Journal of Al-Qadisiyah for Computer Science and Mathematics Vol.16.(2) 2024,pp.Math 62–69

Available online at www.qu.edu.iq/journalcm
JOURNAL OF AL-QADISIYAH FOR COMPUTER SCIENCE AND MATHEMATICS
ISSN:2521-3504(online) ISSN:2074-0204(print)

An Allocation and Optimization Procedure for ARPANET Network Reliability Using Particle Swarm Optimization

Laith Adnan Hadi Almhana*

Islamic Azad University Kermanshah Branch, Iran.Email: almhanalaith08@gmail.com

ARTICLE INFO

Article history:
Received: 17 /04/2024
Revised form: 01 /05/2024
Accepted : 02 /05/2024
Available online: 30 /06/2024

Keywords: Enhanced particle swarm optimization (PSO)

Reliability optimization,

Reliability allocation

ABSTRACT

In this paper we study the expansion of the telecommunications industry and the complexity of communications infrastructures. For ARPA networks such as ARPANET, the provision of integrated broadband service is essential. The new technology used in ARPANETs and changing traffic patterns have attracted much research attention to network topological design. Most topological design researchers have proposed techniques based on expensive exchange-based hardware. We presented a realistic ARPA network model in this study. ARPA network architecture includes network optimization. We proposed to apply particle swarm optimization (PSO) to enhance the ARPA network, and we also estimated the appropriate distribution of each system component using an exponential cost function. The results of the study show that the optimizer (PSO) had good results.

MSC..

https://doi.org/10.29304/jqcsm.2024.16.21554

1. Introduction

In this work, we examined the optimal assignment to guarantee the dependability of the Advanced Research Projects Agency (ARPA) network [1, 2]. The system's dependability was evaluated by the computation of short paths connecting connection matrices. “Boolean algebra is used to acquire all pathways, and nodes are subsequently removed to provide minimal paths [3, 4]”. The dependability function is examined to find out more about the implemented advanced system's level of safety. In this research, we also study the mathematical challenge of propagating optimal dependability despite the history of networks. The optimization and relevance of the location dictate the dependability needs for each part of a complicated system. Reducing total expenses and improving system lifetime and reliability are the objectives [1, 5]. To increase overall dependability, some components may require a significant amount of allocation, depending on where they are positioned within the system. “Engineers encounter a number of challenges when trying to improve mechanical and electrical systems [6, 7]”. It is the goal of this work to improve the distribution and reliability of complex systems in addition to system cost, which can be expressed in terms of volume, weight, or other metrics. “One of the

Laith Adnan Hadi Almhana

∗Corresponding author

Email addresses: almhanalaith08@gmail.com

Communicated by ‘sub etitor’

two primary elements that determines the dependability of this component is the need for the model to be cost-based in order to validate the correctness of the input component". The parameters of the suggested cost factor are modifiable [8]. This may be used by engineers to assess how the system has changed and to determine how to meet the basic reliability requirements for every machine component. Second, the model has to take the soundness of the input system's analysis into account. Certain situations provide a major problem that, in bigger systems, may become a major undertaking. These situations include basic systems. The findings were obtained using the Algorithm Particle Swarm Optimization (PSO), which helps to resolve optimization problems in complex systems. Utilizing an exponential behaviour model, the cost was calculated as shown in **figure 1** [9].

