Optimal Path Selection in Ad Hoc (MANET) by Using Genetic Fuzzy Petri Net

Hussein Attya Lafta [1], Fadhil Mohammad Salman [2]

[1.] Assistant Professor Doctor, En, Iraq, University of Babylon, Iraq 2014.
[2.] Researcher, University of Babylon-College of Sciences, Iraq 2014.
hzazmk@yahoo.com, fadhilmohammad2000@yahoo.com

Abstract: A mobile Ad-hoc network (MANET) is a dynamic multi hop infrastructure less wireless network established by a group of mobility nodes in which there is no central administration. Due to the absence of central infrastructure and mobility of nodes that led to dynamic network topology, therefore, the routing is one of the most important challenges in ad-hoc networks. This challenge led to the development of many techniques which deal with it, one of the most important techniques it's a fuzzy logic. This paper focuses on finding the optimal path which using to transfer data, so two techniques ware used in this work. The first one is: The network simulator version 2 (NS-2) was used to build network system for study and evaluate the behavior of two routing protocols (AODV and AOMDV), in order to prove that path selected from this routing protocol is not optimal path, the performance metrics that used to evaluate the routing protocols are: the throughput, the dropped packets, average end-to-end delay, normalizes routing load, packet delivery fraction, and jitter. Therefore, we proposed model depended on second techniques Fuzzy Petri Net model, this model was proposed in order to mimic the route discovery mechanism in routing protocols with using a new parameters like (Number of hops, Local Battery level, Received Signal Strength Indicator) as the input to fuzzy inference system. Genetic algorithm has active role in the generating of fuzzy rules, where it used in this work to get the best number of generated rules, depended on training stage to generate fuzzy rules in offline, this fuzzy rules is very important in the reasoning processes to making a final decision about what is path most select.

[Hussein Attya Lafta, Fadhil Mohammad Salman. **Optimal Path Selection in Ad Hoc (MANET) by Using Genetic Fuzzy Petri Net.** *Researcher* 2014;6(8):31-44]. (ISSN: 1553-9865). http://www.sciencepub.net/researcher. 5

Key words: MANET, AODV, AOMDV, NS-2, GA, FL, FIS, PN, FPNs

1. Introduction

Wireless networks can be defined as a network in which the nodes are interconnected by wireless channels, due to increasing development of wireless network, there are two architectures existing for these networks, the first one is known as infrastructure networks and the second one is known as infrastructure less networks. An ad hoc network is a collection of wireless mobile hosts forming a temporary network without needing of any established infrastructure or centralized administration. In such an environment, it may be necessary for one mobile host (node) to enlist the aid of other hosts (nodes) in forwarding a packet from a source node to its destination, Due to the limited range of each mobile host's wireless transmissions. So, MANET's nodes are act as a both host and router by receiving and forwarding data packets [12]. If these nodes change their positions dynamically, it is called a mobile ad hoc network (MANET). The MANET is self-organizing and self-configuring multi hop wireless networks where the structure of the network changes dynamically because of mobility of nodes, the nodes in a MANET are free to move and organize themselves in the arbiter fashion. Wireless mobile ad hoc network (MANET) technology is designed for the establishment of a network anywhere and anytime, without any fixed infrastructure to support the mobility of the users in the network. A MANET can be a standalone network or it can be connected to external networks (Internet). The main two characteristics of MANET are nodes mobility and multi hop, hence multi hop operation requires a routing mechanism designed for mobile nodes. In a MANET networks where there is no infrastructure support and since a destination node might be out of range for source node that transmitting packets, therefore, efficient route is needs to exchange information between different nodes, it's done by using different routing protocols. So, efficient routing procedures is always needed to find a path between nodes, in order to obtain appropriate transfer of the packets between the source node and the destination node [2]. Therefore, a central challenge in the design of ad hoc networks is the development of dynamic routing protocols that can efficiently find and maintains the routes between two communicating nodes (source & destination). This paper focuses on the routing process in order to selected the optimal path between any two nodes, that path is reduce some challenges that faced the MANET such as (end-to-end delay, loss packets, energy consuming, etc) and increasing the performance of network such as (throughput, packets delivery, quality of service, etc). The nodes often have a finite capacity path cache, it may not be possible to store all paths, therefore, always needed to find the optimal path communication between sender nodes to destination node for mobile ad hoc networks. Organization of the paper: Section I provides introduction to wireless ad-hc network. Section II provides overview about some

related work. Section III provides introduction to NS-2. In section IV provides an overview of different theory's that used this work. Section V provides a discussion on the simulation which used NS-2 and the proposed FPN model finally section VI provides the conclusion on the proposed scheme, VII Future Works.

II. Related Works

- ❖ Tzu-Chiang Chiang and et al. at 2009 proposed a knowledge-based inference approach to the new path discovery for multicasting. A fuzzy Petri net agent, which is a special expert system was introduced at each node to learn and to adjust itself to fit the dynamic conditions in a multicast ad-hoc network. the simulation results show that the proposed approach is up to 67.17% more efficient in the packet delivery ratio as compared with a bandwidth effective multicast routing protocol [17].
- ❖ A novel reliable routing algorithm in mobile ad-hoc networks using fuzzy Petri net with its reasoning mechanism was proposed by Zhi-gang Hu, et al. at 2005 to increase the reliability during the routing selection process. which has a fuzzy reasoning mechanism for finding the routing sprouting tree from the source to the destination node in the mobile ad hoc environment, by compared the degree of reliability in the routing sprouting tree, so, the most reliable route can be computed. The results show that the routing reliability was increased by more than 80% then applying the proposed algorithm to the ad hoc on demand distance vector routing protocol [21].
- ❖ The performance of hybrid routing protocol for the MANET was analyzed by Subramanyam, P.V, et al. at 2006, when they used Fuzzy Petri nets model which calculate the level of firing transitions among the nodes and comparing it with a threshold value for the nodes of the network. The results indicated enhanced performance with dynamically viable node movement [14].
- ❖ Tzu-Chiang Chiang and et al. at 2004 described the multicast routing representation using fuzzy Petri net model with the concept of immediately reachable set in wireless ad hoc (MANET) networks which all nodes equipped with GPS unit. A heuristic algorithm is used to compute the multicast tree based on the local network topology with a multicast source. The results shows that the percentage of the improvement is more than 15% when compared the IRS method with the original method [16].

III. Introduction to NS-2

NS-2 simulator is an open source simulator; it is very useful and important for founding and investigating variety of protocols. It use a very large numbers of applications, protocols, network types, network elements, mobility models and traffic models to investigation realistic simulation. At the simulation layer NS-2 uses OTcl (Object oriented Tool Command Language) programming language to interpret user simulation scripts. OTcl language is in the fact an object oriented extension of the Tcl Language. The Tcl language is fully compatible with the C++ programming language. At the top layer, NS-2 is an interpreter of Tcl scripts of the users, they work together with C++ codes [6]. An OTcl script written by a user is interpreted by NS. While OTcl script is being interpreted, NS creates two main analysis reports simultaneously as a files. One of them was called the NAM (Network Animator) object that shows the visual animation of the simulation. The other was called the trace object that consists of the behavior of all objects and all the event occur in the simulation, both of them are created as a file by NS-2. Former is .nam file used by NAM software that comes along with NS. Latter is a ".tr" file that includes all simulation traces in the text format. NS-2 project is normally distributed along with various packages (ns, nam, tcl, otcl , etc.) named as "all-in-one package", but they can also be found and downloaded separately [11]. That means After simulation NS-2 output are Trace file and Nam trace file as a simulation results. To interpret these results graphically and interactively, tools such as Network Animator (NAM) are used. NAM is an animation tool for viewing network simulation traces and packet traces, and XGraph is used as a data plotter to analyze a particular behavior of the network.

