International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 6, Issue 8, ISSN No. 2455-2143, Pages 97-103
Published Online December 2021 in IJEAST (http://www.ijeast.com)

GENETIC ALGORITHM FOR WIRELESS SENSOR NETWORKS

Radha Krishna Karne
Assistant Professor
Dept. of ECE Balaji Institute of Technology & Science, Narsampet

Dudimetla Prasad
Assistant Professor
Dept. of ECE Balaji Institute of Technology & Science, Narsampet

Uzma Naseem
Assistant Professor
Dept. of ECE Balaji Institute of Technology & Science, Narsampet

Ashok Battula
Assistant Professor
Dept. of ECE Balaji Institute of Technology & Science, Narsampet

Karthik Kumar Vaigandla
Assistant Professor
Dept. of ECE Balaji Institute of Technology & Science, Narsampet

***Abstract*—Ad hoc networks are made up of a collection of wireless mobile nodes that form a temporary network with no pre-existing infrastructure or centralized management.Routing policies are crucial in determining how traffic is forwarded across a network. Adhoc networks necessitate a routing method that is very adaptable. Finding the shortest path (SP) between source and destination in a specific period of time to meet Quality of Service standards is one of the most common issues in these networks (QoS). QoS routing is difficult in an Ad hoc network because the topology changes frequently and it takes time since many QoS criteria such as distance, cost, and energy are all variable, and the state information supplied for routing is inherently faulty. The optimum path for Adhoc networks is found using a Genetic Algorithm (GA) in this paper.GA uses natural evolution-inspired methodologies to find answers to optimization problems. Crossover and mutation operations, as well as the proper chromosome structure, are all defined.**

***Keywords*— Adhoc Network, QoS, Genetic Algorithm, computer simulation, optimization.**

I. INTRODUCTION

A wireless Ad hoc network is a non-centralized network type.In this situation, the sensor nodes are unfamiliar with their network's topology and must instead discover it[8,9].Because it does not rely on pre-existing infrastructure, such as routers in wired networks or access points in infrastructure, the network is known as Adhoc. Because each node contributes to routing by passing data to other nodes, network connectivity determines which nodes send data. The process of determining which paths in a network to send data along is known as routing. The routing algorithm [3] must collect, organize, and transmit information about the condition of the network. It should be capable of establishing feasible routes between nodes, sending traffic in the desired direction, and achieving high performance. A genetic algorithm (GA) is a search tool for finding perfect or approximate answers to optimization and search issues in computing. GAs are a sort of evolutionary algorithm (EA) that uses evolutionary biology processes such as inheritance, mutation, selection, and crossover to make decisions. Using operators like selection, crossover, and mutation, new children are created/evolved from chromosomes. The chromosomes that match the best are passed down to the following generation [6].The weaker candidates have a lower likelihood of being promoted to the following generation. People will live, reproduce, and die is

International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 6, Issue 8, ISSN No. 2455-2143, Pages 97-103
Published Online December 2021 in IJEAST (http://www.ijeast.com)

determined by the GA object. It also keeps track of statistics and determines how long the evolution will last. This procedure is repeated until the chromosomes have found the most appropriate solution to the challenge. In summary, the population's average fitness improves with each iteration, therefore repeating the process for several iterations yields better outcomes. In the majority of GAs, there is no obvious stopping condition. When it's time to stop, we need to let the algorithm know. Genetic algorithms are divided into four groups[5].The first is the standard' simple genetic algorithm [1],' which employs non-overlapping populations to create a whole new group of people with each generation. The second method is based on overlapping populations and is a steady-state genetic method[2,4].'We can choose how much of the population should be replaced each generation in this version. The 'incremental genetic algorithm,' in which each generation has just one or two children, is the third type. Custom replacement methods show how the following generation should be absorbed into the population using incremental genetic algorithms. A newly produced child, for example, could take the place of its parent, a random person in the population, or the person who is the most like it. The 'deme' genetic algorithm is the fourth type. Using a steady-state method, this algorithm grows many populations in parallel. The algorithm migrates some persons from each group to one of the other populations every generation. We employed a steady-state genetic method in our study.