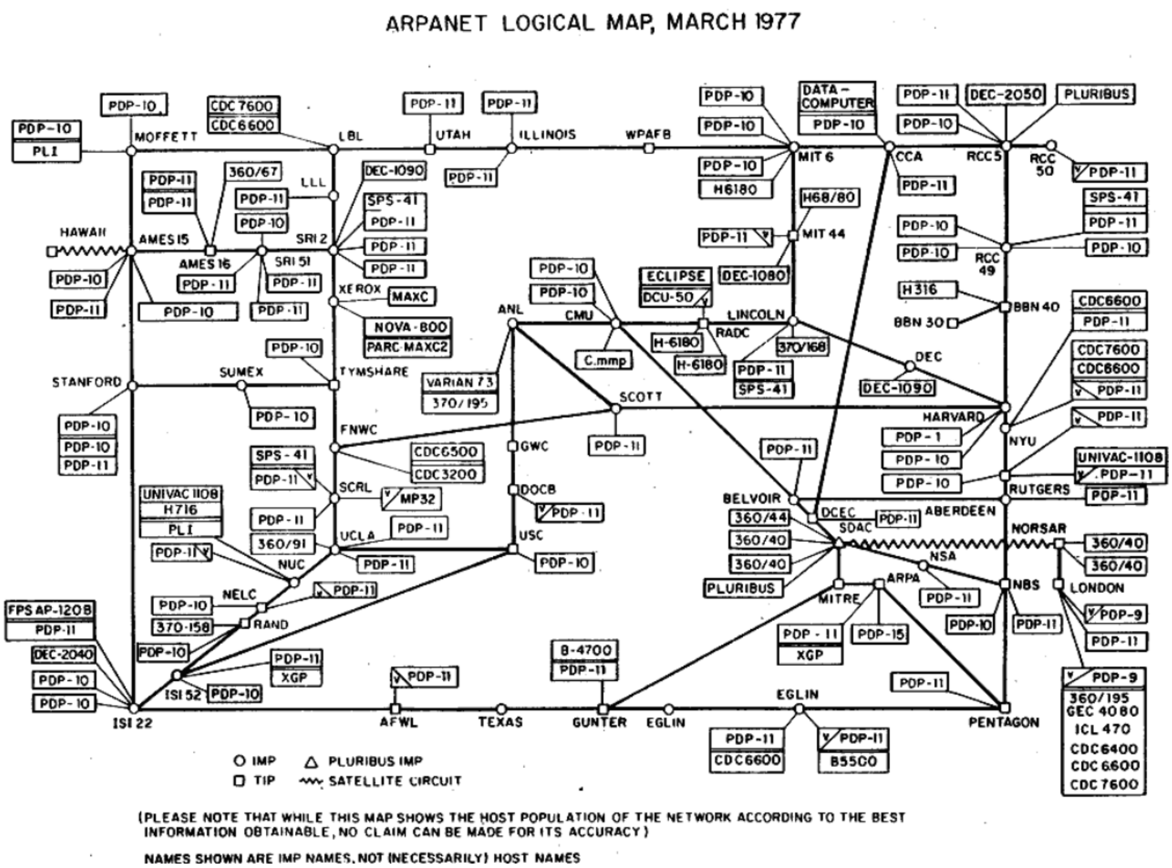

Figure 1: ARPA network, schematic map.

1.1 (ARPA) reliability allocation and network optimization

Consider a (ARPA) network where components relating to dependability are included [10, 11] . The following notes are utilized by us:

$C_i(R_i)$ = cost of element;

$0 \leq R_i \leq 1$ = reliability as an element;

R_s = dependability of the system;

$C(R_1, \ldots, R_n) = \sum_{i=1}^{n} a_i c_i(R_i)$ is the overall cost of the system, where $a_i > 0$;

RG stands for "systems reliability objective [5,19].

"There are several possible outcomes due to the system's modular design and the unique responsibilities that each component performs". "The sum of the numerous different system components, each with a unique level of reliability, is our capacity [10, 12]". "The system's ability to distribute resources appropriately to all or some components is the ultimate goal. Problems are necessary for nonlinear programming [12, 13]". The restriction is relevant even if it is nonlinear, and it may be studied in connection to costs:

$$Minimized\ C(R_i, \dots, R_i) = \sum_{i=1}^{n} a_i C_i(R_i),\ a_i > 0,$$

Subject to:

$$R_s \geq R_G$$

$$0 \leq R_i < 1, in\ which\ i = 1, \dots, n \qquad [1]$$

Assume that $C_i(R_i)$ meets certain requirements and that the partial cost function is fair. Positive, differentiated functions that increase from $\left[\Rightarrow \frac{dC_i}{dR_i} \geq 0\right]$. are referred to us [14]. The Euclidean convexity $C_i(R_i)$ part costs function is equivalent to the fact that its derivatives, $\frac{dC_i}{dR_i}$, are monotonically raised, i.e., $\frac{d^2C_i}{dR_i^2} \geq 0$. [3]

