Genetic Algorithm:

Step 1: Set $\theta = 1$. Randomly generate N solutions to form the first population, $P1$. Evaluate the fitness of solutions in $P1$.

Step 2: Crossover: Generate an offspring population Off_θ as follows:

2.1. Randomly choose two solutions $X_a \in P_\theta$ and $X_b \in P_\theta$ based on the fitness values.

2.2. Using a crossover operator, generate offspring and add them to Off_θ.

Step 3: Mutation: Mutate each solution $X \in \text{Off}_\theta$ with a predefined mutation rate.

Step 4: Fitness assignment: Evaluate and assign a fitness value to each solution $X \in \text{Off}_\theta$ based on its objective function value and infeasibility.

Step 5: Selection: Select N solutions from Off_θ based on their fitness and copy them to $P_\theta + 1$.

Step 6: If the stopping criterion is satisfied, terminate the search and return to the current population, else, set $\theta = \theta$ +1 go to Step 2.

The values of its parameters, such as, population size, crossover rate, mutation rate etc. can be appropriately chosen to balance both the quality of the solution and the computational work.

An initial population of a fix number of solutions is randomly selected to form the first generation of the GA. Crossover and mutation operators are then performed on the population members to produce subsequent generations. An effective GA depends on complementary crossover and mutation operators. The crossover operator dictates the rate of convergence, while the mutation operator prevents the algorithm from prematurely converging. Each member of the population is evaluated in accordance with its fitness, which is used as basis for selecting parent solutions and for culling inferior solutions from the population [14]. However, there are some difficulties in determining appropriate values for the parameters and a penalty for infeasibility. If these values are not selected properly, a GA can rapidly converge to a local optimum or slowly converge to the global optimum [13]. The population size and number of generations enhance the solution quality while increasing the computation. GA's generally yield several good solutions (optimal or near-optimal) and thus multiple solutions obtained provide much flexibility in decision-making for reliability design. Though, these methods yield better optimal solutions at the cost of temporal efficiency. Genetic Algorithms are most suited to unconstrained optimization problems. Apart from this, sometimes for some problem GA may converge prematurely while finding optimal solution because the prior knowledge is not exploited and the local search information is not explored. Over and above this, it takes time to tune appropriately the unknown GA parameters like population size, crossover probability, maximum generation and mutation probability, to make a balance between exploitation and exploration in the search space [22]. Hence, adapting Genetic Algorithms to constrained reliability redundancy optimization problems is a challenging task. In literature, most common method in genetic algorithms to handle constraints is to use penalty functions such as Adaptive Penalties, Death Penalty, Static Penalties, Dynamic Penalties, Annealing Penalties, Segregated GA, Co-evolutionary Penalties etc. [22, 23].A new genetic search algorithm using penalty function to penalize infeasible solutions by reducing their fitness values in proportion to their degrees of constraint violation metric and its performance is presented in the following sections.

V. PENALTY FUNCTION

Meta-heuristic algorithm developed is based on the approach of Bazarra et al. [24] and Yeniay [25]. The penalty function approach is used to convert the nonlinear programming problem with equality and inequality constraints into an unconstrained problem, or into a problem with simple constraints [26]. This is achieved by including the constraints into the objective function via a penalty parameter to penalize any violation of the boundaries/constraints. It is generally difficult to find an effective and efficient penalty function, which can inherit the constraints. Defining penalty function for solving a problem varies with the type of the problem.

The combination of distance metric together with the length of the search has been very effective for this type of search problems. The algorithm developed is based on an adaptive penalty function and the search pattern Agarwal and Gupta [16, 26]. During iterations the penalty function adapts itself according to the deviations of current solution from the best feasible/infeasible solution obtained by the algorithm. This helps the search not to go too far into the infeasible region. It also incorporates the ratio of the count of constraints violated to the total number of constraints as one of the distance metric. The algorithm also incorporates the number of iterations elapsed as length of search in penalty function; hence justifying the name dynamic adaptive penalty function. Although GA algorithms have proven to be efficient for solving reliability redundancy optimization of system, one of their major disadvantages is the increase of computational effort involved.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 5, May 2013

The general penalty function approach is as follows.

$$\text{Max / Min} \qquad R_s(X) \text{or } \text{CPM}_{P_l} \tag{9}$$
$$\text{Subject to} \qquad x \in A \text{ and } x \in B$$

where A and B are limits of search space. The problem can be reformulated as state:

$$\text{Max / Min} \qquad R_s(X) \text{or } \text{CPM}_{P_l} + p(d(X,B)) \tag{10}$$
$$\text{Subject to} \qquad x \in A$$

where $d(X,B)$ is a function measuring the distance of the vector X from the region B, p is a monotonically non-decreasing penalty function such that $p(0) = 0$. Measure of various distance metrics (d) include a count of violated constraints, the Euclidean distance between X &B, a linear sum of the individual constraint violations, or this sum raised to some exponent β a user defined severity parameter [16, 26, 27].