II. PROPOSED MODEL

The Ad hoc network under consideration is represented as g(x, y) a fully connected graph with X nodes, given the collection of nodes (X).The total distance is equal to ∑d the sum of the distances between each of the path's nodes. The sum of the energy between a path's nodes is its total energy (y). The purpose is to discover the shortest path between the source and destination nodes, Xsrc and Xdest. We start with a small group of people (first generation). Each string's fitness value in this generation is determined. A new generation is then created using the reproduction operator. Before re-examining all solutions, a pair of strings from the new generation is picked and a crossover is performed, with a specific probability gene and modified. This process is continued until all of the termination conditions are met. The best solution of all time is saved and returned at the end of the process. Figure 1 represents flowchart of Genetic algorithm.

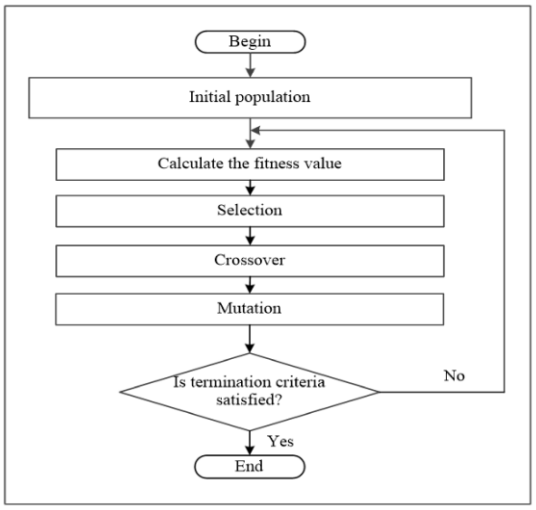

Fig. 1 Flowchart of Genetic Algorithm

A. Mathematical Model

Each link x1->x2, E in an undirected graph g(x, y) is connected with two criteria. The set of nodes is X, while the collection of edges is z.For i=1 to |z|, the distance di(x1->x2) and the energy y(x1->x2) are the same. The goal is to discover the shortest path that maximizes the fitness function's value between a given source 'S' and a particular destination 'D.'

$$F = \sum y \, X \frac{\text{path}}{\sum d}$$

Path values is written as 1000 if the path exists and it is written as 1 if there is no path available.

Algorithm 1: Genetic Algorithm Routing in Ad hoc Networks
Step 1: Based on the number of nodes in the network, create an initial population of chromosomes.
Step 2: Set the maximum number of iterations and the termination count.
Step 3: 1: maximum number of iterations

a. Determine the fitness of each chromosome and save the results in a fitness array. Find max fitness of populations.
b. b. If the highest fitness inside the termination count does not change, then break out because there has been no improvement.
c. c. Using a roulette wheel, choose two parents (Parent1 & Parent2) from the population for reproduction based on their fitness rating.
d. Using the crossover operator, create two offspring (Child1 and Child2).
e. On Child1 and Child2, use the mutation operator.
f. Determine Child1's and Child2's fitness levels.

International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 6, Issue 8, ISSN No. 2455-2143, Pages 97-103
Published Online December 2021 in IJEAST (http://www.ijeast.com)

g. Find the chromosomes in the present population that are the least fit.
h. If child1 and child2's fitness exceeds that of the population's least fit, such chromosomes should be replaced with child 1 and child 2.
i. Step 3 should be repeated until the maximum number of iterations is reached.

Step 4: Finish
Step 5: Find the chromosome in the population with the highest fitness. This chromosome's genes determine the shortest path between the source and the destination.
Step 6: Come to a halt.

III. IMPLEMENTATION

The GA is used to find the optimum path for networks with N number of nodes constructed in NS2. A fixed, limited alphabet is used to explain the genetic structure of a chromosome. The alphabet 0 1 is commonly used in GAs [2,4]. Table No. 1 shows an example of a chromosome that has been implemented. A path connecting nodes n1, n2, and n4 is represented by the chromosome. The graph's nodes are represented as 1, whereas nodes that aren't part of the graph are represented as 0. As a result, the chromosomal number is '1 2 4 0 0'. The number of nodes determines the size of the starting population. The amount of time it takes for the software to run is recorded. This allows us to estimate how long the network will take to use the GA.

