Journal of Advances in Computer Research
Quarterly pISSN: 2345-606x eISSN: 2345-6078
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 7, No. 2, May 2016), Pages: 23-40
www.jacr.iausari.ac.ir

An Efficient Fault-Tolerance Routing Algorithm for Mobile Ad-hoc Networks[1]

Fatemeh Tavakoli✉[1], Meisam Kamarei[2], Gholam Reza Asgari[1]

1)Department of Computer Engineering, Aligudarz Branch, Islamic Azad University, Aligudarz, Iran
2)University of Applied Science& Technology (UAST), Tehran ,Iran

tavakoli@iau-aligudarz.ac.ir; kamarei@uast.ac.ir; asgari@iau-aligudarz.ac.ir

Received: 2015/05/06; Accepted: 2015/09/02

Abstract

In this paper, an efficient fault-tolerant routing algorithm for Mobile Ad-hoc Networks (MANETs) is presented. The proposed algorithm increases the network fault-tolerance using natural redundancy of Ad-hoc networks. This algorithm is carried out in two stages; 1) the selection of backup nodes 2) the selection of backup route(s). In the first stage, the proposed algorithm chooses nodes with the same path as backup nodes. Prediction and diagnosis of nodes' paths is performed through backup tables. Since the selection of backup nodes is fulfilled, the proposed algorithm begins fault-tolerance routing. For this purpose, initially the proposed algorithm provides the main route between each pair of source & destination nodes based on DSR routing algorithm. Then, from a destination node towards a source node, the backup route(s) is established between the chosen backup nodes in the first stage. Experimental results taken from NS-2 simulator demonstrate that in comparison with previous methods the proposed increases; 1) 10% the package delivery ratio against the percentages of faulty nodes and, 2) 22% package delivery ratio against the pause time of various mobile nodes.

Keywords: *Backup Nodes, Mobile Ad-hoc Networks, Fault-tolerance, Redundancy, Routing.*

1. Introduction

Fault tolerance is an important capability of Mobile Ad-hoc Networks (MANETs) enabling the system to function properly in case a fault occurs in the system components [2-3].The purpose of fault tolerance mechanism is to provide the network users with reliable MANETs. In this regard, even when faults occur, the system proceeds to function properly. Therefore, the most important aim of fault tolerance techniques is assuring the provision of consistent and reliable services to the network users [4].MANETs are extremely susceptible to fault in their performance. Faults occurring in the operation of mobile nodes and the transferred data are the main sources of faults in MANETs. The most important causes of faults in the operation of MANETs

[1]The submitted manuscript is an extended version of the work previously proposed in [1]. The extensions of the current version include: (1) a wide range of simulation experiments added to the current version. The previous version of the paper only chooses backup nodes to faulty nodes, but (2) the current version of paper uses the backup nodes that are chosen by the pervious method to make backup routes. Therefore, the current version of paper makes backup routes as multiple paths routing to fault tolerance enhancement against packet loss. Thus, the current version of paper with backup nodes selection and multiple paths routing tries to increase MANET fault tolerance.

are considered as [5-6] limitations in the computing resources of nodes, limitations in the energy supply of nodes, faults occurring in one of the hardware resources of the mobile nodes, and faults occurring in transmitted data. Therefore, increasing the fault tolerance of networks against various types of faults is an important issue on which the researchers are required to pay more heed.

Redundancy at different levels of the network is the most effective tactics to increase fault tolerance of the network [6-8]. Hence, during the network lifetime, each point within the network is covered by at least one node. So, redundancy is a natural phenomenon in ad-hoc networks. Considering the fact that node in MANETs play the role of both the router and the host simultaneously, redundancy in MANETs at the level of routers and nodes seems to be quite normal. Therefore, the use of natural redundancies in MANETs is likely to increase their tolerance against various faults. The appropriate data exchange within the network is the paramount factor to evaluate the performance of the proposed routing algorithm for MANETs [9]. Accurate data delivery rate depends highly on the performance of the routing algorithms. Therefore, making different decisions based on the natural redundancies of MANETs and routing algorithms play an important part in increasing the fault tolerance of such networks against various sorts of faults.

In this regard, researchers have attempted to increase the fault tolerance of the system against those faults occurring in data transmission through applying redundancy at the level of transmitted data using multi-path routing algorithms [7]. In multi-path routing algorithms, several identical data are transmitted to the destination node via different paths. Researchers have been looking forward to observe that between several transmitted data, at least the successful data transmission is achieved from the source to the destination nodes [5][8][10] .The biggest drawbacks of multi-path routing algorithms are increasing redundant transmissions within the network, and increasing the number of end to end delaying of the transmitted data [11]. Although multi-path routing algorithms only regard redundancy at the router nodes level, due to the increase of the network fault tolerance in data transmission, little heed has been paid on redundancy at the host nodes level.

In this paper, a fault tolerance routing algorithm is presented which applies the natural redundancy of MANETs to increase the networks` fault tolerance. For this purpose, the proposed algorithm functions in two phases: 1) choosing the backup nodes and, 2) choosing the backup routes. In the first step, the proposed algorithm uses the nodes with the same movement path as the backup nodes for the main nodes. Identifying and predicting the path of mobile nodes is conducted through the backup nodes` table. After selecting the backup nodes, the proposed algorithm starts fault tolerance. For this purpose, a path is first established based on DSR routing algorithm from each source node to its paired destination node along the main route. After that, from the source to the destination nodes, the backup path(s) between the backup nodes, which have already been identified by the proposed algorithm, usually route are created.

The rest of the paper is organized as follows. In the second section, the review of the related work is presented and discussed. Section three explains the proposed fault tolerance algorithm. Simulation results of the proposed algorithm are presented in Section four. Finally, section five has been allocated to concluding and future research concluding this paper.

2. Related Work

MANETs are highly susceptible to fault. Therefore, routing algorithms should guarantee the tolerance of systems against various types of faults. In the present section, some of the most important relevant studies which have already been conducted regarding fault tolerance routing algorithms in MANETs are investigated.

The authors in [15] have proposed a fault tolerance routing algorithm based on the learning automata which is capable of routing using multi-path routing, even in the presence of defective nodes in MANETs. In the proposed algorithm, the automata theory has been used for optimizing the route selection, reducing the overhead in the network, and learning about the faulty nodes. In [16] an algorithm is proposed which establishes two paths between source and destination nodes. In this algorithm, the backup paths are generated during the routes response process. Furthermore, the handling process and the local improvement of the local path are performed to improve data transmission and fault tolerance of the network. In [7], the authors have provided a fault tolerance routing algorithm based on the DSR algorithm. The proposed algorithm attempts to find two paths, if possible, from the source node to the destination one. During the process of path detection, this protocol identifies several paths which are recognizable by the DSR protocol. Simulation results of the proposed algorithm reveal that in term of number, it has the fewer data overhead and that the size of control messages is lower in comparison with DSR algorithm. In [17], the authors have probed the communication failures in the network based on distributed genetic algorithms (GA) with various topologies. They have also evaluated the distributed GA performance and behavior under various levels of continual communication failures using the network sorting issue as a practicable application.

