Genetic Algorithms in Network Routing

Dhanushga Lionel
Ontario Tech University
Ontario, Canada
dhanushga.lionel@ontariotechu.net

Abstract

A routing algorithm is a set of operations used to direct Internet traffic. When a packet leaves its source, there are many paths that it can take to get to the destination. The routing algorithm uses mathematical techniques to find the best path to take. IP routing protocols compute paths based on the topology of the network, without regard to the current traffic load on the routers. The genetic algorithm (GA) is used for routing optimization. It uses the principles of natural selection and genetic mutations to solve the routing protocol. The algorithm finds the shortest path between the source and destination nodes. This type of algorithm is excellent for searching through large and complex data sets.

Index terms - Genetic Algorithms, Dijkstra's Algorithm, Routing Algorithm

Introduction

In networking procedures, packets need to travel from one location to another location, which can be far away. Routing is used so that packets can travel safely to the intended destination. However there are many cost factors such as

- Hops: number of intermediate routers between source and destination router
- Latency: time delay in processing packet through the router over the route
- Congestion: length of packet queue at the router
- Bandwidth: available capacity of router to support network traffic. This decreases as network traffic increases.
- Reliability: relative amount of downtain that a particular router could experience due to malfunctions
- Maximum Transmission Unit (MTU): largest packet size that router can forward without fragmenting said packet.

To determine the optimal route packets can take routing algorithms are used. Routing algorithm is a mathematical procedure that a dynamic router uses to calculate entries for its routing table [1]. There are 2 common types of routing algorithms, adaptive algorithm and non-adaptive algorithm.

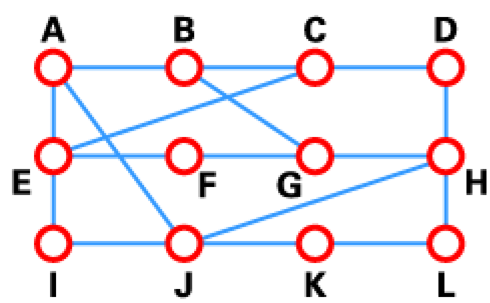

Figure 1: Sample Routing Algorithm

. Adaptive routing algorithms change as their routing decision to reflect changes in topology or traffic load on links. The opposite is non-adaptive algorithms. OSPF (open shortest path first) protocol is used to find the shortest route between the source and destination node. OSPF is a dynamic protocol. This paper will expand on how Genetic Algorithms are better alternatives to OSPF.

Shortest Path Algorithm (Dijkstra's Algorithm)

Choosing the shortest path for routing means choosing the optimal path packets will take when traveling in the network. The most common type of shortest path Algorithm is Dijkastra's algorithm[6]. The algorithm calculates the shortest path between two points on a network, using a graph of nodes and edges. Each node is assigned a cost value. The algorithm divides the vertices into two sets: tentative, and permanent sets [7]. The algorithm chooses the nodes, makes them tentative and examines them. If these nodes pass the reiteria the nodes are then made permanent.

The following steps define Dijktra's algorithm [7][6].

1. Start with the source node: the root
2. Assign the root with cost 0 and make it a permanent node
3. Examine each neighboring node of the node that was the last permanent node
4. Assign a cumulative cost to each node and make it tentative
5. Among the list of tentative nodes:
    a. Find the node with smallest cumulative cost and mark it as permanent. Permanent nodes will not be checked ever again; its cost is final and is recorded.
    b. If a node can be reached from more than one direction, select the direction with shortest cumulative cost
6. Repeat steps 3-5 until every node becomes permanent.

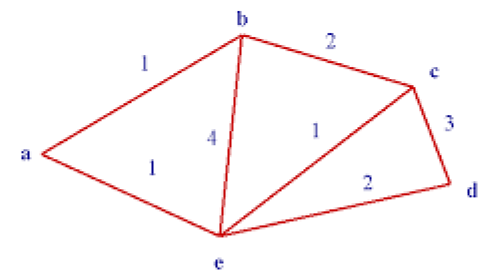

Figure 2: Network Topology

As shown in Figure 2, by applying Dijkatra's algorithm the shortest between node A and node D, the optimal path is A->E->D.

Genetic Algorithm

Genetic algorithms, GA, are techniques inspired from natural selection genetics and evolution[6]. GAs are used to generate high-quality solutions for optimization problems and search problems [8].