A. Fitness Function

To see if the chromosome has any routes, a function is written. Equation 1 is used to determine fitness based on distance and energy, which allows for fitness-based chromosomal selection. To improve the possibility of chromosomes with a pathway being chosen in the following stage, the equation is multiplied by a big value.

B. Selection

We used the 'Roulette wheel selection' technique in our implementation, which we narrowed down as the optimum selection technique. The fundamental benefit of proportionate roulette wheel selection is that no individuals are excluded from the population, allowing for the selection of all of them. As a result, demographic diversity is preserved. Two chromosomes from the initial population are chosen based on fitness values. The chosen chromosomes will be referred to as parent chromosomes, and they will be passed on to the next step, when further genetic operators will be applied.

Algorithm 2: Roulette Wheel Selection
Step1: Using the equation fitness Sum=cum u, find the sum of all chromosomal fitness values [Cumulative Sum] (fit)
Step 2: Create two distinct random integers. r 1 and r 2
Step 3: If the cumulative Sum of ith chromosome >= r 1 in the population, ith chromosome is chosen as parent1.
Step 4: If the cumulative Sum of ith chromosome >= r 2 in the population, ith chromosome is chosen as Parent 2 & Parent 2! =Parent 1
Step 5: Come to a halt.

C. Crossover

On both of the designated chromosomes, crossover is conducted. Crossover can be accomplished in a number of ways. We used a single point crossover in our GA deployment. Selecting a point on the parent organism strings is required for single point crossover [2,3]. All data in either organism string is shared between the two parent organisms after that point. The animals that are handed on to the following level are the freshly created children. Experimentation determines the possibility of crossover, and we've chosen a crossover of 0.9. A single site of crossover is depicted in Figure 2.

Algorithm 3: Single Pont crossover Operation
Step 1: Determine the number of common nodes in parent1 and parent 2 using index.
Step 2: Make a random integer that will be used to designate the crossing points of parent1 and parent 2.
If index is greater than zero,
a. All genes from parent 2 are transferred beyond the crossover point, whereas genes from parent 1 are kept up to the crossover point, resulting in child1.
b. Parent 1 copies all genes beyond the crossover point, while parent2 genes are duplicated from 1 to the crossover point, resulting in child 2.
c. If necessary, remove the duplicated nodes from child1 and child 2. Child 1 and child2 are the names of the children of parents 1 and 2.
Step 4: Come to a halt.

Figure 2 represents Single point crossover, in which child 1 & child 2 is getting different features from parents 1 & 2

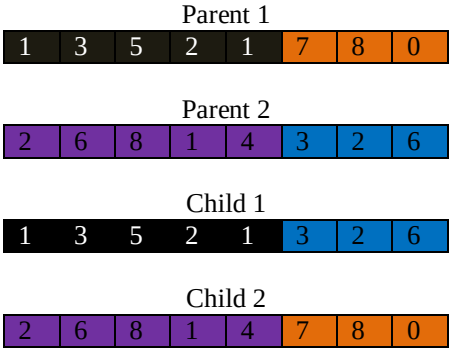

Fig. 2 Single point Crossover

International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 6, Issue 8, ISSN No. 2455-2143, Pages 97-103
Published Online December 2021 in IJEAST (http://www.ijeast.com)

D. Mutation

The GA becomes more unpredictable as a result of the mutation process. The multi-point mutation operator substitutes a random value between the upper and lower gene borders for the value of two specified genes. The multi-point mutation operator substitutes a random value between the upper and lower gene borders for the value of two specified genes. As shown in Figure 3, two random points are built, and the node at that position is replaced with a randomly generated node within the range, taking care to verify that the replacement node does not already exist in the Child chromosome. The fitness of each of the kid chromosomes is calculated after the mutation process and compared to the fitness of all other chromosomes in the beginning population. If any of the chromosomes in the original population are less suited than the offspring, they are eliminated and replaced with the child chromosome. This process is carried out until the termination condition is satisfied.

algorithm 4: Multi-point Mutation Operation
Step 1: Based on the number of genes in the child, create I, J two random mutation sites on the child.
Step 2: mutnum1= Create a node at random within the range
a. while Ith point on child
b. Child(I)=mutnum1 if mutnum1 is not in child.
otherwise, repeat Step 2.
Step 3: While on child,
a. Mutnum2=Generate a node within the range at random.
b. Mutnum2=Generate a node at random within the range Child(J)=mutnum 2 if mutnum 2 is not in child. Step 3 should be repeated if necessary.
Step 4: Come to a halt.