IV. Theory of The Work

A- Genetic Algorithms (GAs)

Genetic Algorithms (GAs) are general-purpose search algorithms that represent evolutionary optimization approach, proposed by John Holland in 1970. They are particularly applicable to problems which are large, non-linear & possibly discrete in nature. GA try to work on principle of natural selection, as in natural selection over the time individuals with "good" genes survive whereas "bad" ones are rejected [1]. As the natural evaluation has the following feature [13]:-

- ❖ The individual characteristics are encoded as a gene on the chromosome.
- ❖ Compute the fitness function value to each chromosome according to the environment in which it

exists.

- ❖ Individual chromosomes has strong behaviors are able to survive and produce next
- ❖ generations of strong individual chromosomes.
- ❖ genetic operators can be generate a new variants during reproduction The individuals by its operation called (mutation).

In Genetic Algorithm the solution of the problem is encoded on a string of bits that represented as a gene comparable with the chromosome of the biological system analogy. The Genetic Algorithm keeps a population of randomly selected chromosomes to combine by mutation or crossover techniques and produce the offspring having new characteristics, which in turn replaces the low fitness old chromosomes. This process is repeated until we find a chromosome with best fitness and repeated characteristics for the successive generations of the population. Which finally represent the optimal solution for the problem [10].

B- Fuzzy Petri Nets

The classic Petri nets was used to modeling system that accurately describe is unsuitable in the systems of uncertain (imprecise) and ambiguous information, While fuzzy logic is dealing with uncertain and ambiguous data. According to the uncertain information is to be display with fuzzy logic, therefore to be useful integration theory of fuzzy in petri nets to increase the power of modeling. In 1988, was performed this work by Looney and several authors of Petri Net, where they are combine the collection of artificial intelligence due to its adequacy to represent the reasoning process as a dynamic discrete event system with petri net and design types of petri nets that are more or less compatible with the theory of Petri Net but more powerful. The techniques that are combine the Petri Nets and fuzzy sets, called Fuzzy Petri Nets (FPNs), that use for knowledge representation and as a control in the fuzzy management systems [19]. The formalism of Petri nets can be used to model fuzzy-rule based systems by simply identifying some elements (places and transitions) and features (marking function) of Petri-net's formalism with the basic elements of a fuzzy-rule based such as knowledge base (KB), propositions (rules), degree of truth of the rules and implication relationships, where transitions serve as rules, places serve as propositions, and markings are assigned fuzzy values between 0 and 1. [8]. FPNs is an application specific Petri Nets based approach developed to represent uncertain operations and approximate conditions in areas such as robotics, traffic control, communication, medical diagnosis, flexible manufacturing systems and fuzzy controllers. FPN model was introduced for the specification of rule-based reasoning using propositional logic. Places are interpreted as conditions having fuzzy truth values (tokens), while transitions represent the fuzzy decision values of rules, The relationships from places to transitions and vice versa are represented by directed arcs. Reasoning in the fuzzy petri net can be performed by iteratively maxing (generalized OR) and mining (generalized AND) transitions and fuzzy truth values of tokens, respectively [4]. Because normal PN cannot deal with vague or fuzzy information such as "very high" and "good", several Fuzzy Petri Nets (FPN) have been introduced. As a model of knowledge-based systems, fuzzy Petri net model for expert systems is called Adaptive Fuzzy Petri Nets (AFPN), This model has both the benefits of a fuzzy Petri net and the learning ability of a neural network to have the ability of learning like neural networks, where each transition serves as neurons, therefore it can be learned as well as it can be train to adapt the change situation. Accordingly an AFPN model can be used for dynamic knowledge representation and inference [18]. The firing rule of transition's in the fuzzy petri net , don't remove the token from the input place after transition firing, therefore, FPNs unbounded places because token will be unlimited in places [7]. The FPNs was expanded from a petri net is a bipartite graph that has place and transition nodes like the petri nets. But, in FPNs a token incorporated with a place is associated with a real value between 0 and 1, so, the transition is associated with a certain factor (CF) real value between 0 and 1. FPNs is a promising modeling tool for expert systems and it is suitable for fuzzy knowledge representation and reasoning. A generalized fuzzy petri nets is defined as a 10-tuple, table (1) [20]:

Table (1): Format Definition of Fuzzy Petri Nets.

FPN = (P, T, I, O, D, W, μ, f, α, β)	
Symbols	Description
P	{p1, p2, . . , pn} denotes a finite nonempty set of places.
T	{t1, t2, . . , tm} denotes a finite nonempty set of transitions.
I	P → T was the input function, a mapping from places to transitions, input incidence matrix.
O	T → P was the output function, a mapping from transitions to bags of places.
D	{d1, d2, . . , dn} denotes a finite set of propositions, that interprets fuzzy linguistic, P ∩ T ∩ P, |P| = |D|.
W	WI U WO, is a finite set of input and output weights of nets.
μ	T → [0, 1] denotes the certainty factors of fuzzy rules.
F	T → [0, 1] denotes the threshold of a transition firing.
α	P → [0, 1] is an association function which maps from places to real values between zero and one .
β	P → D is also an association function mapping from places to propositions.

That means, each place in FPNs may or may not contain token associated with a truth value between 0 and 1, and the token in the places represent the degree of belong to membership of this real value. The transition has a certainty factor associated between 0 and 1 and it represents the inference engine.

V. A- Simulation Using NS-2

In this paper, the NS-2 with two routing protocols (Ad-hoc On Demand Distance Vector (AODV) and Ad-hoc On Demand Multipath Distance Vector (AOMDV)) are used in the simulation, in the different scenarios, the main reasons to select the AODV routing protocol is that it was represented as a single path routing protocol, and selected the AOMDV routing protocol because it was represented as a multipath routing protocol. Figure (1) explains the main stages of simulation used in this work, that implemented by using NS-2, where the code was written in a Tcl language and the outputs of the NS-2 are trace file and NAM file. The trace that is used in this work is old trace file because it contains all the fields that are required in the computation of the performance metrics. This performance metrics was computed according to several steps that programmed using AWK programming language:

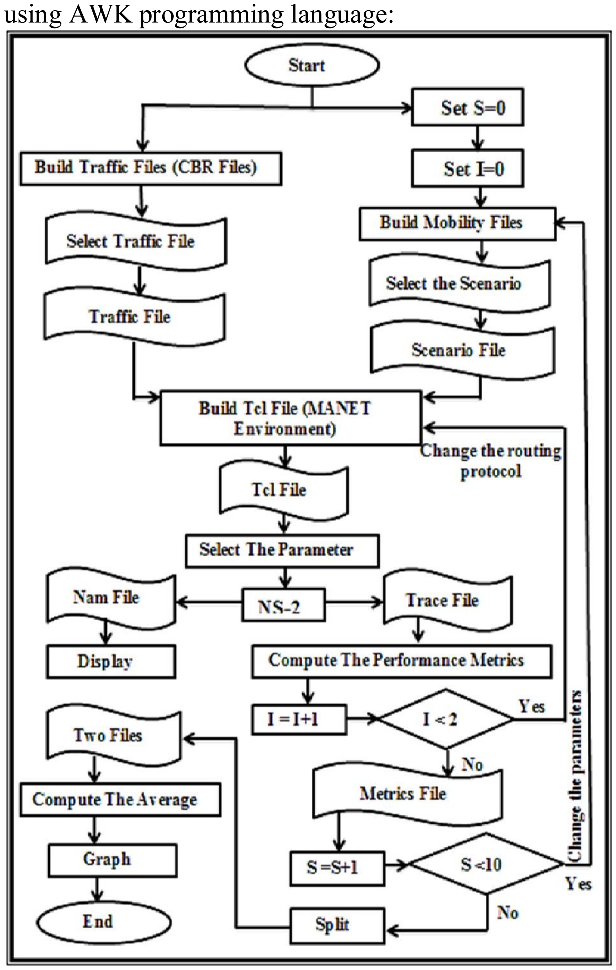


Figure (1): Flowchart for Stages of Simulation.