GAs simulate natural selection, so individuals who adapt to changes are able to survive and reproduce. At its simplest, it is "survival of the fittest".

The thought process behind the creation of GAs was to develop a system that is robust and adaptable to the environment as the natural system. The GA is initialized with a population of candidate solutions. Each candidate is usually coded with a binary string called a chromosome [6].

A set of chromosomes form a population, which is evaluated and ranked by a fitness evaluation function [7]. The fitness function is instrumental for the GA. The fitness function provides information on how good each candidate really is. The initial population is generated at random. Then the candidates evolve from one generation to the next in 3 main steps, fitness evaluation, selection and reproduction [7].

First, the population is evaluated using the fitness evaluation function. The population is ranked based on their fitness. A new generation is created, the goal is to improve fitness. To improve the fitness there are 3 main operators with probabilistic rules [7], reproduction, crossover and mutation. Reproduction is applied to the current population. This results in the intermediate population.

Second, the GA selects the parents, those who have a better chromosome are selected. This can be accomplished by ranking the chromosome or choosing a better fitness value.

Third, the GA reproduces the children from the parents using crossovers and/or mutation operators [7].

Crossovers is the mating between two individuals. Two individuals are selected, and crossover sites are chosen randomly. The genes at each crossover site are exchanged, which can make a new individual [8].

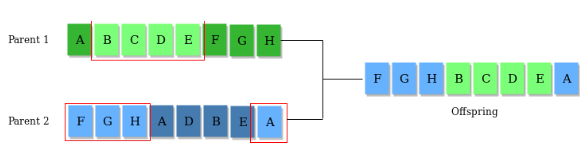

Fig 3: Crossover of Genetic algorithm [8]

Mutation works by inserting random genes in an individual to maintain diversity in the population and avoid premature convergence [8].

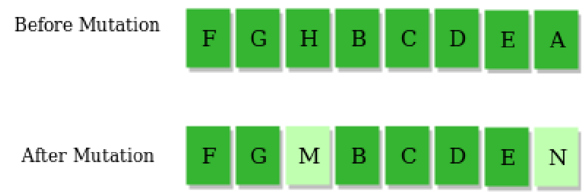

Fig 4: Mutation of Genetic algorithm [8]

Example GA in Networking

This section will present an example of using Genetic algorithm in a networking problem to find the shortest path in the most efficient and effective method.

There is a network with N number of nodes. The cost between nodes is recorded for optimization. The total cost is the sum of the number of hops to get from the source node to the destination node.

In this proposed problem, the feasible solution is one that can get from source to destination node. Optimal solution is the shortest solution. In the beginning, a random population is generated, which represents both feasible and infeasible solutions. Unfeasible solutions are solutions that cannot reach the destination node.These unfeasible solutions are assigned the lowest fitness function which is 0. The proposed problem is as follows.

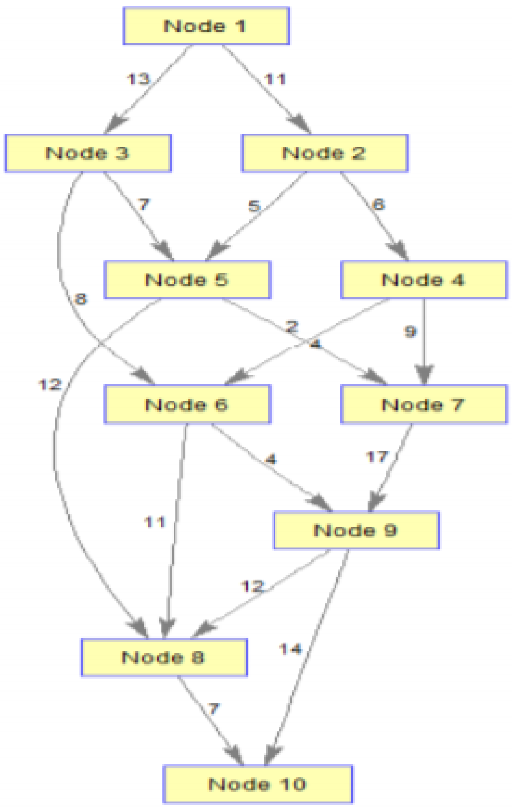

Figure 5: Proposed problem

In figure 5, each node is assigned a number from 1 to 10. The source node is 1, and the destination node is 10. There are a number of feasible options, such as 1-3-5-7-9-8-10, but this is not optimal.