On the basis of above discussions, the dynamic adaptive penalty function is designed to solve the constrained reliability redundancy optimization problems of complex systems. It is an adaptation of penalty function used by [16] for series parallel systems to complex systems. It accounts for the amount of infeasibility and the maximum improvement possible in the current solution of a generation, and exhibiting severity of the penalty imposed.

$$(\text{CPM}_{P_l}(X))_P = \text{CPM}_{P_l}(X) - \left({}^{W}/_{k}\right)\left(\sum_{j=1}^{k} \frac{\gamma_j^{cos}}{Cos_j}(1-\lambda\theta)\right)^{\beta}. \tag{11}$$
$$\exp\left(\frac{1-\text{CPM}_{P_l}(X)}{\text{CMP}_{P_l}(X^u) - \text{CMP}_{P_l}(X^f)}\right)$$

The adapted/penalized composite performance measure $(\text{CMP}_{P_l}(X))_P$ is a function of composite performance measure $\text{CPM}_{P_l}(X)$ and the severity of the constraints. The metric $\left({}^{W}/_{k}\right)$ calculates the ratio of the number of constraints violated to the total number of constraints. The distance metric $\left(\sum_{j=1}^{k} \frac{\gamma_j^{cos}}{Cos_j}(1-\lambda\theta)\right)^{\beta}$ evaluates the sum of the ratios of magnitudes of constraint violations to total resources available incorporating the dynamic aspect, and increases the severity of the penalty for a given distance as the search progresses, with β as (user-defined) severity parameter, λ a positive constant, (taken as 0.005 after experimentation), and θ the generation number. The adaptive term, $\exp\left(\frac{1-\text{CPM}_{P_l}(X)}{\text{CMP}_{P_l}(X^u)-\text{CMP}_{P_l}(X^f)}\right)$, takes care of the maximum possible improvement in the current solution with respect to the difference between the un-penalized values of the best infeasible and feasible solutions obtained up to the time, $\text{CMP}_{P_l}(X^u)$ and $\text{CMP}_{P_l}(X^f)$, respectively.

When $\text{CMP}_{P_l}(X^u) \leq \text{CMP}_{P_l}(X^f)$, then penalty is not imposed on the infeasible solutions. Moreover, the impact of the adaptive penalty is such that the infeasible solutions having less reliability are penalized more and so the search is restricted to the promising infeasible solutions.

VI. BOUNDS FOR SYSTEM

The bounds to each of the subsystem determine the region of search. The lower bound to the system is generally known from the system design. For systems with heterogeneous redundancy allocation lower bound vector is taken as: L = (1, 1, …, 1) for all x_i . While the upper bound is determined from the constraints, allocating the whole resource of constraint j to x_i, and determine the maximum value of x_i while keeping all other subsystems at their lower bounds.

VII. STOPPING FUNCTION

In this algorithm two stopping criteria, i.e., number of generations and/or the desired system reliability are considered. The number of generation is considered as one of the stopping criteria for the algorithm and this is user or problem dependent. The second criterion for terminating the algorithm is the desired value of the system reliability.

VIII. COMPUTATION AND RESULTS

To illustrate the performance of the proposed algorithm a network having six arcs {x_1, x_2, x_3, x_4, x_5, x_6} and five minimal path sets {y_1, y_2, y_3, y_4, y_5} as shown in the Figure.2 is considered and solved for capacity related constraint redundancy reliability optimization using CPM. The network shown in Fig. 2 is a bench mark problem, considered by Hayashi & Abe (2008) [20].

Using Baye's method, the Reliability of the system can be expressed as:

$$R_s(X) = R_3[1- Q_6\{1-(1-Q_1Q_2)(1-Q_4Q_5)\}] \tag{12}$$
$$+ Q_3[1-(1-R_2R_5)(1-R_1R_4)]*Q6$$

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 5, May 2013

TABLE 1 DATA FOR FIG. 1

i	***1***	***2***	***3***	***4***	***5***	***6***
r_i	0.70	0.75	0.8	0.85	0.70	0.90
c_{1i}	2	3	2	3	1	3
Cos_1	30					

The problem is solved for data given in Table 1 and by assuming that each flow path has a capacity of 100. The total flow through network at any time should not be less than 200 in any case.