By achieving an all-out framework cost foundation, subject to R_G, the prior method lessens the limitation on system dependability [15].

1.2 Particle Swarm Optimization (PSO)

PSO is an unconventional optimization method that may be used for ARPA Network optimization [3]. Encoding, initiation, evaluation, cloning, crossover, mutation, and termination are PSO processes that may be briefly explained [16, 17]. Coding: The variables in the optimization problem are modeled as genes in this stage. The main chromosomes are chosen at random from a group of chromosomes with different gene makeup. These arbitrary chromosomes, whose size is defined by the arbitrary number of chromosomes, comprise the population. Evaluation: Based on the gene order, each chromosome in the community is assigned a fitness rating. A chromosome's fitness value in a population establishes how likely it is to survive changes in gene order. Chromosome replications are assessed using fitness values derived from various genetic combinations [7]. The goal of reproduction is to reduce the quantity of harmful chromosomes in the offspring while increasing the amount of good ones. Crossover: This procedure involves the switching of the mother's and father's chromosomes. "Two chromosomes are selected at random from the population to serve as parent chromosomes". At the crossing sites—which are selected such that the number of crossing points is less than the total number of genes on the chromosome—the genes are exchanged. Two new chromosomes are made using genes from each of the parent chromosomes. The primary focus of this exercise is the intersection of two points. Mutation: To produce a new chromosome that differs from the chromosomes that are currently in the population, a mutation process is used. A random selection is made for the modified chromosome. "Among all the genes in the modified chromosome, a mutant gene is chosen at random, and its value is then modified". Until the average level of fitness in the population is almost constant, the procedure is continued. Decoding the best chromosome in the neighborhood is the last step in solving the optimization issue. In a previous research, PSO was used differently and partially to build the ARPA network in order to boost bandwidth [18]. It was decided not to use meta-inference for the entire ARPA network planning problem. The basis of particle swarm optimization is gene evolution. PSO ignores the knowledge-producing process of cultural progress. The quick convergence of local optima is one of the drawbacks of PSO-based systems. Enhanced PSO offers a workaround for this restriction. Local search strategies are used in enhanced PSO stages to achieve better results. In this study, the local search strategy that was considered is the hill climb algorithm. The idea behind hill climbing is to always aim to improve upon your current level of fitness. The algorithm searches for these states as they become available; if any of these states do, the process ends.

1.3 Application in a complicated system

We want to structure the ARPA network into a network with parallel or series connections between its components, making it easier to estimate. With n components, the dependability of a series and parallel network is, respectively:

$$R_s = \prod_{i=1}^{n} R_i \qquad [8]$$

$$R_s = 1 - \prod_{i=1}^{n}(1 - R_i) \qquad [3]$$

In this case, $R_i$denotes the component i's dependability, while R_N denotes the ARPA network's reliability [16].

Equations (1) and (2) will be used to compare the dependability of each complex network with the provided p minimum paths.

$$R_s = 1 - \prod_{z=1}^{p}\left(1 - \prod_{j=\alpha}^{\omega} R_j\right)$$

In this instance, the index of the first component of a minimum pa z is represented by α, while the index of the last component is represented by ω.

Equation (3) may be applied to ascertain the reliability of the intricate network seen in **Figure** 2.