Simulation Algorithm

Step1:- Start.

Step2:- Set S=0 (S represent the number of scenario file (movement file)).

Step3:- Build the traffic generation file "CBR file" that is generated by "cbrgen.tcl" file that supported by NS-2. This script is found in (ns-allinone-2.34/ns-2.34/ind_util/cmu _ scen_gen/).

Step4:- Set I=0 (I represents the number of routing protocols that are used in this work).

Step5:-Build MANET's scenario (movement file) using support of NS-2 by the "setdest" script.

Step6:- Build "tcl" script that represents simulation environment of MANET with mobility model for one routing protocols.

Step7:- Select suitable parameters that input to this "tcl" file in the NS-2 in order to perform the simulation, and the outputs are "NAM" file to display and trace file contain all the simulation event to analysis.

Step8:- Analyze the trace file and compute the performance metrics for the network (throughput, drop packet, end to end delay, jitter, packets delivery and normalize routing load).

Step9:- Increment I by 1.

Step10:- If (I< 2) then go to step6 (to implemented another routing Protocol) and save the new results in metrics file, Otherwise, S=S+1.

Step11:- If (S<10) then go to step5 (S is the number of MANET scenarios). Otherwise, go to the step12.

Step12:- Split the resulted file in to two files (each one contains the results of one routing protocol).

Step13:- Compute the average of the performance criterion for each routing protocols file and put it in final file.

Step14:- possibility draw the results with the suitable parameter by using Xgraph this script is supported by NS-2 or draw it as a histograms.

Step15:- End.

Simulation Environment

A discrete event Network Simulator NS-2 version 2.34 was used in this work to simulate the mobile ad-hoc network, depending on the "Tcl" programming language that is used to build the simulation environment of the mobile ad-hoc network. There are several performance metrics that can be used to analysis the performance of mobile ad-hoc network or the protocols in order to understand it performance as following:-

1. The Throughput

It is the amount of digital data transmitted per unit time from the source node to the destination node, it is usually measured in bits per sec [9].

2. Packet Delivery Fraction (PDF)

It is the ratio between the number of packets originated by the "application layer" CBR sources and the number of packets received by the CBR sink at the final destination [9].

3. Dropped Packets

It is the number of packets that sent by the source node and unsuccessful to reach to the destination node [3].

4. Normalize Routing Load (NRL)

Is the total number of control packets (include RREQ, RREP, RERR and REP_ACK packets) divided by number of transmitted data packet in the network [15].

5. Average End-To-End Delay

Is the average time taken by data packets when released by sources until reach to their destinations [15].

6. Average Jitter

It is the absolute value of the difference between the end-to-end delays of two sequential packets, The average jitter is obtained by summing the jitter of all received packets divided by the total number of the received packets [5].

Table (2): Network Parameters Used During The Simulation.

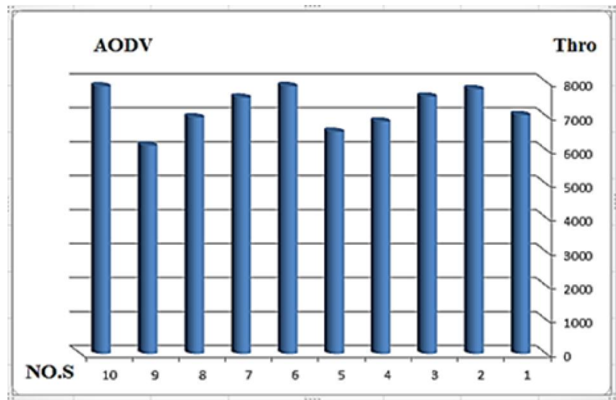
Parameter	Value
The simulator	NS-2 version 2.34
MAC	802.11
Propagation model	Two ray ground
Routing protocols	AODV & AOMDV
Simulation time	75s
Traffic generator	CBR
Antenna	Omni Antenna
Packets size	512 bytes/packet
Transition rate	2.0 packets/second
Mobility model	Random waypoint model
Pause time type	Uniform
Speed type	Uniform

Table (3) Shows the suggested parameters used to build simulation scenario that input to "Tcl" script, as follows:-

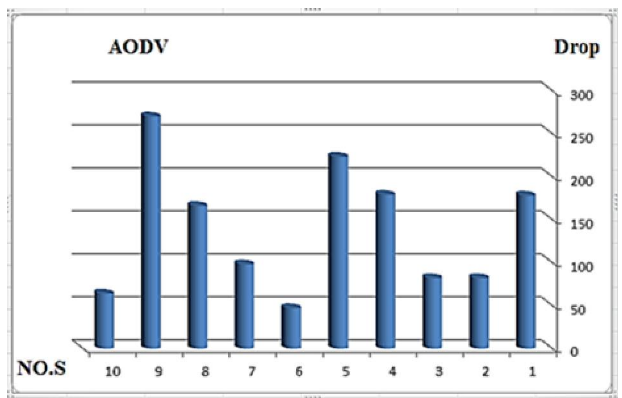
Table (3): Parameters Used During Create Scenario.

Parameter	Value
Node number	30
Pause time	8.00s
Max node speed	20.00 m/s
Area	1000m*1000m

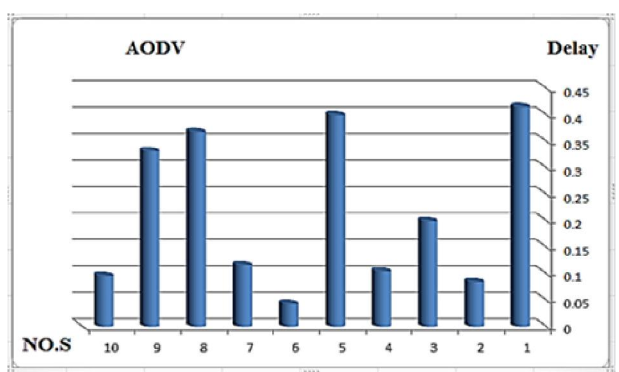
The number of results obtained after applying this collection of parameter according to suggested steps explained previously, figure (2) describes the results of simulation with AODV routing protocol for 10 scenarios as histograms:-



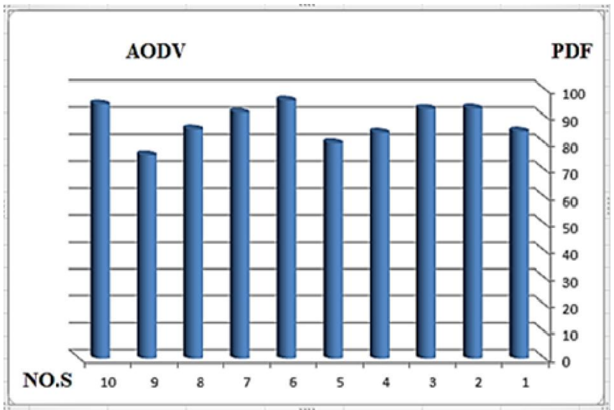
a . Histogram of throughput.