In [18] the restriction of the fault tolerance power of AODV routing protocol in MANETs is proposed. The first and foremost purpose of this protocol is to adopt better to topology modifications concerning scalability and mobility. In comparison with the original AODV protocol, the efficiency of the protocol reveals to be higher and that the likeliness of removing packets, delay time, operation power, and routing overhead has improved.

In [19] a procedure has been proposed for increasing the fault tolerance with optimizing the distance between nodes using graph theory which considers the least weight based on the algorithm to reduce the power consumption of each node. The purpose of topology monitoring is preserving intended topology specifications of the network to improve the performance of the algorithm in the network. In [20], the authors have proposed a fault tolerant cluster head based on a routing protocol in MANETs which is likely to reduce the malfunctioning nodes in the network. The proposed protocol guarantees message delivery in the presence of faulty nodes and also reduces overhead. The performance of this protocol is compared by MMMH, AODV, and DSR protocols. This protocol is proved to be functioning better in comparison with these protocols in the presence of faulty nodes. In [21] the authors have proposed the location based that fault tolerance routing algorithm in MANETs. In this algorithm, which is based on the geographical location data, the network is divided into sub-networks. When a fault occurs during the typical routing process in the network, the proposed algorithm chooses an alternate path which had not been implemented in the hop. Route selection depends on the local information of the neighboring subnets. The proposed algorithm is able to distract the route of the faulty area solely using the local

information of the neighboring networks which is totally suitable for the dynamic networks. In [22], the authors have proposed a spiral routing algorithm in MANETs inspired from a millipede. The proposed algorithm has used a biologically inspired technique which makes that use of a light source to reduce the routing overhead and improve the fault tolerance in malfunctioning links. In [23] the authors have analyzed the fault tolerant adaptive replication routing protocol. This protocol takes advantage of time-taking messages to detect faulting in the network. This allows the investigation of the routing protocol`s efficiency based on the three parameters of packet delivery ratio, routing overhead, and throughput under five influential factors including network size, transmission rate, nodes mobility, pause time, and an optimal number of copies.

In [24] authors have proposed a mutual exclusive algorithm in MANETs. The proposed algorithm is able to tolerate both the host and the link failures using time-based mechanism. Simulation results show that the proposed algorithm functions properly under various conditions, especially when mobility is high or load level is low. Besides, it is capable of storing most of the communication costs. In [25] a new method for replicating files within MANETs has been proposed which is based on prognosticating the deviations of the mobile nodes. The only reliable way to simplify file sharing in mobile networks is to replicate filing. In [26] authors proposed that power restrictions of fault tolerant AODV in MANETs. The main objective of this protocol is gaining more compatibility to modify the topology in the network concerning better scalability and mobility. When compared with the original AODV, the performance of this protocol show to improve the risk of the packets removal, throughput, delay, and routing overhead.

3. Proposed Algorithm

Fault tolerance defined the ability of the network to function properly against the failures nodes and communication channels defined .The happening of faults in MANETs can be quite persistent or transient. Besides, faults can originate from hardware or software sources [27]. Transient faults are usually removed on their own immediately after they happen in the network. Therefore, like almost all other studies and researches in this area, the present paper is aimed to increase the fault tolerance of the network against persistent faults. Mobile nodes persistent faults within a path can happen due to such causes as disappearing of the nodes because of their movement, ending energy sources of the nodes, environmental disasters like fire and flood, damaging the nodes by human beings or animals, etc. [28]. In order to increase faulting tolerance of the networks most researchers have only focused on the faults happening in the transmitted data.

The more researchers have considered fault occurrence on data transferring to the network fault-tolerance enhancement. In this regard, multi-path routing algorithms have been proposed. Of course, fault tolerance enhancement against nodes faulty has large impact on the network fault tolerance enhancement. This paper proposes an efficient fault tolerance algorithm that 1) chooses backup nodes to increase the network fault tolerance against faulty nodes, 2) uses chosen backup nodes to make a multi-path routing algorithm to increase the network fault tolerance against faulty data transferring.

3.1. Select the Backup Nodes[1]

Considering the fact that MANETs are always in motion and these networks are very dynamically, so the selection of backup nodes is a big challenge for all researchers. To solve this problem, the adjacent nodes are to be taken as backup nodes. On the other side, defining different algorithms for these networks need to take placing with the least number of redundant data transmissions. For reducing the amount of redundant data transmission within the network, the proposed algorithm uses the following strategies:

- Backup nodes can be used either as cold or hot in order to be replaced for the primary nodes. In the cold backup method, immediately after a fault happens in the primary node, the backup node is replaced for the primary node. In hot backup method, backup nodes are replaced for the primary nodes during certain time intervals alternatively [5]. In [5] it has been shown that the cold backup method has less data over head and that it is more efficient than the hot backup method. Therefore the proposed algorithm applies the cold backup nodes to replace with primary mobile node.
- Node with the same route require less data exchange compared to the nodes which have just been placed in the primary nodes route. Therefore the proposed algorithm uses those nodes as the backup nodes which most likely have the same route as the primary node. Therefore, the nodes are informed of the adjacent nodes in specified time intervals and after receiving data from the adjacent nodes, each node attempt to insert new data in its backup nodes table, as shown in table 1.

The proposed algorithm also defines the backup nodes table for each mobile node to assign backup nodes to them. Table 1 show the structure of a typically backup table of mobile nodes. This table is used to store the information related to the adjacent nodes in each mobile node. $Node.ID.$ In this table represents the ID belong to the adjacent node. $Time$ Field this table represents the period of time when the two nodes have been adjacent. $Re-select$ Field in this table represents the number of times that each node has been waiting to receive data from the adjacent nodes in order to specify their current state. $Status$Field in this table represents the enabled or disabled status of the adjacent nodes. To update the backup table, the following instructions should be followed step by step:

Table 1. The structure of a backup table in mobile [1] ***nodes.***

Node. ID	Time	Re-Select	State
10	12.54	0	1
5	0	0	1
7	22	1	0

a. In order to be informed of the current status of adjacent node, mobile nodes attempt to send a hello message at certain time intervals.
b. Before sending hello message to adjacent nodes, each node set the status field value of all the rows in the table equal to zero. The reason is being able to identify enabled or disabled status of all adjacent nodes right after the backup table is updated.
c. As soon as the adjacent nodes receive the hello message, the ID is transmitted to the node which has sent the hello message.

d. The sender node searches the received IDs in its backup table right after the receiving the response. After searching the nodes IDs in the backup table, one of the following issues may occur :