E. Termination

No change in population fitness and stall generation are examples of algorithm halting circumstances. Maximum number of generations The requisite fitness, i.e. the best path, is assumed when the termination criteria are metWe defined a number of iterations as well as a criterion that the fitness created does not change after a specific number of runs in our code. The clock is stopped at this point to record how long it took the algorithm to solve the routing problem in total.

IV. SIMULATION RESULTS

In the network configurations, 50 sensor nodes are randomly deployed in the 1000× 1000 meter field area using software tool NS2(Network Simulator 2). In these configurations, different scenarios are verified and the corresponding results are discussed in detail. Table 1 describes the parameters considered for network construction for verification of genetic algorithm.

Parameter	**Type**
Network size	1000 X 1000
Number of nodes	50
Simulation time	200
Type of Routing	GENETIC
Initial energy of node	100
MAC	IEE 802_11
Queue type	Drop Tail Pri Queue
Type of channel	Wireless channel
Agent	UDP
Source node	7
Destination node	12
Simulation start time	15.0 sec
Simulation stop time	195.001 sec
Packet size	2000
Packet Interval	0.05
Data rate	1.3Mbps

Table 1: Network Parameters

International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 6, Issue 8, ISSN No. 2455-2143, Pages 97-103
Published Online December 2021 in IJEAST (http://www.ijeast.com)

Figure 3 depicts the fitness function and forward path for the network with 50 nodes, in which 7 being the source node and 12 being the destination node. Table 2 represents the simulation results of the network which we have considered for analyzing genetic algorithm and the same is shown in figure 4 which represents simulation results after running code.

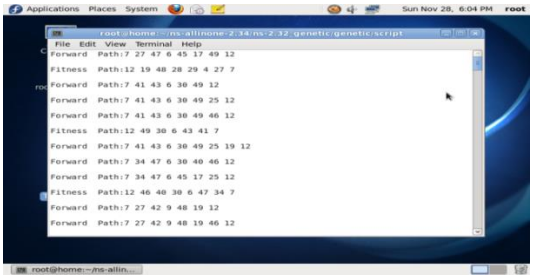

Fig. 3. Forward path and Fitness functions

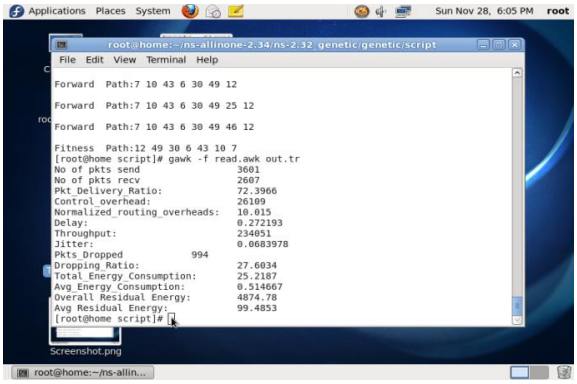

Fig. 4. Simulation results after running code

Parameter	**Type**
Number of packets sent	3601
Number of packets received	2607
PDR	72.3966
Control overhead	26109
Normalised_routing_overhead	10.015
Delay	0.272193
Throughput	234051
Jitter	0.0683978
Packets Dropped	994
Dropping Ratio	27.6034
Total_energy consumption	25.2187
Average_energy consumption	0.514667
Overall residual energy	4874.78
Average Residual energy	99.4853
Number of nodes	50

Table 2: Simulation results after running code

Figure 5a & 5b represents Visualization of NAM file for the network we have considered, where nodes are sending packets from source to destination through other nodes called auxiliary nodes.