The chromosome is represented by string bits which are in binary form. After crossover and mutation operators are applied to these chromosomes. During crossover, the string that has the best fitness wll be selected to be the parent. A second parent will be chosen which contains common numbers as the first parent.

Parent 1: 1-2-4-7-9-10
Parent 2: 1-3-5-7-9-8-10

The underlined parts of each parents, are the crossover parts. If we apply crossover to these two parent chromosomes, the resulting child chromosomes are:

Child 1: 1-2-4-7-9-8-10
Child 2: 1-3-5-7-9-10

Each path is encoded using binary numbers where each node in the chromosome is encoded by a 4-bit binary number. This is shown in Table 1 below.

Node	**Binary Code**	**Linked Node**
1	0001	2,3
2	0010	4,5
3	0011	5,6
4	0100	6,7
5	0101	7,8
6	0110	8,9
7	0111	9
8	1000	10
9	1001	8,10
10	1010	Destination

Table 1: Binary coding of network nodes

Each chromosome used illustrates a possible solution for this network. The maximum length of the chromosome is the number of nodes multiplied by the number of bits used in the particular node. Since is node is represented by 4 bits, then there is a total of 40 bits in a chromosome (10*4).

The initial population of chromosomes can be randomly generated, so that each chromosome has random genes, even though the source and destination nodes are constant. After the population is randomly generated, the fitness function is established. The cost of the path used by each node is used to calculate the fitness. By increasing the fitness value, the most efficient path is chosen.

The fitness function equation is $F(X) = (X/\Sigma X) + (n/\Sigma n)$. X is the number of paths possible, and Σn is the number of nodes in the network. ΣX is the summation of all paths.

After finding F(X) for each path, the fitness is calculated by the equation Fitness = F(X)/Max F(X).Max F(X) is the maximum F(X) in the list. This means that the maximum F(X) will be given a fitness of 1.

GA Issues

Genetic Algorithms provide useful problem solving and optimized solutions, but it does come with some limitations. Issues that may be present in GA can include real time applications, parameters, wrong choice of fitness function, deception, and bias. In this section, these topics will be explained into further detail.

Firstly, the use of GA is not really suited for real time applications because of varying response times. Real time applications require immediate response times in their processes. With GA, this is not always possible since there the factor of population and generations come into play. The GA may not always solve a task in time, which can hinder the quality of service for an application. In particular, for network routing, GA is often forced to terminate after a set amount of iterations due to the network's constraints.

Another issue with GA is the parameters that can be associated with it. GAs need the right population size, mutation, and crossover rate to produce the correct results. For example, a small population size will not give a GA enough time to produce the most accurate results, but a higher population can take longer to find the optimal solution. Also, a high frequency of genetic change with mutation and poor selection scheme will result in the population changing too fast, thus not having any convergence [4].

The fitness function is a really important factor for a GA. The objective is to code in a way so that a higher fitness can be attained and better solutions are found for the problem. Coming up with the proper fitness function is often the hardest part when formulating a problem using GA. The wrong choice of a fitness function can lead to the GA not finding a solution or providing the wrong solution, which can be catastrophic. For example, a fitness function of zero is not good, as it does not provide an idea of how close the solution is to the correct answer. The function must give low values to bad solutions and high values to good solutions, until the best solution is found [4].

Certain deceptive functions of GA can sometimes be difficult to optimize. A building block may still have bad chromosomes, thus causing the building block to fail. A solution to this problem is to use a messy GA. Messy genetic algorithm solves problems by combining small building blocks that are well tested to form long strings that cover all the features of the problem [5].

Lastly, there is a risk of slow convergence and settling for local minima due to inappropriate parameters. If the GA lacks population diversity or has a small population, then it can fail to converge to a global optimum. This causes a “bias” in the GA. This is due to poor selection of fitness functions, leading to bad chromosomes. There is no guarantee that the GA is able to find the proper solution, and it can lead to problems if the population has a lot more subjects.

GA Application

The applications of GAs play a critical role in network routing. In this section, the main focus will be on the routing aspects, but will also explain extra features of networking that are required such as quality of service. However, the main application that will be explained is for routing and will focus on the usage of various techniques. The techniques will involve the usage of the shortest path routing problem, multicast and broadcast routing, dynamic shortest paths, and hybrid genetic algorithms.