1. The new $Node.ID$ may not be found in the backup table. In this case, receiver node inserts the new nodes ID into the backup table. Obviously the time when the two newly nodes have been adjacent to each other must be zero. Besides, the value of $re-select$ field for this sample of nodes is zero and the value of the status field is considered as active or one. In table 1 it is supposed that node 5 has the following condition. This table specifies the nodes which have recently been placed next to the current node.
2. The new $Node.ID$ already exists in the backup table and the value of $re-select$ field of the node is zero. Obviously, in this case the adjacency time of the two nodes and also the nodes adjacency status field are changed from inactive, i.e. zero, to active and no change is required in the value of $re-select$ field. This has happened for the node 10 in table 1. This state indicates that the new node has already been adjacent to the current node and that the adjacency time of the two nodes must increase.
3. No response is sent from the nodes which have already been inserted in the table. This may happen due to the fact that the new node and the current node have not been adjacent yet may originate from an occurrence of faults in the transmitted data resulting in unsuccessful transmission of data. Using the adjacent nodes field in the backup nodes table enables us to separate nodes having this status. In such cases, the value of status field of mobile nodes is zero, i.e. inactive, in the backup table. For those nodes with such states, the proposed algorithm gives them another chance to specify the status of all adjacent nodes. This is because it is possible that either the transmitted data has been lost while transmitting or the new node may be far away from the current node in a way that it is out of access for the current node for a while. In both of the above-mentioned states, it is likely that several new nodes having similar conditions may become adjacent to the current node again. For this reason, mobile nodes usually add one unit to the $re-select$ field of the new nodes when the time given to the hello message is up. If in the process of sending a hello message a response is received from the nodes with zero value for their $re-select$ field, the proposed algorithm sets the value of re-select field equal to zero again and tries to upgrade the time and status fields of the node again.

At the end of each sending hello message stage, the proposed algorithm removes the rows from the backup table with the $re-select$ field value of 2 and inactive status field. This happens because after allocating 2 chances to each node, most likely the nodes move to other points of the network and are no longer adjacent to the current node. Therefore the value of $re-select$ field must always be 0, 1 or 2.

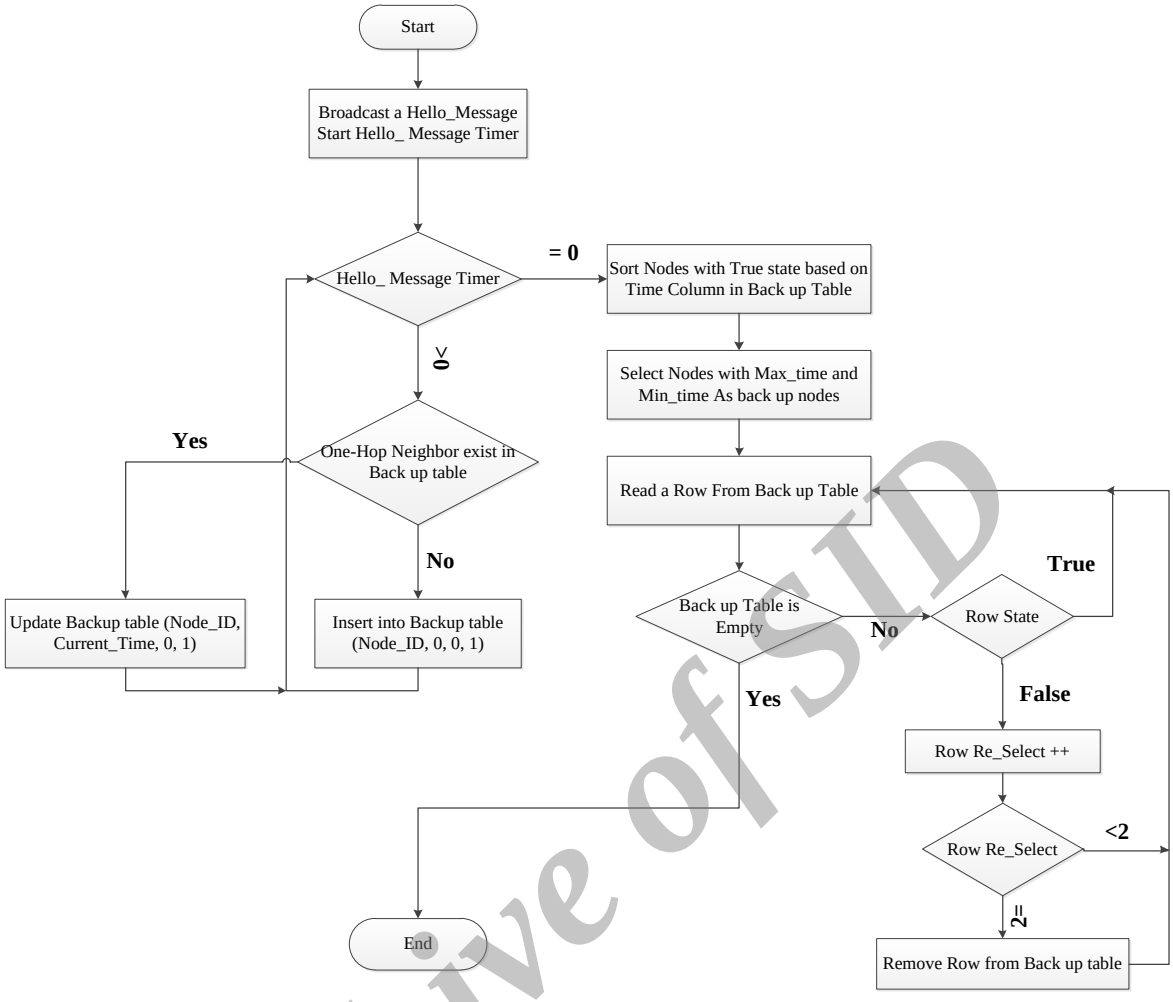

Figure1: Pseudo code of the proposed algorithm[1].

The nodes movement route is the most essential and effective parameter to be used to choose the backup nodes in MANETs. This is possible using the nodes movement history which enables the researchers to predict the movement route of the nodes. The movement of nodes is pretty random to all points of the network [8].This makes it almost impossible to predict the future movement rout of the nodes. Regarding the backup table, there must be the plethora of possibilities about the movement of the nodes. For instance, the nodes which have been adjacent for a long time, most likely will have the same movement route in future. On the other hand, the nodes which have recently been adjacent will probably move in the same direction for a long period of time. For the sake of increasing the accuracy of guesses made about the same movement route of the nodes, both the above-mentioned probabilities are taken into account. According to the pseudo code of the proposed algorithm in figure 1, after nodes are informed of their adjacent nodes status, the backup table of mobile nodes is sorted as ascending and descending based on the duration of adjacency only for those nodes with active state field. The selection of backup nodes for each node will be as a set of paired nodes. Hence, for each node a pair of two adjacent nodes is taken as the backup nodes one of which with the highest adjacency time and the other with the least adjacency

time. If a fault happens in the operation of each node, the selected backup nodes are replaced for the removed node.