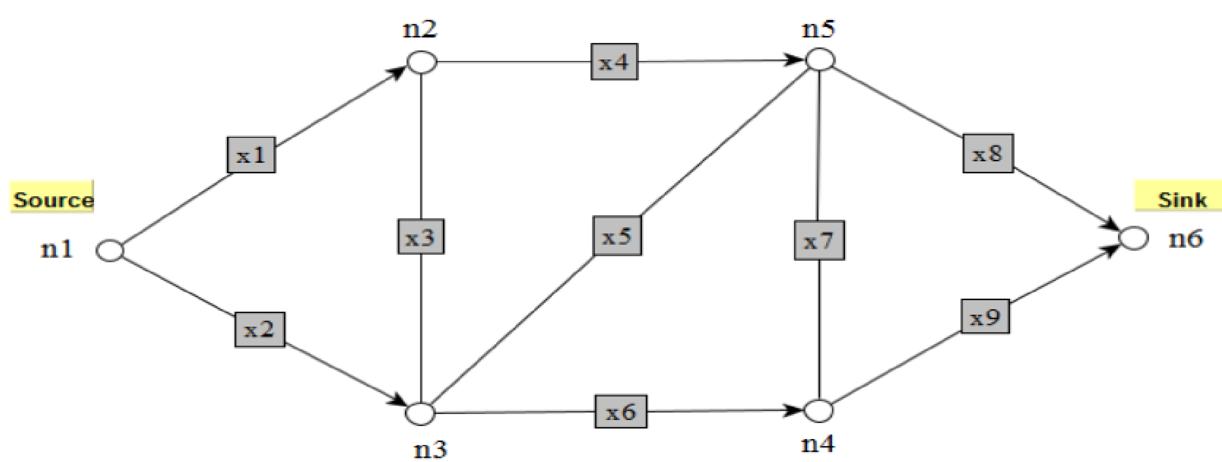

Figure 2. adapted ARPA network

$$\begin{aligned} S = \ & \{Mps_1 = \{x_1x_4x_8\}, Mps_2 = \{x_2x_5x_8\}, Mps_3 = \{x_2x_6x_9\}, Mps_4 = \{x_1x_3x_5x_8\}, \\ & Mps_5 = \{x_1x_3x_6x_9\}, Mps_6 = \{x_1x_4x_7x_9\}, Mps_7 = \{x_2x_3x_4x_8\}, Mps_8 = \{x_2x_5x_7x_9\}, \\ & Mps_9 = \{x_2x_6x_7x_8\}, Mps_{10} = \{x_1x_3x_5x_7x_9\}, Mps_{11} = \{x_1x_3x_6x_7x_8\}, \\ & Mps_{12} = \{x_1x_4x_5x_6x_9\}, Mps_{13} = \{x_2x_3x_4x_5x_8\}\} \end{aligned}$$

$$R_s = 1 - [1 - p_r\{x_1x_4x_8\}] \times [1 - p_r\{x_2x_5x_8\}] \times [1 - p_r\{x_2x_6x_9\}]$$

$$\times [1 - p_r\{x_1x_3x_5x_8\}] \times [1 - p_r\{x_1x_3x_6x_9\}] \times [1 - p_r\{x_1x_4x_7x_9\}]$$

$$\times [1 - p_r\{x_2x_3x_4x_8\}] \times [1 - p_r\{x_2x_5x_7x_9\}] \times [1 - p_r(x_1x_2x_6x_7)]$$
$$\times [1 - p_r\{x_2x_6x_7x_8\}] \times [1 - p_r\{x_1x_3x_6x_7x_8\}] \times [1 - p_r\{x_1x_4x_5x_6x_9\}]$$
$$\times [1 - p_r\{x_2x_3x_4x_5x_8\}] \qquad [16]$$

Note: “When the $i-$ th component is successful, then $R_i = 1$, and when it fails, then $R_i = 0 \; \forall \; i = 1, \cdots, 9$, these lead to

$R_i^n = R_i$ [12]”.