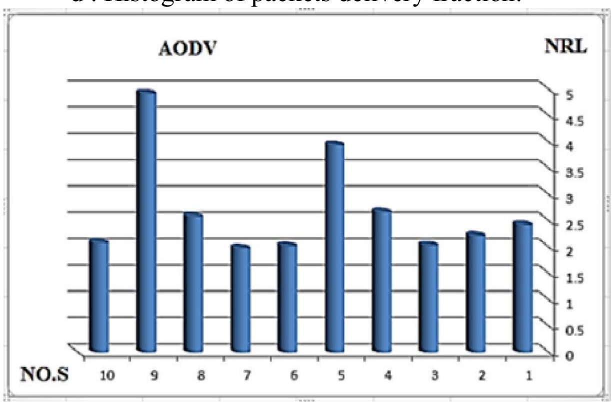
b . Histogram of drop packet



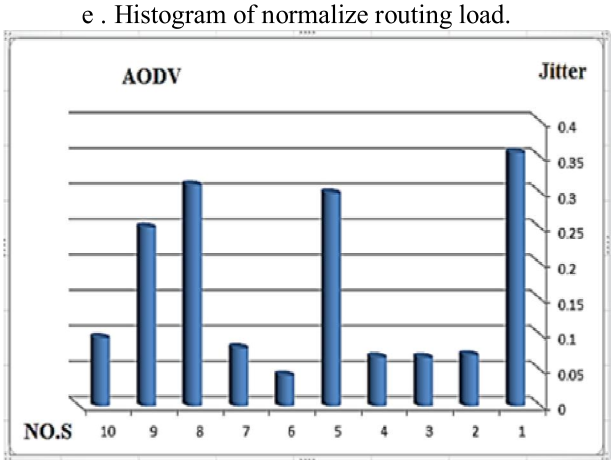
c . Histogram of end to end delay.



d . Histogram of packets delivery fraction.



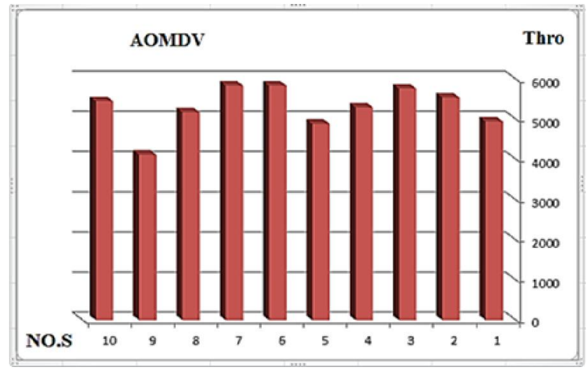
e . Histogram of normalize routing load.



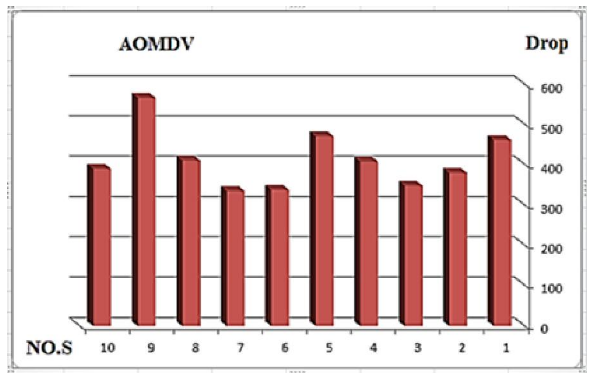
f . Histogram of jitter.

Figure (2): Histograms of AODV Results.

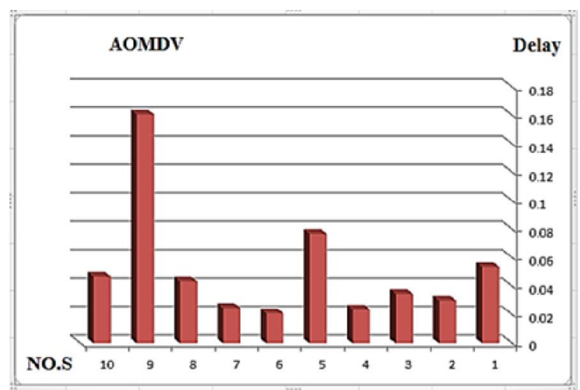
The following figure (3) describes the results of simulation with AOMDV routing protocol for 10 scenarios as histograms:-



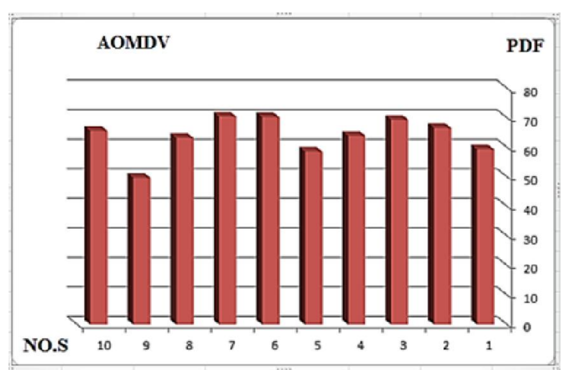
a . Histogram of throughput.



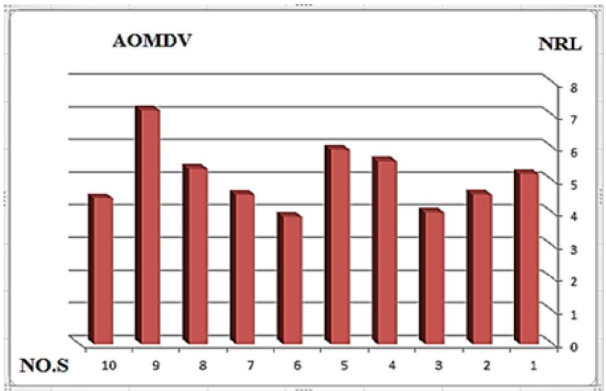
b . Histogram of drop packet.



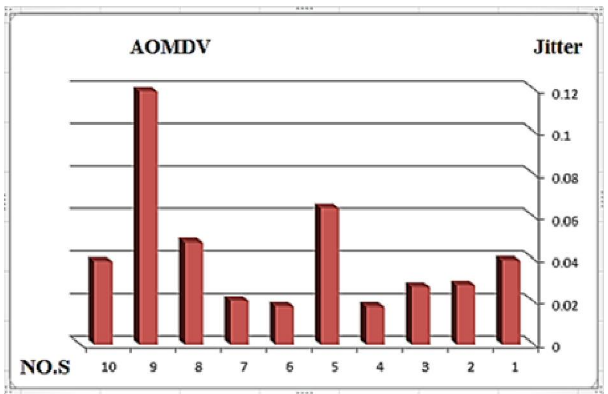
c . Histogram of end to end delay.



d . Histogram of packets delivery fraction.



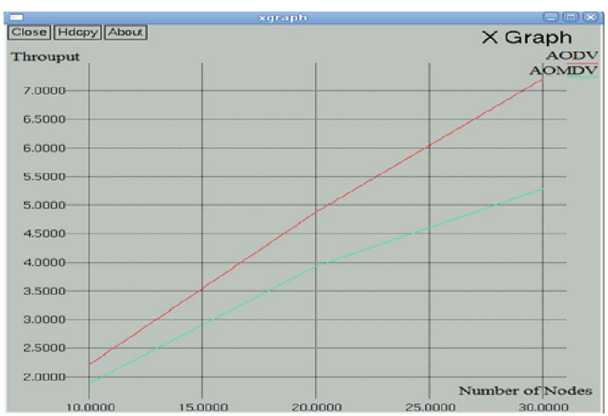
e . Histogram of normalize routing load.



f . Histogram of jitter.