Considering the random movement route of the nodes, the proposed algorithm is making an attempt to predict the nodes movement route. Thus after completing the backup table, the proposed algorithm selects a pair of nodes as the backup nodes for the mobile nodes. As soon as fault happens in the performance of the nodes, firstly the node having the most duration of adjacency is chosen as the backup node. If a fault or problem happens for the first backup node, the backup node with the least adjacency time is replaced for the primary mobile node. For sure there are some priorities in assigning the nodes as the backup nodes for the primary mobile node. Therefore the selection of the main backup node to be replaced for the primary node is done based on those priorities. The proposed algorithm chooses several backup nodes for each node, based on the nodes movement. The proposed algorithm is a distributed algorithm and it can be used for MANETs with high dynamically. Figure 1 show pseudo code of the proposed method. According to this figure, it can be seen backup nodes are selected based on backup table of mobile nodes.

3.2. The Proposed Fault Tolerance Routing Algorithm

In this section, the fault tolerance of network against data transmission faults and also the reliability of transmitted data increases using the backup nodes selected in the previous section. The most important causes of unsuccessful data transmission in MANETs are wireless communication channel faults between mobile nodes that are 10%to 30% [29] and the completion of the input buffer capacity [30].

To increase the fault tolerance against data transmission faults, the use of the multipath routing algorithms has been paid much heed on by the researchers. In multipath routing algorithms several paths are created between source and destination nodes [31]. Researchers take two views into account in order to use different paths between source and destination nodes. In the first view, among all available paths only one of them is used as the main route for data transmission while the rest of them are taken as the backup paths for the main route. On the other hand, in the second view, the source node attempts to broadcast multiple copies of the same data through all the available paths in the network towards the destination node. Researchers are looking forward to observing that among all these transmissions, at least the successful data transmission occurs between source and destination nodes [30]. Minimizing redundancy transmission and choosing the best paths as the backup path for the main path are considered to be the most important challenges for the researchers in the effective development of multipath routing algorithms. Furthermore, the increase of network traffic, the increase of energy consumption of nodes, and the increase of end to end delay are the major drawbacks of the aforementioned algorithms. In the following the fault tolerant routing algorithm is presented which attempts to increase the fault tolerance of the network using backup routes and nodes.
Fault tolerant routing algorithm is fulfilled in two phases:

1. Creating the main path between source and destination nodes
2. Selecting the backup path for the main path

Creating the main path between source and destination nodes using the routing procedures is done on demand. Therefore, the node that is supposed to transmit the data

to a defined destination node broadcasts the request to establish a path to the destination node in the network right away. The identification of main path between source and destination nodes is carried out based on DSR routing algorithm.

DSR routing algorithm is an efficient and suitable routing algorithm for MANETs. This algorithm is performed in two phases of route discovery and route maintenance. In the discovery phase, the source node tries to find a path to the destination node. This process is applied when the source node is trying to transmit data to the destination node, but no path between the two nodes already exists. Route maintenance is a mechanism by which the source node is able to detect changes in the network connectivity, since in this case, the validity of the reserved path is expired and is no longer usable. Once the route maintenance process recognized that the route is no longer valid, the source node is compelled to transmit data through another route. Route maintenance is only used when the source node is transmitting data to the destination node [32].

Route discovery in DSR algorithm initiates when the source node inserts the address of the destination node into a packet and broadcasts it to all the neighboring nodes. If a node receives the RD-request from its neighboring nodes and is not able to observe the above-mentioned packet, the node attaches its identification number (ID) to the end of the packet and transmits it to all the neighboring nodes again. This routine continues by all nodes until the destination node received data. After receiving the first RREQ packet, the destination nodes sends a RREP packet, which has been inserted in the RREQ packet, back to the destination node through a specified route.

Regarding DSR algorithm, only one path between source and destination nodes is taken into consideration at a moment. In cases of nodes failure along the path, the path between source and destination nodes is omitted and it is unlikely that plethora of transmitted data is entered the destination node. On the other hand, increasing route updating phases in this algorithm increase both the network congestion and the energy consumption of nodes.

First, the proposed fault tolerant routing algorithm establishes a main path between the source and the destination nodes using the DSR algorithm. Then using the backup mobile nodes, the algorithm considers several backup routes for the main route. For this purpose, the following phases need to be followed to establish main and backup paths.

1- In order to create a route to the destination node, the source node broadcasts the RREQ packet, similar to the DSR algorithm, within the network.
2- After receiving the RREQ packet, each node act similar to the DSR algorithm and adds its ID to the end of the packet, sending it back to all its neighbors.
3- After the RREQ packets were transmitted, each node update its backup table using the algorithm described in figure 2. Therefore, the node needs to send a "hello message" to all its neighbors. Concurrent with the release of RREQ packet, the backup tables of the nodes along the route are also updated.
4- After receiving the RREQ packet, the destination node initiates the process of reserving the main path by broadcasting RREP packet from the destination node. Reserving the main path is performed similar to the DSR algorithm. However, the most important issue in transmitting the RREP packet back is creating backup routes parallel with the reserved main route. Unlike creating the main route, the process of creating backup routes begins from the destination nodes side.

5- To create backup routes, it is needed to add some content fields to the RREP packet. The number of fields added to the RREP packet is equal to the number of the defined backup paths. Since the backup nodetakingalgorithm of the nodes uses two nodes as the backup nodes for the main node,hence, two additional content fields are to be added to the RREP packet to add the backup nodes ID to them.
6- The destination node adds the IDs of its two backup nodes to the end of the RREP packet and, similar to the DSR algorithm, transmits it to the source node within the RREQ packet from the same route which was already used.
7- After receiving the RREP packet, the existing nodes on the way back to the source node, store the IDs of the backup nodes of the previous main node in the packet header in their local memory and replace their backup nodes IDs to the previous ones.
8- Nodes receiving RREP packet send the previous backup nodes IDs to their own backup nodes. According to the previous backup node IDs, the new backup nodes try to create a path for their own nodes. Thus, concurrent with the returning of the RREP packet to the source node, two backup paths are created parallel with the main path between the existing backup nodes along the main path.
9- If there is no path from the present node to the nodes of the previous phase, the present newly selected node informs the main node of the matter. Since there are several backup nodes for each main node, the main node sends a request to its backup nodes and asks for the previous nodes IDs. If one of the backup nodes of the previous phase is neighboring the present backup nodes, the later is replaced for the former.