GAs are well suited for the shortest path routing problem as they have been used for developing adaptive routing solutions. The routing table needs to have a pair for each source destination and they need to be constructed. The number of nodes would depend on the topology of the network and if the network is large enough, the number of routes for a source destination becomes huge. In simulation, crossover and mutation are run on variable length chromosomes [9].

The proposal of GA being used for multicast and broadcast routing has also been studied upon. The routing scheme is discussed so it meets bandwidth and delay, while maintaining a quality of service in wireless networks. All the possible routes are encoded between the source and destination and implemented into the GA in the form of population chromosomes. The GA then uses its evolutionary techniques to choose the best chromosomes for repopulation. This is done by using a specific fitness function, which is formed using different constraints of the bandwidth. Another proposal for the broadcast scheduling problem was discussed. The steps to solve this algorithm was to first find a possible solution, then obtain a maximal transmission by using the fittest chromosomes through the GA. Multiple other operators are then applied to the throughput. These factors can be either mutation or crossovers [10].

There has also been discussion how GA can be used for the dynamic shortest path routing problem. Algorithms such as Dijkstra perform well in a fixed infrastructure, but on a dynamic scale, they can be unusable due to the high computational complexity. The GA uses immigrants and memory schemes in solving the dynamic shortest path routing problem in real world networks [11].

Finally, the use of hybrid GAs have been proposed in network routing. Ant algorithms were proposed, combined with GAs for route discovery. The observations were that the ant algorithm led to a reduction in size for the routing table. The algorithm also belongs to a broader class of AI called swarm intelligence. The use of the ant algorithm and GA techniques led to effective computation of routes between any node pairs. The proposed hybrid algorithm creates a starting population, determines the forward ants, generates a new population using operators and fitness functions, and finally updates the routing table upon completion. The algorithm performs well in both generating routes and maintaining efficient load balancing [12].

Conclusion

In conclusion, genetic algorithms are efficient and effective methods for finding the shortest path in a large network. The chromosome length depends on the number of nodes in the network. Genetic algorithms will produce the same or better results than Dijkastra's algorithm. Issues in GA can include using real time applications, parameters, wrong choice of fitness function, deception, and bias. And applications required various techniques using the shortest path routing problem, multicast and broadcast routing, dynamic shortest paths, and hybrid genetic algorithms. In the future, genetic algorithms can be improved by reducing the length of chromosomes used in the nodes, which will reduce the complexity.

References

Introduction

[1] "Routing Algorithm." *Network Encyclopedia*, 21 Aug. 2019, networkencyclopedia.com/routing-algorithm/.

[2] "Computer Network: Routing Algorithm - Javatpoint." *Www.javatpoint.com*, 2018, www.javatpoint.com/computer-network-routing-algorithm.

[3] "Classification of Routing Algorithms." *GeeksforGeeks*, 17 Oct. 2019, www.geeksforgeeks.org/classification-of-routing-algorithms/.

[4] "Genetic Algorithms." *Genetic Algorithms: The Reality*, 2006, www.doc.ic.ac.uk/project/examples/2005/163/g0516312/Algorithms/Reality.html.

[5]M. Mitchell, An introduction to genetic algorithms. MIT press, 1998

[6] Kumar, Kuldeep, and M Ramkala. "Role of Genetic Algorithm in Routing for Large Network." *International Journal on Computer Science and Engineering (IJCSE)*, vol. 6, no. 1, Jan. 2014

[7]Nagib, Gihan, and Wahied G Ali. "Network Routing Protocol Using Genetic Algorithms." *International Journal of Electrical & Computer Sciences*, vol. 10, no. 02, Apr. 2010

[8]"Genetic Algorithms." *GeeksforGeeks*, 23 Aug. 2018, www.geeksforgeeks.org/genetic-algorithms/.

[9] C. W. Ahn and R. S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," Evolutionary Computation, IEEE Transactions on, vol. 6, no. 6, pp. 566–579, 2002.

[10] N. Banerjee and S. K. Das, “Fast determination of qos-based multicast routes in wireless networks using genetic algorithm,” in Communications, 2001. ICC 2001. IEEE International Conference on, vol. 8, pp. 2588–2592, IEEE, 2001.

[11] S. Yang, H. Cheng, and F. Wang, “Genetic algorithms with immigrants and memory schemes for dynamic shortest path routing problems in mobile ad hoc networks,” Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 40, no. 1, pp. 52–63, 2010.

[12] F. C. NK and S. D. K. Viswanatha, “Routing algorithm using mobile agents and genetic algorithm,”