Figure (3): Histograms of AOMDV Results.

The following figures describes the simulation results for AODV & AOMDV routing protocol with (10,20,30) nodes, depended on Xgraph script which supported by NS-2 simulator:-



Figure(4): Comparison Throughput for AODV & AOMDV with versus number of nodes (10 , 20 , 30).

Figure (5): Comparison Dropped Packets for AODV &AOMDV with versus number of nodes (10, 20, 30).

Figure (6): Comparison Average End-To-End Delay for AODV&AOMDV with versus number of nodes (10, 20, 30).

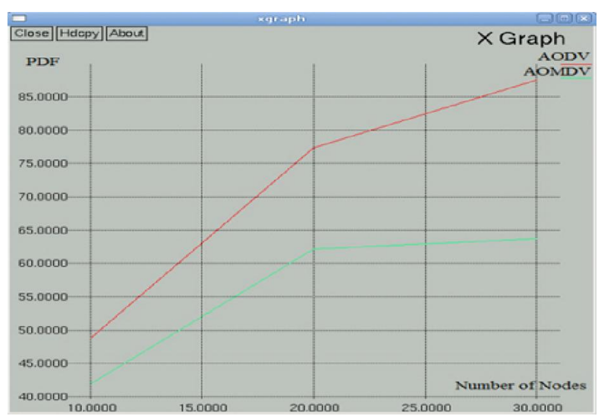


Figure (7): Comparison Packet Delivery Fraction (PDF) for AODV&AOMDV with versus number of nodes (10, 20, 30).

Figure (8): Comparison Normalize Routing Load (NRL) for AODV&AOMDV with versus number of nodes (10, 20, 30).

Figure (9): Comparison Average Jitter for AODV & AOMDV with versus number of nodes (10, 20, 30).

The obvious differences in previous results with (AODV & AOMDV) routing protocol for 10 scenarios, that led to conclusion, that path used from this routing protocols to transfer data is not optimal path because it was selected with regardless to some of important issues that influence on the behaviors of network such as (limited battery power for nodes, congestion in the network, number of hops, signal power, …, etc). In order to select the best nodes to be part of the routes, a Genetic Fuzzy Petri Net is proposed to mimic the route discovery process, where, depending on this route metrics, the decision of path selection is done.

V. B- The Proposed GFPN System.

In this work the GFPN model is developed to simulate the route discovery process in ad-hoc routing protocols depending on the inference process from antecedent to the consequent propositions, which use reasoning process of the two techniques (Fuzzy set and Petri nets).

Figure (10) explains the main stages of proposed FPN system:-

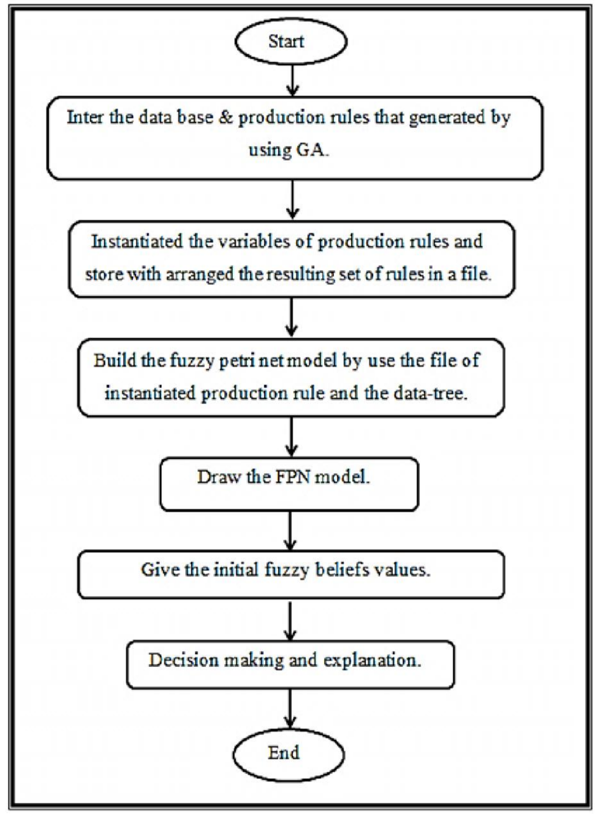


Figure (10): Flowchart for Modeling of The Proposed System.

V. B-1. The Data Base

The data base used in this work is organized in the form of data-tree having a depth of three levels. The root of tree be in the first level (represents starting pointer), predicates be in the second level and fuzzy beliefs that must correspond to each predicate (second level) are in the third level. This organization helps in efficient searching of the data base. The fuzzy beliefs must be collect from sources with good authentication level. Each belief must have a certainty factor. For more explain show the figure (11).

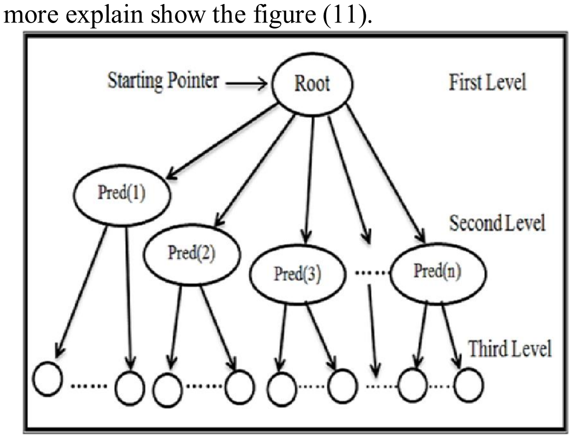


Figure (11): Tree of The Data Base.

The following algorithm is used for create data-tree, figure (12) shows this algorithm.

Input :- Data file (file containing the data base).

Output :- Tree of data base.

Process :-

1. Begin
2. Set (i ⟵ 0), where (i) is file pointer.
3. While not end of the data base file do,
4. Create a root node of data base tree,
5. Open the file containing the data base for reading,
6. From the data base file read a predicate,
7. i ⟵ i +1,
8. If the predicate is found at the second level of the data-tree, then mark the node, else create a new node corresponding to that predicate,
9. Search the beliefs of that predicate in the third level of the data-tree,
10. If the belief is not found, then add it in the third level as a child of the marked predicate,
11. End while,
12. Close the data file,
13. End.

Figure (12): Algorithm for Creating Data Tree.

V. B-2. Production Rules

The rules for decision making are created by using the genetic algorithm. GA generate the rules by using the Michigan approach, where each individual in population represents as a rule with its inputs and outputs. So, all the rules are evaluated off-line before being applied in the proposed system. The evaluation algorithm assigns strength to every classifier with a non-zero degree of activation. Figure (13) shows the training steps of genetic algorithm as following:-

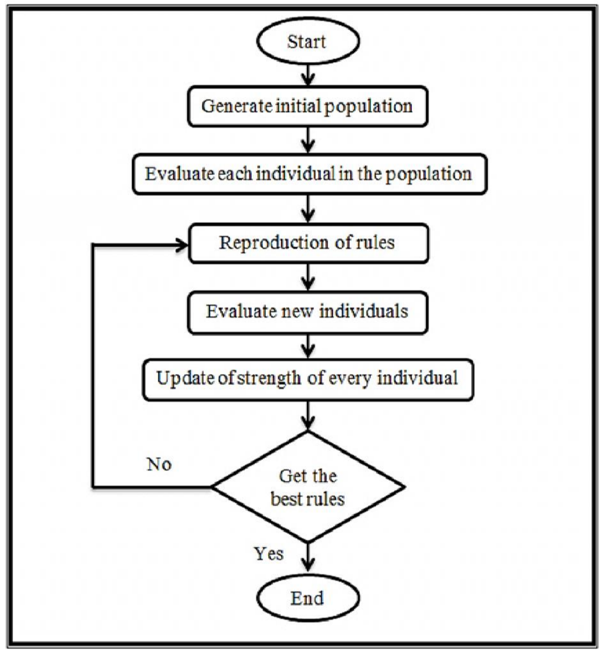


Figure (13): Flowchart for Production Rules Algorithm.