It is clear from figure 2 that concurrent with the return of RREP packet towards the source node, the backup routes are also created between the backup nodes. Since the proposed algorithm first attempts to predict the direction of the nodes, and then uses the nodes with the same directions the backup nodes, it is likely that backup route are created in the same direction as the main path. On the other hand, mobile nodes attempt to get familiar with their neighboring nodes at the same time as RREQ packet releases. Therefore, since the information about the neighboring nodes is totally recent, the possibility of having the same directions for the backup nodes and the main node until the last moment of the routes life increases. However, the existence of at least one of the backup routes is likely to increase fault tolerance in the network. The proposed algorithm pseudo code for selecting backup paths can be seen in figure 2.From figure 2 clearly after receiving the RREQ packet in the second phase of the algorithm, mobile nodes broadcast RREQ and "hello message" packets to all their neighbors simultaneously. According to the above-mentioned algorithm, the reservation of the backup routes is made in the fourth phase of the backup route reservation algorithm. Besides, the backup routes when RREP packet is returning to the source node are reserved and specified.

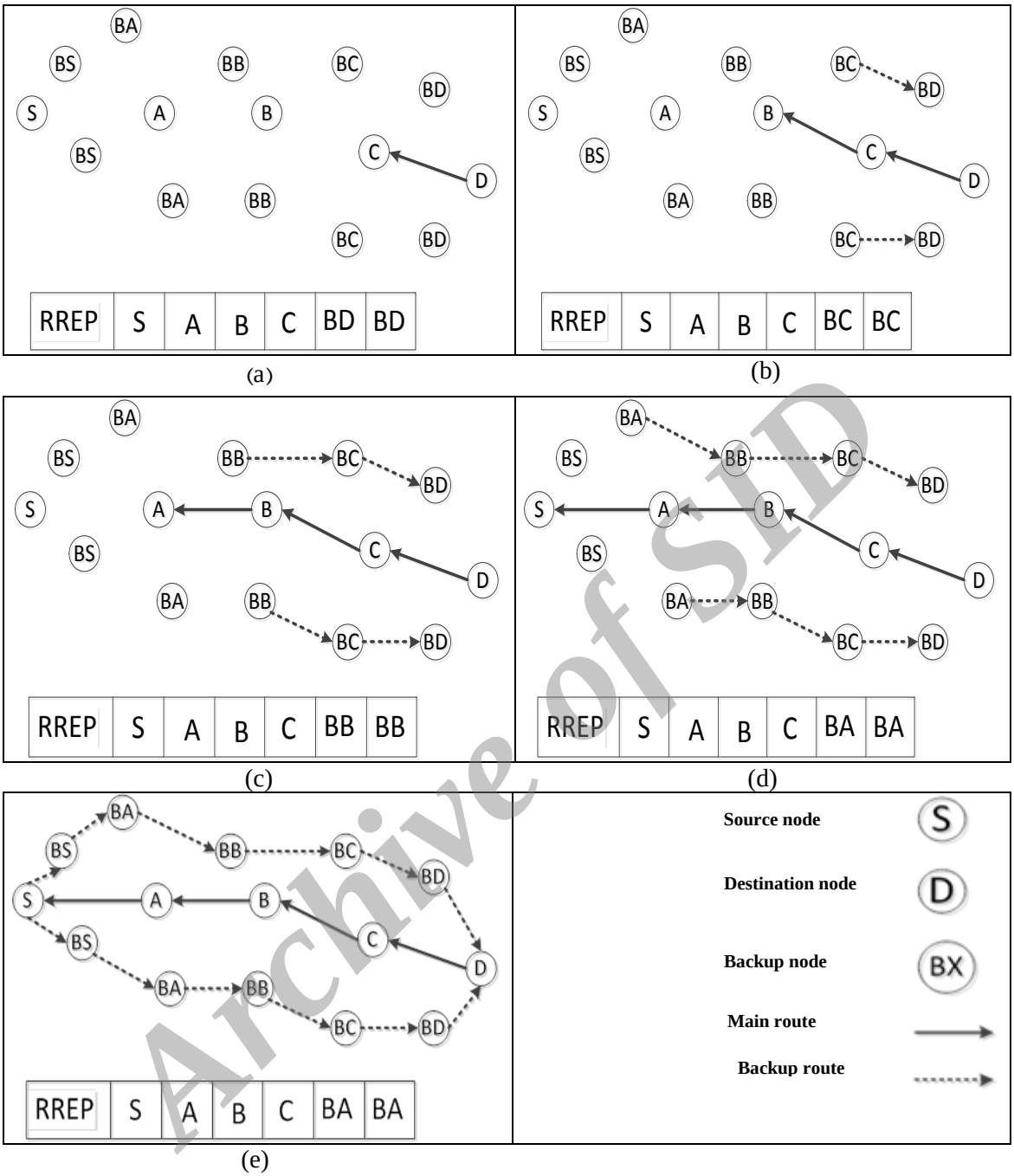

Figure 2. The process of creating the main route and backup routes between source and destination nodes of the proposed algorithm

```
Output Backup Route Reservation between Source & Destination nodes
Step 1: Flood RREQ packet by Source node
Step 2: If Receive RREQ in intermediate node then
        Flood RREQ packet
        Flood Hello message& Update Backup table
Step 3: If Receive RREQ in Destination node then
  Insert two backup node_ID into RREP and sent to previous hop
Step 4:While Receiver node is nor Source node
If receiver node is main node then
Read Backup nodes fieldsfromRREP packet
   Transfer previous backup node_ID to itself Backup node
   Replace itself backup node_ID to RREP packet
  Transfer RREP packet to previous Hop
 End
 If Receiver node is backup node then
Send message to previous backup node
End
```

Figure 3. Pseudo code the backup path selection of proposed algorithm

4. Evaluation of the Proposed Fault Tolerant Routing Algorithm

In this section the performance of the proposed fault tolerance algorithm is evaluated using NS-2.35 simulator. Various parameters used in the simulation process are selected according to table 2of [27]. For this purpose, a total of 50 nodes are randomly distributed in an area of 2000m × 2000m.The nodes speed and mobility vary based on the stopping time. Mobility model of the nodes in the simulation environment is also a model of random movement. The radius of data transmission by the nodes is considered 250 m. All implemented scenarios of simulation carry out totally in 900 seconds.

Another parameter implemented in MANETs simulation scenarios that needs to be taken into account is traffic generation model for mobile nodes. For this reason, the model presented in [27] has been used in order to generate mobile nodes traffic model. In the proposed model in order to transmit and receive the defined data, four pairs of nodes are used as the source and destination nodes in which the source nodes attempt to create packets regularly and in certain time intervals and transmit them to the destination nodes. Traffic generation model used in the proposed algorithm is the CBR model, and the schedule of data the source nodes set transmission according to table 3. The fields in table 3 are the source node address, destination node address, numbers of packets transmitted, size of the packets, packet transmission intervals, and transmission start time and end time by the traffic generating source, respectively. Choosing and setting various parameters affecting simulation scenarios initiates the proposed algorithm implementation process.