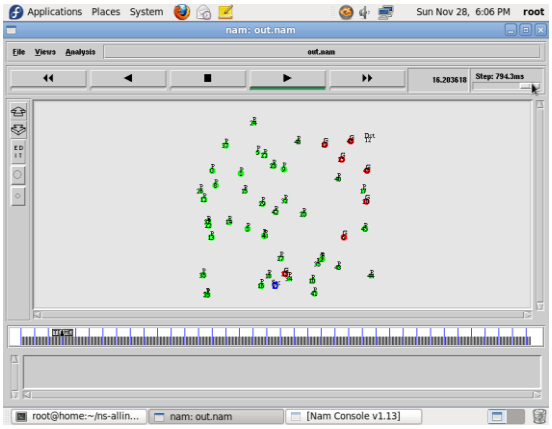

Figure 5a: Visualization of Nam file of the network

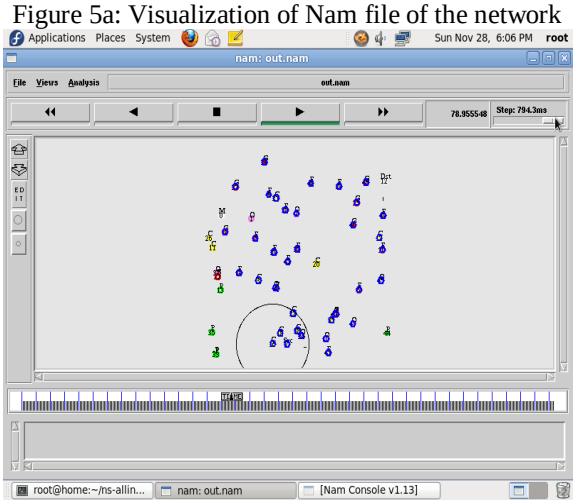

Figure 5b: Visualization of Nam file of the network

Jitter is when there is a time delay in the sending data packets over network. It occurs by network congestion, and sometimes route changes. Figure 6 represents the relation between the two parameters simulation time and Jitter. Using GA approach Jitter in the network is drastically decreased with the same simulation parameters considered with AODV protocol.

International Journal of Engineering Applied Sciences and Technology, 2021
Vol. 6, Issue 8, ISSN No. 2455-2143, Pages 97-103
Published Online December 2021 in IJEAST (http://www.ijeast.com)

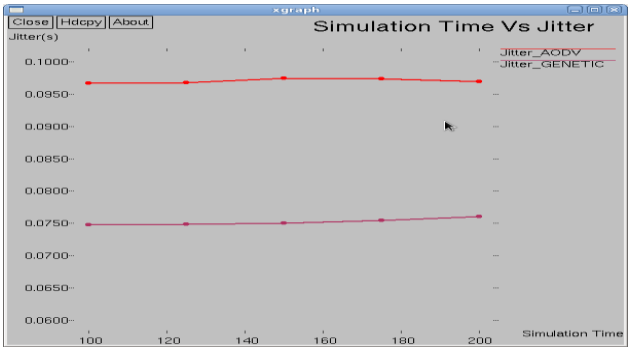

Figure 6: Simulation Time Vs Jitter

The packet delivery ratio (PDR) is the total number of data packets arrived at destinations divided by the total data packets sent from sources. In other words Packet delivery ratio is the ratio of number of packets received at the destination to the number of packets sent from the source. Figure 7 represents the graph providing the relation between Simulation time and PDR. It is providing good PDR around 67% for GA compared to AODV providing 52%.

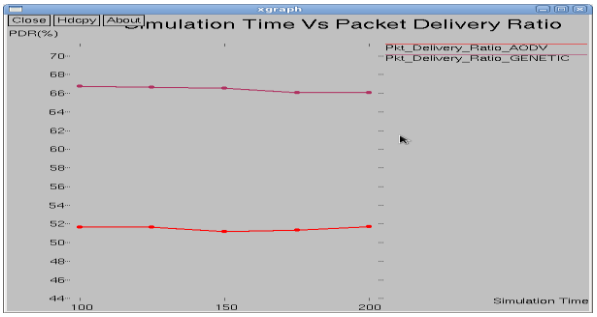

Fig 7: Simulation Time Vs PDR

Packet loss occurs when one or more packets of data travelling across a computer network fail to reach their destination. Figure 8 represents the relation between Simulation time and Packets dropped. As the simulation time is increases packets dropped is reduced in GA compared to the AODV.