Production Rules Algorithm

Step1:- Start.

Step2:- Create random initial population, where eachindividual in this population represents rulewith its inputs and outputs and initial strength0.5. Every input and output represents gene,hence, the length of chromosome is 4 gene(according to the fuzzy input parameters:Number of hops, Local Battery level,Received Signal Strength Indicator) with theoutput.

Step3:- The individuals (rules) in the population areevaluation by getting membership degree ofevery gene in the chromosome and then hasstrength of rule to represent fitness of eachrule. Where the active rule according to itsfitness is used in crossover to get a newindividual, this new individual is replace byworst individual.

Step4:- The binary tournament selection is used forselecting the best individual from thepopulation and uniform crossover is used toproduce new individual from the parent (activerules). So, the soft mutation modifies one of thecharacteristic points (a, b, c) of the trianglefuzzy set.

Step5:- In this stage every new individual is evaluatedby checking if it is active to get the strength ofrule.

Step6:- The strength of rules is update with every newgeneration, depending on the system'sperformance and rule adaption rate, in order touse it to compute the fitness of each individualto every new generation.

Step7:- The suitable rules according to the specificmeasures have got go to the step8, Otherwise,go to the step4.
Step8:- End.

The following table (5) shows the set of rules that result from training phase of GA with use of the following abbreviation, as in table (4).

Table (4): Abbreviations That Used in The Production Rules Process.

Symbol	Description
N	Number of hops
B	Local battery level
S	Received signal strength indicator
O	Optimal path
L	Low
M	Medium
H	High

Table (5): List of Generated Rules.

NO	Rule
1	"if n is h and b is l and s is h then o is m "
2	"if n is l and b is m and s is m then o is m "
3	"if n is h and b is m and s is h then o is m "
4	"if n is m and b is l and s is l then o is l "
5	"if n is m and b is m and s is l then o is l "
6	"if n is m and b is l and s is h then o is m "
7	"if n is h and b is h and s is l then o is l "
8	"if n is m and b is h and s is m then o is m "
9	"if n is l and b is l and s is l then o is l "
10	"if n is h and b is h and s is l then o is l "
11	"if n is h and b is m and s is l then o is l "
12	"if n is l and b is m and s is l then o is m "
13	"if n is l and b is l and s is m then o is l "
14	"if n is l and b is l and s is h then o is m "
15	"if n is m and b is h and s is l then o is l "
16	"if n is m and b is m and s is m then o is m "
17	"if n is l and b is m and s is h then o is h "
18	"if n is l and b is h and s is l then o is m "
19	"if n is h and b is h and s is h then o is m "
20	"if n is m and b is l and s is m then o is l "
21	"if n is m and b is h and s is h then o is h "
22	"if n is h and b is h and s is m then o is m "
23	"if n is h and b is m and s is m then o is l "
24	"if n is m and b is m and s is h then o is m "
25	"if n is l and b is h and s is m then o is h "
26	"if n is h and b is l and s is m then o is l "
27	"if n is l and b is h and s is h then o is h "

V. B-3. Fuzzy Petri Net Model Building

The FPNs model is created depending on arranged fuzzy rules and data base, after the last was represented as a tree. The algorithm which used to build the FPN model is explained in figure (14) as follows:-

Input :- List of production rules and tree of data base.
Output :- Fuzzy Petri Net Model.

Process :-

1. Begin
2. Set (r $\longleftarrow$ PRs), where PRs is list of production rules,
3. While ($r \neq 0$) do
4. For (i $\longleftarrow$ 1 to i $\longleftarrow$ r),
5. Call the tree of data base and PRi,
6. If all the antecedent clauses of PRi are exist in the data base tree then do
7. Store the antecedent-consequent pairs of PRi into the WM (WM is working memory), for using it to create FPN model,
8. Search the antecedent stored in WM,
9. If it is found then mark those places else create new place and marked it, create one transition and the consequent marked it and establish connectivity from the antecedents to the transition and from the transition to the consequent places,
10. Add the consequence to the data base if not found,
11. Eliminate PRi from the list of PRs,
12. r $\longleftarrow$ r − 1,
13. End while,
14. Mapping the propositions, memberships of the fuzzy production rules to the places and transitions of the model,
15. End.

Figure (14): Algorithm for Building a FPN Model.

The following interface helps the user to create the FPN model with aid the given database and production rules. This interface consists of several interactive sub interfaces like (Create FPN, Draw fuzzy Petri net, Input fuzzy initial values, and Decision making with explanation). Figure (15) shows these interface.

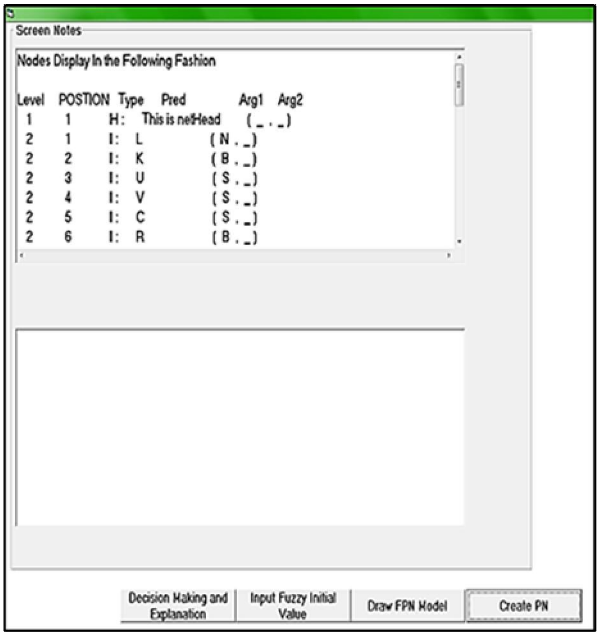


Figure (15): The User Interface for Creating FPN Model.

Figure (16) shows the FPN model of proposed system:-

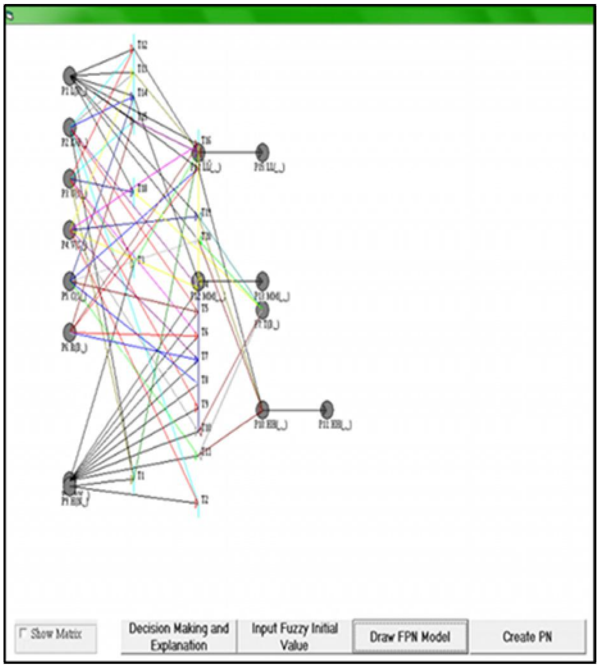


Figure (16): The Fuzzy Petri Net Model.