Table 2.Parameter values inthe simulation

Terrain	$2000m \times 2000m$
Simulation time	900 s
Number of nodes	50
Mobility model	$random\ way\ point$
Speed of the mobile nodes	$0\ to\ 10\ m/s$
MAC protocol	$IEEE$ 802.11
Radio propagation range	250 m
Radio propagation model	$Two\ Ray$
Channel Type	$Wireless\ Channel$

***Table 3.*Timing of traffic TGTFG generation between nodes**

source	Destination	items to send	item size	interval	start time	end time
18	16	10000	512	5 S	70 S	100 S
10	27	10000	512	2.5 S	82.49S	199 S
21	0	10000	512	0.8 S	91.39S	248 S
14	17	10000	512	1.1 S	107.8S	274 S
18	16	10000	512	5 S	70 S	100 S

Figure 4 show the impact of the number of faulty nodes on the received packet rate at the source node. Clearly with the increase of faulty nodes within the network, the received packets rate reduces not only in the proposed algorithm, but also in DSR algorithm [28] and AntHocNet [28]. However, since two backup nodes are chosen in the proposed algorithm, the percentage of packet delivery is higher compared the two other algorithms.Figure 5 show the backup nodes updating costs against the percentage of the faulty nodes. According to figure 5, the growth of the redundant packets exchange in 0% to 50% of faulty nodes in the proposed algorithm is about 10% while it is almost 30% in the two algorithms DSR and Ant Hoc Net. The proposed algorithm uses several pairs of nodes simultaneously as alternate nodes for the main node. Therefore, the backup tables are updated only when no backup nodes are available. Thus, according to figure 5 the reduction of the number of nodes updating has reduced the exchange rate of the redundant packets. The impact of the pause time of the nodes is another essential parameter that needs to be considered to evaluate and compare the performance of the proposed algorithm with the previous ones. Pause time is inversely related to the dynamicity of the network; e.g. the less the pause time of the nodes is, the more their speed in the network is, leading to the increased network dynamicity. According to figure 6, the increase of the pause time decreases the mobility of the nodes and the dynamicity of the network. The reduction of the network`s dynamicity improves the accuracy of the backup nodes table in mobile nodes. According to figure 6, it can be observed that increasing the nodes pause time is likely to increase the data exchange rate in the proposed algorithm. Figure 7 show the impact of pause time of the nodes on the redundant packets exchange rate within the network. According to figure 7, it can be mentioned that increasing the pause time of mobile nodes decreases the redundant

packets exchange rate within the network. On the other hand, because of the exchange of the redundant packets in order to reserve the backup routes in the proposed algorithm, the number of redundant packets exchange is more than those of DSR and Anti HocNet algorithms. As mentioned, transmitted redundant packets accompanied with the RREP packets for reserving backup routes is the most important reason of increasing the exchange rate of the redundant packets in the proposed algorithm in comparison with the DSR algorithm. Figure 7 show that increasing in the pause time of the nodes reduces the redundant packets exchange rate in the proposed algorithm. This is due to the increased stability of the main and the backup paths between source and destination nodes, at the high pause time. Figure 8 show the end to end delay of packet delivery against the nodes pause time. Based on the figure, it can be seen that increasing the pause time of the nodes, declines the average end to end delay packet delivery. This is because the increase of the nodes pause time increases the accuracy of the backup tables and consequently the average of successful data transmission as well. On the other hand, the increased accuracy of the backup tables and the reduced dynamicity of the network reduce the updating time needed for the routes. Therefore, increasing the pause time of the nodes is likely to decrease the nodes end to end delay. Figure 9 show the development of the proposed algorithm against the rate of successful delivery of packets to the destination node. Increasing the number of nodes in the network increases redundant packets exchange within the network that is necessary to specify the backup nodes. Increasing the redundant packets exchange lead to networks congestion. Increasing networks congestion reduces the rate of delivery of packets within the network. Although increasing the number of nodes in the network increases the number of backup nodes for each node and backup tables accuracy, network congestion is the most important reason to reduce the successful exchange of packets within the network, since it tries to exchange the redundant packets.

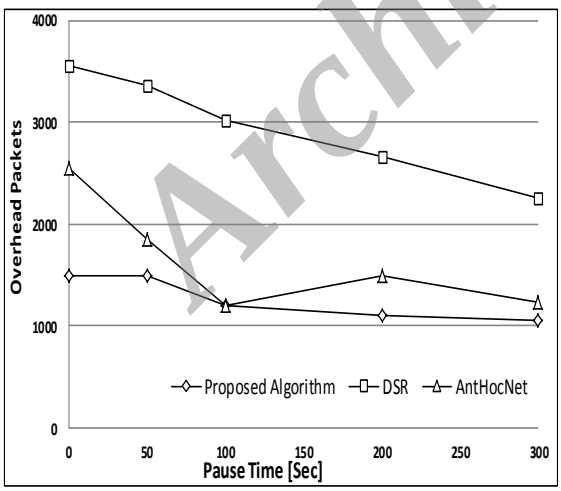

Figure7. Effect of pause time on the exchange ratio of overhead packet

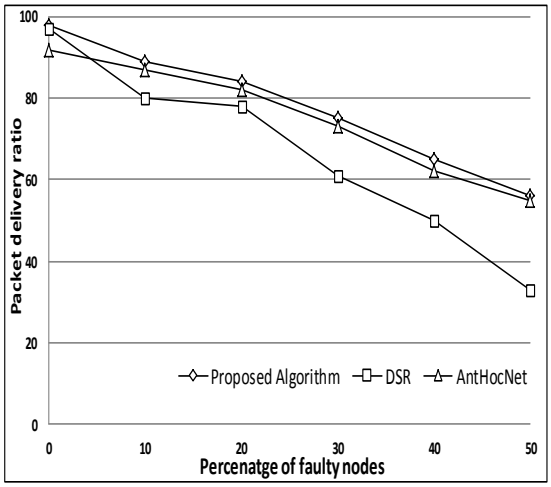

Figure4.Effect of number faultynodes vs. packet Deliveryratio.

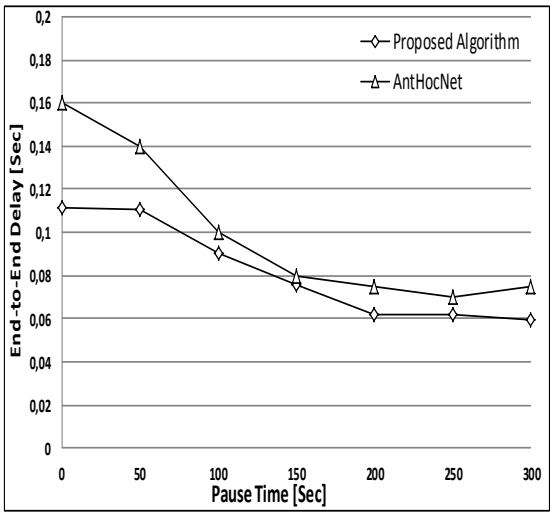

Figure8. Effect of pause time on average end to end delay packet delivery.