Using the above comment, equation (5) becomes the following polynomial.

$$\begin{aligned} R_S = {} & R_1R_4R_8 + R_2R_5R_8 + R_2R_6R_9 + R_1R_3R_5R_8 + R_2R_3R_4R_8 + R_1R_3R_6R_9 + R_1R_4R_7R_9 \\ & + R_2R_5R_7R_9 + R_2R_6R_7R_8 - R_1R_2R_3R_4R_8 - R_1R_2R_3R_5R_8 - R_1R_2R_4R_5R_8 \\ & - R_1R_2R_3R_6R_9 - R_1R_3R_4R_5R_8 - R_2R_3R_4R_5R_8 + R_1R_3R_5R_7R_9 + R_1R_3R_6R_7R_8 \\ & + R_1R_4R_5R_6R_9 + R_2R_3R_4R_7R_9 - R_2R_5R_6R_7R_8 - R_1R_4R_7R_8R_9 - R_2R_5R_6R_7R_9 \\ & - R_2R_5R_6R_8R_9 - R_2R_5R_7R_8R_9 - R_2R_6R_7R_8R_9 + 2R_1R_2R_3R_4R_5R_8 - R_1R_2R_3R_4R_7R_9 \\ & - R_1R_2R_3R_5R_7R_9 - R_1R_2R_3R_6R_7R_8 - R_1R_2R_4R_5R_6R_9 - R_1R_2R_4R_5R_7R_9 \\ & - R_1R_2R_4R_6R_7R_8 - R_1R_3R_4R_5R_6R_9 - R_1R_2R_4R_6R_7R_9 - R_1R_3R_4R_5R_7R_9 \\ & - R_1R_3R_4R_6R_7R_8 - R_1R_2R_4R_6R_8R_9 - R_1R_3R_4R_6R_7R_9 - R_1R_3R_5R_6R_7R_8 \\ & - R_2R_3R_4R_5R_7R_9 - R_2R_3R_4R_6R_7R_8 - R_1R_3R_4R_6R_8R_9 - R_1R_3R_5R_6R_7R_9 \\ & - R_2R_3R_4R_6R_7R_9 - R_1R_3R_5R_6R_8R_9 - R_1R_4R_5R_6R_7R_9 - R_2R_3R_4R_6R_8R_9 \\ & - R_1R_3R_5R_7R_8R_9 - R_1R_4R_5R_6R_8R_9 - R_2R_3R_4R_7R_8R_9 - R_1R_3R_6R_7R_8R_9 \\ & + 2R_2R_5R_6R_7R_8R_9 + R_1R_2R_3R_4R_5R_6R_9 + 2R_1R_2R_3R_4R_5R_7R_9 + 2R_1R_2R_3R_4R_6R_7R_8 \\ & + 2R_1R_2R_3R_4R_6R_7R_9 + R_1R_2R_3R_5R_6R_7R_8 + 2R_1R_2R_3R_4R_6R_8R_9 + R_1R_2R_3R_5R_6R_7R_9 \\ & + R_1R_2R_4R_5R_6R_7R_8 + R_1R_2R_3R_4R_7R_8R_9 + R_1R_2R_3R_5R_6R_8R_9 + 2R_1R_2R_4R_5R_6R_7R_9 \\ & + R_1R_3R_4R_5R_6R_7R_8 + R_1R_2R_3R_5R_7R_8R_9 + 2R_1R_2R_4R_5R_6R_8R_9 + 2R_1R_3R_4R_5R_6R_7R_9 \\ & + R_2R_3R_4R_5R_6R_7R_8 + R_1R_2R_3R_6R_7R_8R_9 + R_1R_2R_4R_5R_7R_8R_9 + 2R_1R_3R_4R_5R_6R_8R_9 \\ & + R_2R_3R_4R_5R_6R_7R_9 + 2R_1R_2R_4R_6R_7R_8R_9 + R_1R_3R_4R_5R_7R_8R_9 + R_2R_3R_4R_5R_6R_8R_9 \\ & + 2R_1R_3R_4R_6R_7R_8R_9 + R_2R_3R_4R_5R_7R_8R_9 + 2R_1R_3R_5R_6R_7R_8R_9 \\ & + 2R_2R_3R_4R_6R_7R_8R_9 + R_1R_4R_5R_6R_7R_8R_9 - 2R_1R_2R_3R_4R_5R_6R_7R_8 \\ & - 3R_1R_2R_3R_4R_5R_6R_7R_9 - 3R_1R_2R_3R_4R_5R_6R_8R_9 - 2R_1R_2R_3R_4R_5R_7R_8R_9 \\ & - 4R_1R_2R_3R_4R_6R_7R_8R_9 - 2R_1R_2R_3R_5R_6R_7R_8R_9 - 3R_1R_2R_4R_5R_6R_7R_8R_9 \\ & - 3R_1R_3R_4R_5R_6R_7R_8R_9 - 2R_2R_3R_4R_5R_6R_7R_8R_9 + 5R_1R_2R_3R_4R_5R_6R_7R_8R_9. \end{aligned}$$