V. B-4. FPN Inputs

The input to the proposed system is three parameters (number of hops, Local Battery level, Received Signal Strength Indicator), that Fuzzified by using triangular membership function, where its input values is converted to fuzzy values according to membership function that related to each input parameters as follows:-

A. <u>Number of Hops (N):-</u> This is the length of the path. This input variable is divided to three fuzzy sets as shown in table (6), and the membership function is shown in figure (17).

Table (6): Fuzzy Sets for The Number of Hops (N).

Input Range to The Number of Hops (N)	Fuzzy Set
0 – 10	Low (L)
10 – 30	Medium (M)
20 - 40	High (H)

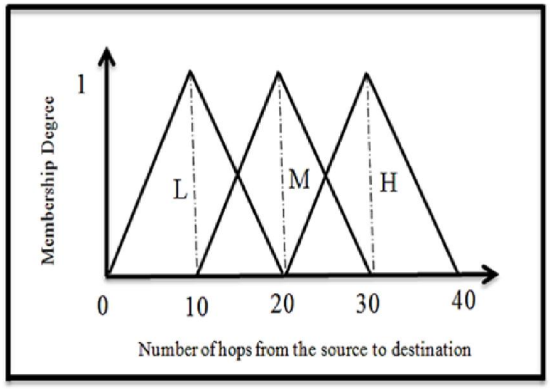


Figure (17): Membership Function for Number of Hops

B. <u>Local Battery Level (B):-</u> This represented the battery level of the node, this input variable is divided to three fuzzy set as shown in table (7), and the membership function is shown in figure (18).

Table (7): Fuzzy Sets of Local Battery Level (B).

Input Range of Local Battery Level (B)	Fuzzy Set
0 – 50	Low (K)
25 – 75	Medium (R)
50 – 100 %	High (T)

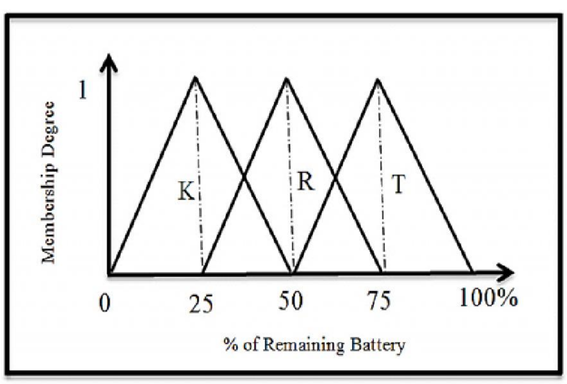


Figure (18): Membership Function of Local Battery Level (B).

C. Received Signal Strength Indicator (S):- This represented the strength of the received signal is an indicator of the quality of communications between two nodes. This input variable is divided to three fuzzy set as shown in table (8), and the membership function is shown in figure (19).

Table (8): Fuzzy Sets of Received Signal Strength Indicator (S).

Input Range of Received Signal Strength Indicator (S)	Fuzzy Set
0 – 50	Low (U)
25 – 75	Medium (V)
50 – 100 %	High (C)

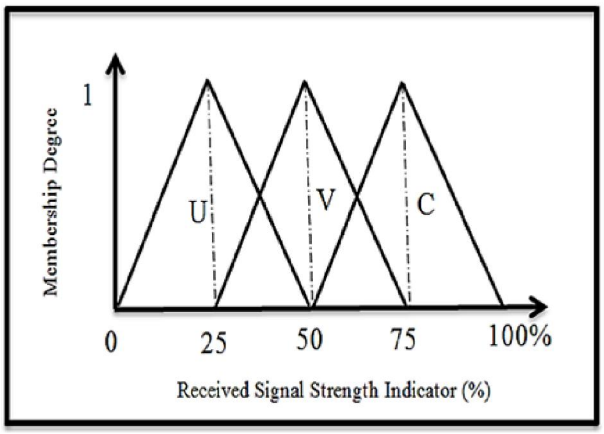


Figure (19): Membership Function of Signal Strength Indicator.

D. Optimal Path:- This is the output of fuzzy system that represents the suitability of a node to be considered for inclusion in the route. This output is divided to three fuzzy sets as shown in table (9), and the membership function is shown in figure (20).

Table (9): Fuzzy Sets for The Optimal Path.

Input Range for The Optimal Path	Fuzzy Set
0 – 0.5	Low (LL)
0.25 – 0.75	Medium (MM)
0.5 – 1	High (HH)

The approach of fuzzification is used to obtain the fuzzy membership degrees for each input arguments (number of hops, Local Battery level, Received Signal Strength Indicator), as mentioned in chapter four, by using triangular membership function. Defuzzification is the final phase in any fuzzy system, as explained in chapter four. According to this phase, the decision making takes place about which path is selected to be output of proposed system (optimal path), with some conclusions such as (Path number, Path nodes, Path gain value), after input random fuzzy initial values. In summary, the path with the higher gain is selected and the information on this path is then used for providing an explanation to the user, as shown in the following figure (21).

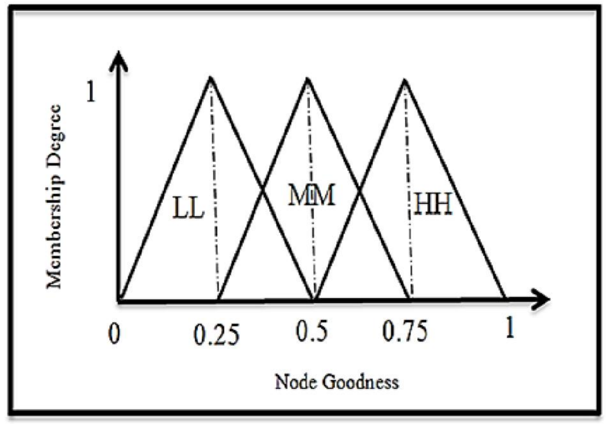


Figure (20): Membership Function of Optimal Path.

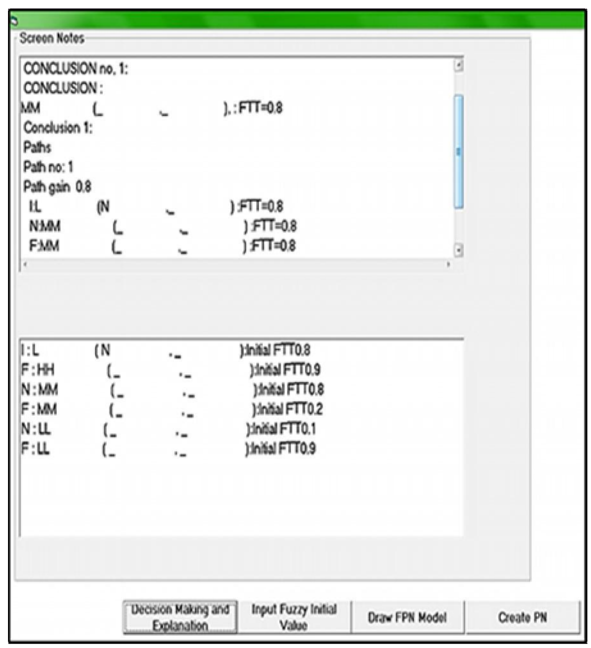


Figure (21): Decision Making and Explanation.

VI. Conclusions

1. The routing is one of the most important challenges in ad-hoc networks due to absence of central administration and mobility of nodes, therefore, many techniques ware development to deal with it like fuzzy logic.