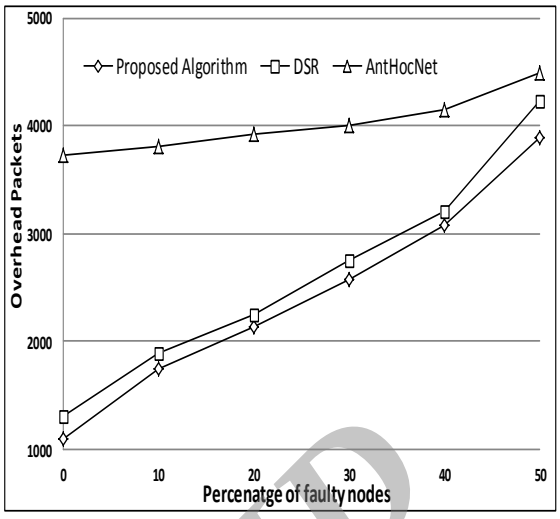

Figure5.Effect offaultynodes on the exchange ratio of overhead packet.

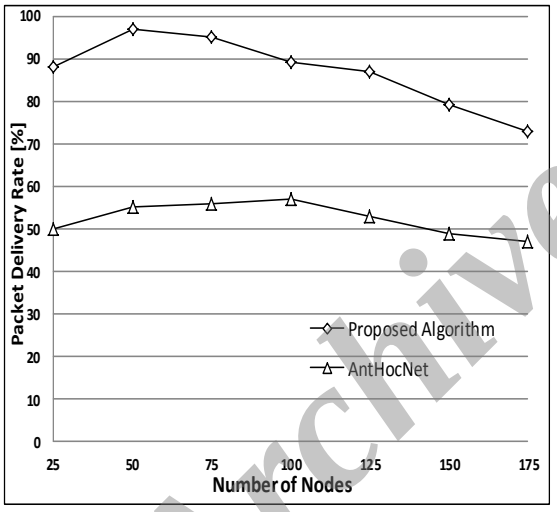

Figure 9 .Development of the proposed algorithm vs. Successful packet delivery ratio

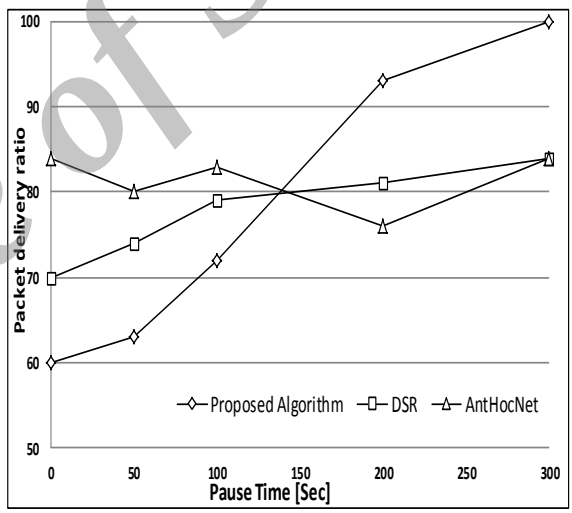

Figure6 .Effect of pause time vs. Packet delivery ratio.

We evaluate end to end delay on our proposed algorithm. Figure 10, shows the network end to end delay versus the faulty nodes. Based on this figure it can be seen with increase in the number of faulty nodes the network end to end delay increase too. Increase in the number of faulty nodes leads to increase in backup route updating. Figure 11, shows the network end to end delay versus the network size. Based on this figure, it can be seen increase in the network size leads to decrease in the network end to end delay.Increase in the network size leads toincrease in number of backup routes as well as the network density. On the other hand, increase in the network density leads to increase the backup table accuracy.

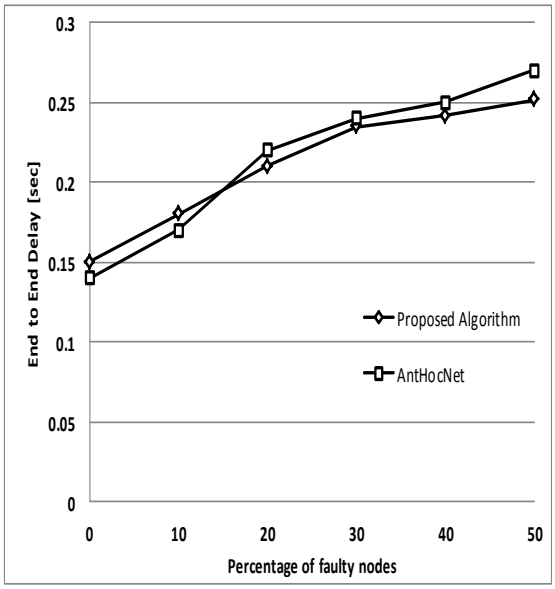

Figure 10.Effect offaultynodes on the network end to end delay.

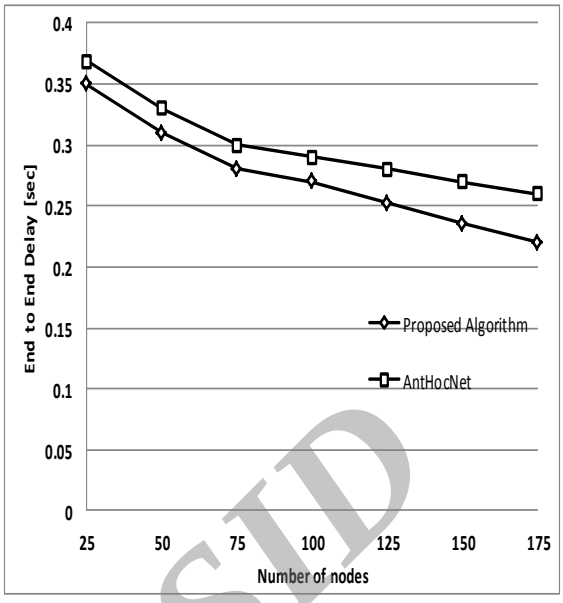

Figure 11.Effect ofthe network size on the network end to end delay.

5. Conclusions

In this paper an efficient fault-tolerant routing algorithm for MANETs is presented which increases MANET fault-tolerancethrough utilizing natural redundancy of Ad-hoc networks. The proposed algorithm is carried out in two stages; 1) the selection of backup nodes and, 2) the selection of backup route. In first stage, the proposed algorithm chooses nodes with the same movement path as backup nodes. Prediction and diagnosis of nodes means performed movement path of backup tables of mobile nodes. Since backup nodes selection is performed, the proposed algorithm begins fault-tolerance routing. For this purpose, initially the proposed algorithm creates the main route between each pair of source & destination nodes based on the DSR routing algorithm. Then backup route/routes are established between backup nodes chosen in the first stage of the proposed algorithmthrough moving from the destination node towards the source node. Experimental results taken from NS-2 simulator show that in comparison with previous methods the proposed model increases; 1) 10% the package delivery ratio against the percentages of faulty nodes and, 2) 22% package delivery ratio against the pause time of various mobile nodes.