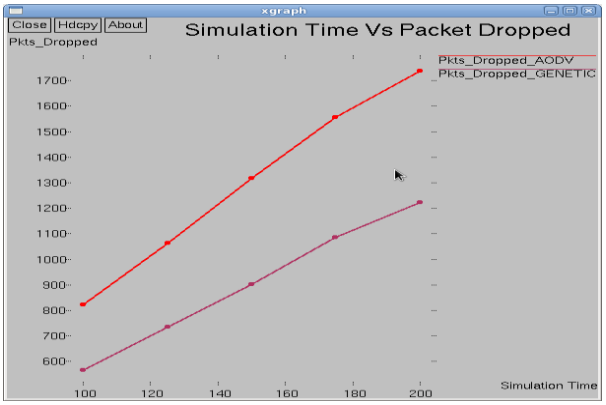

Fig 8: Simulation Time Vs Packets Dropped

Throughput is the actual amount of data that is successfully sent/received over the communication link. Figure 9 represents the relation between Simulation time and throughput. As the simulation time is changes the throughput remains unchanged in both the protocols, but it is improved in GA compared to AODV.

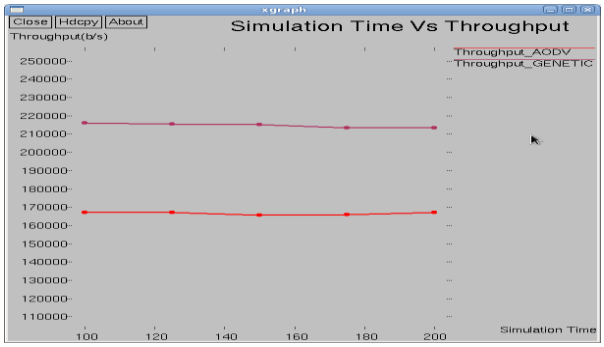

Fig 9: Simulation Time Vs Throughput

V. CONCLUSION

In this paper, genetic algorithms were utilized to determine the best pathways between a given source and a specified destination in an Adhoc Network. To depict the different solutions of the ideal path in Adhoc networks, a new Steady State node based chromosomal structure has been designed. In addition, the crossover and mutation operators have been designed to be more efficient. The Genetic algorithm has been put to the test on a number of network topologies, with the findings published. The overall performance of the genetic algorithm in terms of discovering optimal pathways under cost/distance and energy limitations has been excellent.

VI. REFERENCES

[1]. Dr. Rakesh Kumar, Mahesh Kumar, "Exploring Genetic Algorithm for Shortest Path Optimization in Data Networks ", Vol. 10 Issue 11 (Ver. 1.0) October 2010.

[2]. Yousra Ahmed Fadil, "Routing Using Genetic Algorithm for Large Networks", Vol. 03 , No. 02 , pp. 53 -70 , December 2010.

[3]. Susmi Routray, "An enhanced genetic algorithm for dynamic routing in ATM networks", Associate Professor (ITM), IMT Ghaziabad, India – 201001, Vol. 16 No.2 June , 2010 .

[4]. Chao-Hsien Chu , G. Prem kumar , Hsinghua Chou , "Theory and Methodology Digital data networks design using genetic algorithms", European Journal of Operational Research 127 (2000) .

[5]. Bilal Gonen, "Genetic Algorithm Finding the Shortest Path in Networks", Department of Computer

Science and Engineering, University of Nevada, Reno, Reno, Nevada, U.S.A.

[6]. Usama Mehboob, Junaid Qadir, Salman Ali, and Athanasios Vasilakos, "Genetic Algorithms in Wireless Networking: Techniques, Applications, and Issues", arXiv:1411.5323v1 [cs.NI] 19 Nov 2014

[7]. Pravin. M, Munaf.S and Vetrivelan. P, " An approach for DSP routing in MANET using genetic algorithm", Research International Journal of Recent Scientific Research vol. 3, Issue, 7, pp. 647 - 651, July, 2012

[8]. Radha Krishna Karne, Dr,Sreeja TK, "Review On Vanet Architecture And Applications", Turkish Journal ofComputer and Mathematics Education Vol.12 No.04 (2021), 1745 – 1749

[9]. Radha Krishna Karne Nagaraju Bompelli, Ramadevi Manchala, "IoT Based Smart Sensor Soc Architecture For The Industrial Internet of Things", The International journal of analytical and experimental modal analysis, Volume 12 Isuue X(2020)