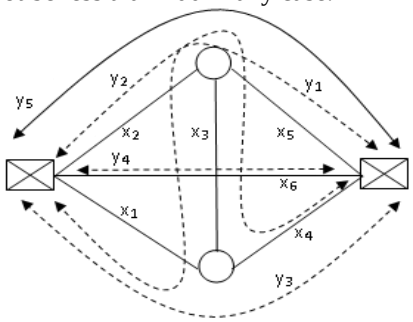

Fig. 2 Illustration Network

The procedure to implement the search using genetic algorithm is as follow:

A. Chromosome representation:

Chromosome representation is a crucial part of any genetic algorithm. There are several ways of chromosome representation for any given problem. The GA works with the binary encodings, but for many industrial problems binary string is not a natural coding. Hence various non-string techniques have been used for such problems: real number coding for constrained optimization problems and integer coding for combinatorial problems. For the nonlinear integer programming problems of the type given by (1), the integer encoding as a vector representation is more efficient.

B. Generation of initial population:

A fixed number of chromosomes are randomly generated to form an initial population. In a chromosome, a gene at any locus is randomly generated integer between the lower limit and upper limit defined for each subsystem. Population size of 100 is taken for each of the test problems.

C. Evaluating the fitness of chromosomes:

The fitness of the chromosomes is found by using penalized objective function value using (5).

D. Genetic Operators:

Two genetic operators Crossover and Mutation are the backbone of any GA. Using these two operators new off springs of the genes are created at each generation.

1) Crossover Operator:

It provides a thorough search of regions of the sample space to produce good solutions. For this GA, crossover rate p_c, denoting the expected number of chromosomes that undergo crossover, is taken as 0.5 for the system structures. However, depending on the size of the system, it can be varied. The total population is arranged in decreasing order of fitness i.e. fittest to the least fit chromosome. A random number p is generated such that $0 \leq p \leq 1$ for each of the population member and a chromosome is selected as a parent if and only if $p < p_c$. If the number of such selected chromosomes is odd then the last chromosome selected as parent is dropped from the mating pool. Single-point crossover method is used to mate the parents and cut position is generated randomly from the interval $[1, n]$.

2) Mutation:

Mutation operator produces random changes in various chromosomes. The mutation rate is the rate p_m controls the rate at which new genes are introduced into the population for trial. If it is too low, then some genes that would have been useful are never tried out, but if it is too high, then there will be much random perturbation and the off springs will start losing their resemblance to the parents and the algorithm will lose the ability to learn from history. For the present GA, p_m is taken as 0.20. Random number p is generated for each gene within the interval from interval [0, 1] as $0 \leq p \leq 1$. If $p < p_m$ then the gene is randomly flipped to another gene from the corresponding set of alternatives (To reach upon the effective values of the parameters p_c and p_m brief experiments were conducted).

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering
Vol. 2, Issue 5, May 2013

E. Selection:

Each generation after undergoing genetic operations produces good or bad solutions. All the solutions (initial population, children and off springs) are arranged in descending order of fitness and the population is selected for the next generation.

The algorithm has been programmed using MATLAB and tested on a Core i7 VPro 3.4 GHz system. Execution time for the problem is determined. The proposed genetic algorithm gives the optimal solution (2, 2, 2, 2, 1, 3) with system reliability R_s = 0.9578, the optimized subsystem reliability probability R_i and unreliability probabilities Q_i are shown in Table 2.

TABLE 2 OPTIMIZED SUBSYSTEM RELIABILITY/UNRELIABILITY FOR FIG. 1

i	x_1	x_2	x_3	x_4	x_5	x_6
X^*	2	2	2	2	1	3
R_i	0.9100	0.9375	0.9600	0.9775	0.7000	0.9990
Q_i	0.090	0.0625	0.0400	0.0225	0.3000	0.0009
Cap_i	200	200	200	200	100	300

TABLE 3 OPTIMIZED ARC CAPACITY AND CPM FOR FIG. 1

arc $y_i \in Y$	arc Capacity $Cap_{y_i} = \min_{i \in P_l(x_i)} \{C_i\}$	$\mathrm{CPM}_{y_i} = \frac{Cap_{y_i}}{Cap_{max}} . \prod_{i=1}^{n} R_i$
$y_1 = \{1\ 3\ 5\}$	100	0.3058
$y_2 = \{2\ 3\ 4\}$	200	0.8797
$y_3 = \{1\ 4\}$	200	0.8895
$y_4 = \{6\}$	100	0.9990*
$y_5 = \{2\ 5\}$	200	0.3281