1.4 **PSO_ ARPA Network algorithm**

PSO is an evolutionary technique that needs to generate random numbers. The number and quality of statistics produced affect the results of the PSO algorithm [19]. Throughout the search space is the initial iteration. Figure 3 provides a summary of PSO usage. As shown in the following steps:

Step(1): **INPUT** n {no. of nodes}, lb , ub , Rs equation, a_i, b_i and Rg.

Step(2): Initialization of Population.

Step(3):Calculate Fitness for each Particle, C (total cost) and Rs.

Step(4): Find Best Assignment (Best C and Best Rs).

Step(5): Evolving PSO Population to obtain New Population.

Step(6): **GOTO Step(3)** until the termination condition satisfied.

Step(7): **OUTPUT** Best Assignment (Best C and Best Rs).

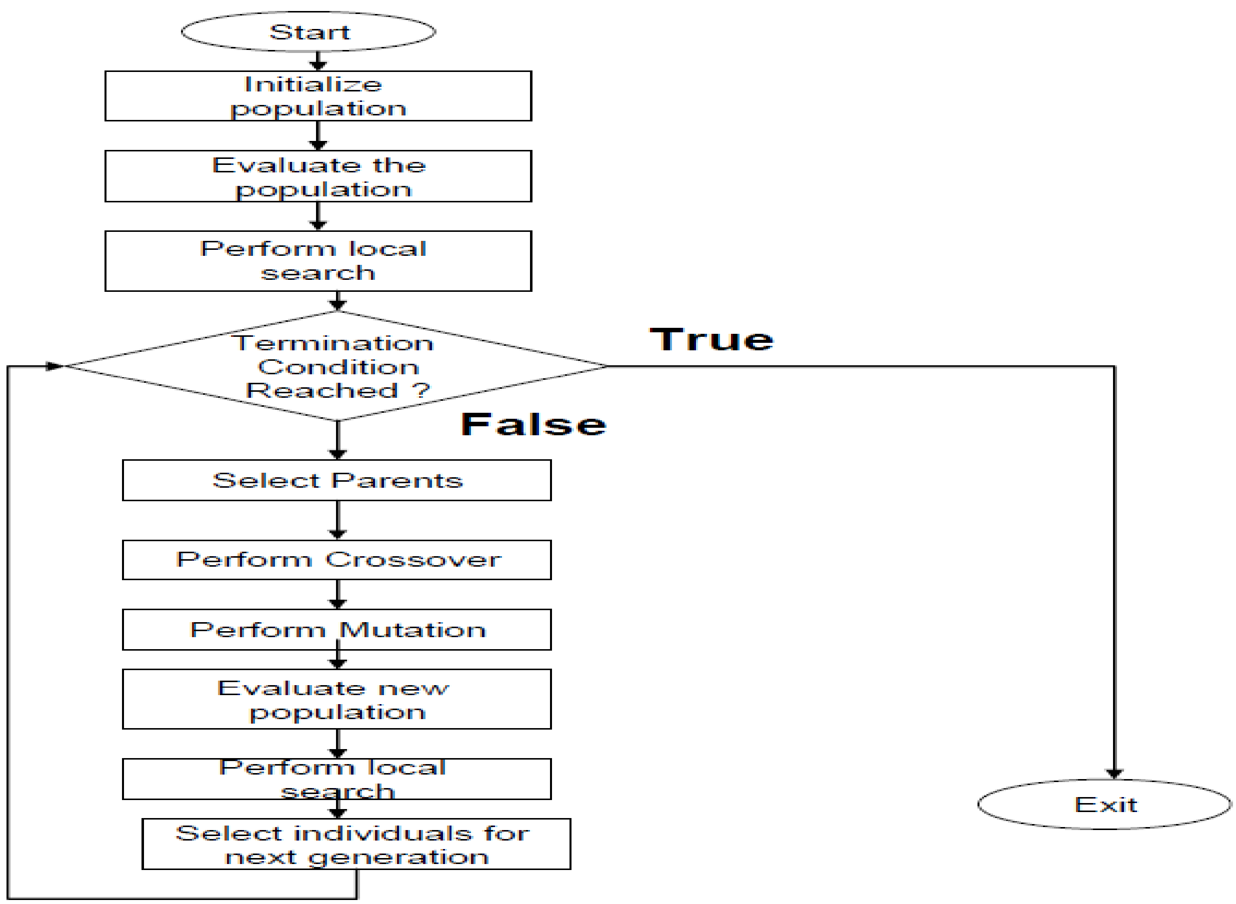

Figure 3. Particle Optimization Flow Chart

1.5 "An exponential feasibility model-based"

"Assume $0 < f_i < 1$ is a feasibility factor [15, 20], $R_{i,max}$ (is maximum reliability), and $R_{i,min}$ (is minimum reliability)".

$$C_i(R_i) = \exp[(1 - f_i)\frac{R_i - R_{i,min}}{R_{i,max} - R_i}],$$

$$R_{i,min} \leq R_i \leq R_{i,max} , i = 1,2, \dots, n.$$

The optimization problem emerges as:

$$Minimize\ C(R_i, \dots, R_i) = \sum_{i=1}^{n} a_i\ \exp[(1-f_i)\frac{R_i - R_{i,min}}{R_{i,max} - R_i}],$$

$$in\ which\ i = 1,2, \dots, n.$$

Subjected to :

$$R_s \geq R_G$$

$$R_{i,min} \leq R_i \leq R_{i,max}, i = 1, \dots, n.$$

Table 1: using PSO and a cost function for reliability allocation optimization.

Components	PSO
R_1	**0.96**
R_2	**0.97**
R_3	**0.92**
R_4	**0.9**
R_5	**0.93**
R_6	**0.9**
R_7	**0.93**
R_8	**0.95**
R_9	**0.95**
R_{system}	**0.99**

6. Conclusion

The purpose of this research is to make a particular complex network more reliable. the system optimization problem that uses computational methods to spread each system component's dependability. Treating this issue as a nonlinear programming problem with constraints on people and material resources (ARPA network dependability) is an alternate approach to solving it. Taking into consideration the locations of different complex network components, the optimal network reliability assignment problem was resolved by applying the Particle Swarm Optimization (PSO) technique. The system's overall assignment was found to be (Rs = 0.99), and the first and second, as well as the fourth and sixth, network components' highest and lowest assignments are shown in the table above. This paradigm has the advantage that all programs can do mathematical calculations because the assignment outcomes were obtained using Matlab software, version R2020a.