2. The network simulator version 2 (NS-2) was used to build a system for studying and evaluating the behavior of two routing protocols (AODV and

AOMDV). From the difference in results obtained after running the simulation for 10 iterations, we conclude that the path selected from this routing protocol is not the optimal path.

3. The lack of an efficient metric to evaluate node conditions in routing protocols, has been solved by the definition of a new efficient metric based on the combination of different node and new network parameters by using a Genetic Fuzzy Petri Net system.

4. The Genetic Fuzzy Petri Net system is used to select the optimal path, depending on three fuzzy input parameters (number of hops, Local Battery level, Received Signal Strength Indicator) and a set of fuzzy rules which are created from training phase of genetic algorithm. This model is used in order to mimic the route discovery mechanism in ad-hoc routing protocols to select the optimal path.

5. Genetic Fuzzy Petri Net system combines the advantage of both fuzzy logic and petri nets, to create useful system that deals with uncertain information and it can provide a visible modeling.

6. Genetic algorithm has an effective role for generating set of fuzzy rules by training stage and selecting the best between them.

7. When the data base is represented as a tree structure, it make the searching operation more powerful and faster.

8. Using the fuzzy logic as a metric in network routing improves the performance of real networks.

9. Using the fuzzy logic as a metric in network routing improves the performance of the intelligent dense monitoring of the network, thus, led to increasing quality of service.

10. The fuzzy logic is a useful and powerful approach that has demonstrated to be effective when combined with other disciplines such as routing approaches for wireless ad-hoc (MANET) networks.

VII. Future Works

This work can be extended in different directions, in the following some suggested ideas are given:-

1. Future research can be done by the addition of new parameters as input to the fuzzy inference system and studying the performance achieved by these new parameters, such as the bandwidth, the load of the link, the traffic load, the power consumption, the total vector cost, the time to life for the path and the node density.

2. In the future research one can use the fuzzy logic in other ad-hoc layers, such as the MAC layer, that will help to provide a priority in the contention period for candidates nodes with better conditions.

3. Depending on the proposed Genetic Fuzzy Petri Net model that is explained in this thesis, the optimal path can be selected. Therefore, a future research can programming the approach of this model in C++ language, in order to combine the result code with routing protocol code in the part that allocated for route discovery process to obtain an efficient routing protocol.

References

1. Anjum Mohammed, Gihan Nagib, "Optimal Routing In Ad-Hoc Network Using Genetic Algorithm", Int. J. Advanced Networking and Applications, Volume: 03, Issue: 05, Pages: 1323-1328, 2012.
2. Akshatha. P. S, Namita Khurana, Anju Rathi, "OPTIMAL PATH FOR MOBILE AD-HOC NETWORKS USING REACTIVE ROUTING PROTOCOL", International Journal of Advances in Engineering & Technology, Sept, 2011.
3. Aliff Umair Salleh, Zulkifli Ishak, Norashidah Md. Din, and Md Zaini Jamaludin, "Trace Analyzer for NS-2", IEEE, pp. 29-32, Student Conference on Research and Development (SCORED), Malaysia, 2006.
4. Vasilis Gerogiannis, D.Achilles Kameas, E.Panayotis Pintelas, "Comparative study and categorization of high-level petri nets", The Journal of Systems and Software (43) 133±160, 1998.
5. Er. Deepinder Singh Wadhwa and Er. Tripatjot Singh Panag, "Performance Comparison of Single and Multipath Routing Protocols in Ad hoc Networks", Int. J. Comp. Tech. Appl., ISSN: 2229-6093, Vol 2 (5), pp.1486-1496, 2011.
6. Jae Chung & Mark Claypool "NS by Example", worcester polytechnig institute, computer science, 2002. Available in the page: http://nile.wpi.edu/NS/ [08/01/2002 9.09.25].
7. Liu, J.HU and J.LI, "Diagnostic knowledge representation and reasoning based on fuzzy fault petri nets", IEEE, 2011.
8. M. Georgi Dimirovski, "FUZZY-PETRI-NET REASONING SUPERVISORY CONTROLLER AND ESTIMATING STATES OF MARKOV CHAIN MODELS", University of Istanbul, Computing and Informatics, 563–576, Vol. 24, 2005.
9. Mamoun Hussein Mamoun, "A New Reliable Routing Algorithm for MANET", International Journal of Research and Reviews in Computer Science (IJRRCS), Vol. 2, No. 3, June, 2011.
10. Oscar Cordon, Francisco Herrera, Frank Hoffmann & Luis Magdalena, "GENETIC FUZZY SYSTEMS: EVOLUTIONARY TUNING AND LEARNING OF FUZZY KNOWLEDGE BASES", Book, Copyright by World Scientific Publishing Co. Pte. Ltd. Vol. 19, 2001.
11. SEM_H DOKURER, "SIMULATION OF

BLACK HOLE ATTACK IN WIRELESS AD-HOC NETWORKS", Master thesis in Computer Engineering submitted to Atılım University, 2006.
12. Siddesh. G.K, K.N. Muralidhara and Manjula. Harihar, "Routing in Ad Hoc Wireless Networks using Soft Computing techniques and performance evaluation using Hypernet simulator", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-1, Issue-3, July, 2011.
13. Sonam Jain and Sandeep Sahu, "Geometric Routing Protocol based on Genetic Algorithm for Minimized Delay in MANETs", International Journal of Electronics Communication and Computer Technology (IJECCT) Volume 2 Issue 3 , May, 2012.
14. Subramanyam, P.V & Chauhan, A, Singh, "Modelling of a Hybrid Protocol for a MANET using Fuzzy Petri Nets", India Conference, 2006 Annual IEEE, Print ISBN:1-4244-0369-3, Sept, 2006.
15. Taqwa Odey Fahad & Prof. Abduladhim A. Ali, "improvement of AODV routing on MANETs using fuzzy systems", University of Basra, Iraq J. Electrical and Electronic Engineering, Vol.7 No.2, 2011.
16. Tzu-Chiang Chiang and Yueh-Min Huang, "Multicast routing representation in ad hoc networks using fuzzy Petri nets", Advanced Information Networking and Applications, 2004. AINA, International Conference on (Volume:2), 2004.
17. Tzu-Chiang Chiang, Cheng-Feng Tai and Ting-Wei Hou, "A knowledge-based inference multicast protocol using adaptive fuzzy Petri nets", International Journal of Expert Systems with Applications, Vol 36, Issue 4, Pages 8115–8123, May, 2009.
18. X. Li, F. Lara-Rosano, "Adaptive fuzzy petri nets for dynamic knowledge representation and inference", Centre for Instrumentation Research, National University of Mexico (UNAM), Expert Systems with Applications (19) 235–241, 2000.
19. Zahra Masoudi Nejhad, Ali Harounabadi and Mohammad Teshnehlab, " Mapping fuzzy formal model (FPN) to fuzzy extended markup language (FXML), (FPNML)", Organized by Khavaran Institute of Higher Education, Bandar-Abbas, Iran, 2013.
20. Zhenhua Yu, Xiao Fu, Yuanli Cai and Mehmet C. Vuran, "A Reliable Energy-Efficient Multi-Level Routing Algorithm for Wireless Sensor Networks Using Fuzzy Petri Nets", journal sensors ISSN 1424-8220, 11, 2011.
21. Zhi-gang Hu, Hao Ma, Guo-jun Wang, Lin Liao, "A reliable routing algorithm based on fuzzy Petri net in mobile ad hoc networks", Journal of Central South University of Technology, Volume 12, Issue 6, pp 714-719, December, 2005.

8/3/2014