* as $0 \leq Cap_{y_i}/Cap_{max} \leq 1$

The capacity of each subsystem of the flow path is taken as 100 and the capacity of flow paths of the network is determined using (3) of proposed approach. The optimized arc capacity Cap_{y_i} and value of composite performance measure CPM_{y_i} are illustrated in Table 3.Thevalue for CPM for an assumed flow of 200 is supposed to pass through the flow path and it comes out to be 1.0000 as shown below:

$$\mathrm{CPM}_{Network} = 1 - \left(1 - \mathrm{CPM}_{y_1}\right).\left(1 - \mathrm{CPM}_{y_2}\right). \left(1-\mathrm{CPM}_{y_3}\right).\left(1 - \mathrm{CPM}_{y_4}\right). \left(1-\mathrm{CPM}_{y_5}\right) \quad (12)$$

$$= 1 - (1 - 0.3058)(1 - 8797)(1 - 0.8895)(1 - 0{,}9990)(1 - 0.3281) \cong 1.0000$$

To demonstrate the adaptability of the penalty function to constrained redundancy optimization of flow networks problem with respect to β, the different values β ($1.0 \leq \beta \leq 8.0$) are used to determine the best feasible solution, average reliability, standard deviation and average time consumed in 10 GA trials. The best value of β for a particular problem is the value of β giving the best feasible solution, highest average reliability, least standard deviation and least average time. These values are shown in the Table 4.

TABLE 4 SENSITIVITY OF $\boldsymbol{\beta}$ FOR FIG. 1 NETWORK

β	Best feasible solution	Average reliability	Standard deviation	Average time
1.0	0.9578	0.9578	0	0.974204
2.0	0.9578	0.9578	0	0.99168
3.0	0.9578	0.9578	0	0.985204
4.0	0.9576	0.9578	0.0001414	1.116115
5.0	0.9577	0.9469	0.0076367	1.013334
6.0	0.9578	0.9587	0.0006364	1.023888
7.0	0.9578	0.9587	0.0006364	1.01852
8.0	0.9578	0.9587	0.0006364	0.973689

The values of β (1.0, 2.0 or 3.0) give the same best feasible solution (2 2 2 2 1 3) with system reliability R_s= 0.9578 and zero standard deviation parameter. However, the same solution and system reliability values in all the 80 GA trials with least average time is obtained for β = 8, but the standard deviation in this case is more in comparison to the cases for β

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

= 1.0, 2.0 or 3.0. The average time consumed is comparable and average reliabilities are same for the selected values of β. It can be noted that higher values of β increases the standard deviation.The above result shows that proposed method is capable of optimizing the flow network to transport the desired capacity through the network with best feasible solution, highest average reliability, best average time and least standard deviation.

IX. CONCLUSIONS

The GA method developed in this study give very promising results. The algorithm attack the search for optimal solution from both infeasible and feasible region sides which is a superior strategy to keep the solution in feasible region for constrained redundancy reliability optimization of flow networks. For the search to precede efficiently toward the final optimal /near optimal solution a dynamic adaptive penalty function using distance based metric, has been utilized. The demonstrations show that GA approach can be a powerful tool for solving problems of constrained redundancy optimization of flow networks. The numerical example demonstrates that the proposed algorithm is fast for designing large, reliable telecommunications networks.