Acknowledgements

Not applicable

References

1. Abuali, F.N., D.A. Schoenefeld, and R.L. Wainwright. *Terminal assignment in a communications network using genetic algorithms*. in *Proceedings of the 22nd annual ACM computer science conference on Scaling up: meeting the challenge of complexity in real-world computing applications: meeting the challenge of complexity in real-world computing applications*. 1994.
2. Deb, K. *Genetic algorithm in search and optimization: the technique and applications*. in *Proceedings of International Workshop on Soft Computing and Intelligent Systems,(ISI, Calcutta, India)*. 1998. Proceedings of International Workshop on Soft Computing and Intelligent ….
3. Sulaiman, H.K., et al., *A model for allocation and optimization of reliability complex network using Particle Swarm Optimization and Genetic Algorithms.* Computer Science, 2023. **18**(2): p. 225-231.
4. Hasslinger, G., S. Schnitter, and M. Franzke, *The efficiency of traffic engineering with regard to link failure resilience.* Telecommunication Systems, 2005. **29**: p. 109-130.
5. Elbaum, R. and M. Sidi, *Topological design of local-area networks using genetic algorithms.* IEEE/ACM transactions on networking, 1996. **4**(5): p. 766-778.
6. Sulaiman, H.K., F.H. Ali, and Z.A.H. Hassan. *Computational models for allocation and optimization of reliability for ROSS network*. in *AIP Conference Proceedings*. 2023. AIP Publishing.
7. Kirkpatrick, S., *Optimization by simulated annealing.* Science, 1984. **220**: p. 4598.
8. Abed, S.A., H.K. Sulaiman, and Z.A.H. Hassan. *Reliability Allocation and Optimization for (ROSS) of a Spacecraft by using Genetic Algorithm*. in *Journal of Physics: Conference Series*. 2019. IOP Publishing.
9. Fidler, B. and M. Currie, *The production and interpretation of ARPANET maps.* IEEE Annals of the History of Computing, 2015. **37**(1): p. 44-55.
10. Davis, L., et al. *A genetic algorithm for survivable network design*. in *Proceedings of the 5th International Conference on Genetic Algorithms*. 1993.
11. Back, T., U. Hammel, and H.-P. Schwefel, *Evolutionary computation: Comments on the history and current state.* IEEE transactions on Evolutionary Computation, 1997. **1**(1): p. 3-17.
12. Liu, X., *Network capacity allocation for traffic with time priorities.* International Journal of Network Management, 2003. **13**(6): p. 411-417.
13. Burrell, A., D. Makrakis, and P. Papantoni-Kazakos, *Traffic monitoring for capacity allocation of multimedia traffic in ATM broadband networks.* Telecommunication Systems, 1998. **9**: p. 173-206.
14. Gerla, M., J.S. Monteiro, and R. Pazos, *Topology design and bandwidth allocation in ATM nets.* IEEE Journal on Selected Areas in Communications, 1989. **7**(8): p. 1253-1262.
15. Kim, S.-B., M.-J. Kim, and S.-I. Lee. *Mathematical models for dimensioning of ATM networks*. in *Proceedings of GLOBECOM'95*. 1995. IEEE.
16. Davis, L. and S. Coombs. *Genetic algorithms and communication link speed design: theoretical considerations*. in *Proceedings of the Second International Conference on Genetic Algorithms on Genetic algorithms and their application*. 1987.
17. Thangaraj, R., et al., *Particle swarm optimization: Hybridization perspectives and experimental illustrations.* Applied Mathematics and Computation, 2011. **217**(12): p. 5208-5226.
18. Feng, X., et al., *Reliability of Electric Vehicle with Wind Turbine Based on Particle Swarm Optimization.* Chemical Engineering Transactions, 2018. **66**: p. 1291-1296.
19. Gad, A.G., *Particle swarm optimization algorithm and its applications: a systematic review.* Archives of computational methods in engineering, 2022. **29**(5): p. 2531-2561.
20. Rios, M., V. Marianov, and C. Abaroa. *Design of heterogeneous traffic networks using simulated annealing algorithms*. in *International Conference on Information Networking*. 2005. Springer.