REFERENCES

[1] H. Nagamochi and T. Ibaraki, "On Onaga's upper bounds on the mean values of probabilistic maximum flows," IEEE Trans. Reliab.,Vol. R-41, pp. 225–229, 1992.
[2] P. K. Varshney, A. R. Joshi, and P. L. Chang, "Reliability modelling and performance evaluation of variable link-capacity networks," IEEE Trans. Reliab.,Vol. R-43, pp. 378–382, 1994.
[3] Y. Chan, E. Yim, and A. Marsh, "Exact & approximate improvement to the throughput of a stochastic network," IEEE Trans. Reliab.,Vol. R-46, pp. 473–486, 1997.
[4] S. Soh and S. Rai, "An efficient cutset approach for evaluating communication- network reliability with heterogeneous link-capacities," IEEE Trans. Reliab.,Vol. R-54, pp. 133–144, 2005.
[5] Y. K Lin, "Reliability of Flow Network subject to budget constraints", IEEE Trans. Reliab.,Vol. (56), pp. 10-16, 2007.
[6] P. Kumar, D. K. ChaturvediandG. L. Pahuja, "A Heuristic Method for Reliability Redundancy Optimization of Flow Networks", Reliability Theory and Applications,Vol.1, #02(25), pp.69-77, 2012.
[7] P. Kumar, "Cardinality Based Approach for Reliability Redundancy Optimization of Flow Networks", Reliability Theory and Applications,Vol. 7,#04(27), pp. 63-71, Dec. 2012.
[8] Y. K.Lin, "Reliability of a computer network in case capacity weight varying with arcs, nodes and types of commodity", Reliability Engineering and System Safety,Vol. 2(5), pp. 1-7, 2006.
[9] G. L. Pahuja, "*Reliability Evaluation and Optimization Recent and New Approaches",* Ph.D. thesis, Kurukshetra University, Kurukshetra, 2004.
[10] Lin Y. K., "Reliability evaluation for an information network with node failure under cost constraint," IEEE Trans. Systems, Man and Cybernetics-Part A: Systems and Humans, Vol. 37, no. 2, pp. 180–188, 2007.
[11] Y. K. Lin, "On a multi-commodity stochastic-flow network with unreliable nodes subject to budget constraint," European Journal of Operational Research,Vol. 176, no. 1, pp. 347–360, 2007.
[12] J. H. Holland, "Genetic algorithm and the optimal allocation of trials", SIAM Journal on Computing,Vol.2(2), pp.88-105,1973.
[13] F. Tillman, C. Hwang, and W. Kuo, *Optimization of systems reliability,* New York: Marcel Dekker; 1980.
[14] D. Coit, and A. Smith, "Reliability optimization of series-parallel systems using a genetic algorithm", IEEE Trans. Reliab.,Vol. 45(2), pp. 254–60, 1996.
[15] D.W. Coit and A. Smith, "Penalty guided genetic search for reliability design optimization", Computers and Industrial Engineering,Vol. 30, no. 4, pp. 895–904, September 1996.
[16] R. Gupta and M. Agarwal, "Penalty guided genetic search for redundancy optimization in multi-state series-parallel power system", J. Comb. Optim.,Vol. 12, pp. 257–277, 2006.
[17] A. Konak, D.W. Coit and A.E. Smith, "Multi-objective optimization using genetic algorithms: A tutorial", Reliability Engineering and System Safety,Vol. 91, pp. 992–1007, 2006.
[18] G. L. Pahuja, "*Reliability Evaluation and Optimization Recent and New Approaches",* Ph.D. thesis, Kurukshetra University, Kurukshetra, 2004.
[19] J. A. Abraham, "An improved algorithm for network reliability," IEEE Trans. Reliab.,Vol. R-28, pp. 58–61, 1979.
[20] M. Hayashi andT. Abe, "Evaluating Reliability of Telecommunications Networks Using Traffic Path Information", IEEE Trans. Reliab.,Vol. 57(2), pp. 283-294, 2008.
[21] P. Kumar, D. K. ChaturvediandG. L. Pahuja,"Constrained Reliability Redundancy Optimization of Complex Systems using Genetic Algorithm", MIT Int. J of Electrical and Instrumentation Engineering, Vol. 1, No. 1,pp. 41-48, 2011.
[22] G. Mitsuo, and S.Y. Young, "Soft computing approach for reliability optimization: State-of-the-art survey", Reliability Engineering and System Safety,Vol. 91, pp. 1008–1026, 2006.
[23] W. Kuo V.R. Prasad, F. Tillman, and C.L. Hwang,Optimization reliability design: fundamentals and applications,Cambridge: Cambridge University Press; 2000.
[24] M.S. Bazaraa, H. D. Sherali and C. M. Shetty,Non-linear programming: theory and algorithms, John Wily and Sons, 2nd edition, 1993.
[25] Y. OzgurYeniay, "Penalty function methods for constrained Optimization with genetic algorithms", Mathematical and Computational Applications,Vol. 10(1), pp. 45-56, 2005.
[26] M. Agarwal and R. Gupta, "Penalty Function Approach in Heuristic Algorithms for Constrained Redundancy Reliability Optimization", IEEE Trans. Reliab.,Vol. 54 (3), 549-558, 2005.
[27] T. Kohda and K. Inoue, "A reliability optimization method for complex systems with the criteria of local optimality", IEEE Trans. Reliab.,Vol. 31, pp. 109–111, 1982.