References

[1] F.Tavakoli, M.Kamarei, GH.Asgari,"fault-tolerance enhancement in mobile Ad-hoc networks using back up nodes,"Journal of Advances in Computer Research (JACR), Vol.6 No.4 November 2015.

[2] D. A. M. D.B. Johnson, "Dynamic source routing in ad hoc wireless networks," Mobile Computing, p. 153–181, 1996.

[3] A. Avizienis, J. C. Laprie, B. Randell, and C. Lan, "Basic concepts and taxonomy of dependable and secure computing," IEEE Transactions on Dependable and Secure Computing,, vol. 1, no. 1, pp. 11-33, 2004.

[4] V. Raychoudhury, et al., "K-directory community: Reliable service discovery in MANET," Pervasive and Mobile Computing, vol. 11, pp. 140-158, 2011.

[5] K. H. a. K. M. R. Srinivasan, "Disjoint multipath routing using colored trees," The International Journal of Computer and Telecommunications Networking, vol. 51, pp. 2163-2180, 2007.
[6] K. Singh, "SURVEY ON FAULT DIAGNOSIS IN MANETs," International Journal of Software and Web Sciences (IJSWS), vol. 13, pp. 26-30, 2013.
[7] R. Ahmed, "A Fault-Tolerant Routing Protocol for Mobile Ad Hoc Networks," JOURNAL OF ADVANCES IN INFORMATION TECHNOLOGY, vol. 2, no. 2, pp. 128-132, 2011.
[8] S. R. a. M. K. P. Thulasiraman, "Disjoint Multipath Routing to Two Distinct Drains in a Multi-Drain Sensor Network," in in Twenty-sixth IEEE International Conference on Computer Communications (INFOCOM), 2007, pp. 643-651.
[9] C. Zouridaki, B. L. Mark, M. Hejmo, and R. K. Thomas, "A quantitative trust establishment framework for reliable data packet delivery in MANETs," in Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks, New York, NY, USA, 2005, pp. 1-10.
[10] Y. He, J. Pu, and Z. Xiong, "A Redundant Multipath Routing for Mobile Ad Hoc Networks," in International Multi-symposiums on Computer and Computational Sciences, 2008 , pp. 75-82.
[11] S. Y. Oh, M. Gerla , and A. Tiwari, "Robust MANET routing using adaptive path redundancy and coding," in First International Communication Systems and Networks and Workshops, Bangalore , 2009, pp. 1-10.
[12] M. K. Kavitha, K. Selvakumar, T. Nithya, and S. Sathyabama, "Zone Based Multicast Routing Protocol for Mobile Ad-Hoc Network," in International Conference on Emerging Trends in VLSI, Embedded System, Nano Electronics and Telecommunication System (ICEVENT), Tiruvannamalai , 2013, pp. 1-6.
[13] J. Zhou , Y. Lin, and H. Hu, "Dynamic Zone Based Multicast Routing Protocol for Mobile Ad Hoc Network," in International Conference on Wireless Communications, Networking and Mobile Computing, Shanghai , 2007, pp. 1528-1532.
[14] J. Zhou, S. Wang, J. Deng, and H. Feng, "ZBMRP: A Zone Based Multicast Routing Protocol for Mobile Ad Hoc Networks," Mobile Ad-hoc and Sensor Networks, vol. 3794, pp. 113-122, 2005.
[15] S. Misra, et al., "A learning automata-based fault-tolerant routing algorithm for mobile ad hoc networks," J Supercomput, vol. 62, p. 4–23, 2012.
[16] M. Khazaei and R. Berangi, "A Multi-Path Routing Protocol with Fault Tolerance in Mobile Ad hoc Networks," in Proceedings of the 14th International CSI Computer Conference (CSICC'09), Tehran, 2009, pp. 77-82.
[17] Y. Gong and A. S. Fukunaga, "Fault tolerance in distributed genetic algorithms with tree topologies," in IEEE Congress on Evolutionary Computation, Trondheim, 2009, pp. 968-975.
[18] B. J. Oommen and S. Misra, "A Fault-Tolerant Routing Algorithm for Mobile Ad Hoc Networks Using a Stochastic Learning-Based Weak Estimation Procedure," in IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'2006), 2006, pp. 31-37.
[19] R. K. Shial, K. Reddy, and K. L. Narayana, "An Optimal RPC Based Approach to Increase Fault in Wireless Ad-Hoc Network," Advances in Computer Science and Information Technology. Networks and Communications, vol. 84, pp. 372-382, 2012.
[20] Q. Yang and L. P. Kong, "A Fault-Tolerance Cluster Head Based Routing Protocol for Ad Hoc Networks," in IEEE Vehicular Technology Conference, 2008, p. 2472–2476.
[21] Z. Jipeng and X. Chao , "A Location-Based Fault-Tolerant Routing Algorithm for Mobile Ad Hoc Networks," in WRI International Conference on Communications and Mobile Computing, 2009, p. 92–96.
[22] A. Olufemi and L. Jeong, "SMiRA: A bio-inspired fault tolerant routing algorithm for MANETs," in International Conference on ICT convergence (ICTC), 2012, p. 78–84.
[23] S. Swati and S. Madhavi, "Statistical study of performance metrics of Adaptive Fault Tolerant Replication Routing Protocol for MANET," International Journal on Computer Science and Engineering (IJCSE), vol. 4, no. 11, 2012.
[24] W. Weigang , C. Jiannong , and Y. Jin Y, "A fault tolerant mutual exclusion algorithm for mobile ad hoc networks," Journal of Pervasive and Mobile Computing, vol. 4, no. 1, pp. 139-160, 2008.
[25] M. Pushpalatha, V. Revathi, and K. Rishav , "Fault tolerant and dynamic file sharing ability in mobile ad hoc networks," in IEEE International Conference on Advances in Computing, Communication and Control, 2009, pp. 474-478.
[26] K. V. a. R. Umarani, "A Fault Tolerant Power Constraint AODV Protocol for MANET," Orient. J. Comp. Sci. &Technol, vol. 5, no. 1, pp. 55-61, 2012.
[27] S. Misra, S. Dhurandher, M. Obaidat, K. Verma, and P. Gupta, "A low-overhead fault-tolerant routing algorithm for mobile ad hoc networks: A scheme and its simulation analysis," Simulation Modelling Practice and Theory, vol. 18, p. 637–649, 2010.
[28] S. Misra, S. Dhurandher, M. Obaidat, K. Verma, and P. Gupta, "A low-overhead fault-tolerant routing algorithm for mobile ad hoc networks: A scheme and its simulation analysis," Simulation Modelling Practice and Theory, vol. 18, p. 637–649, 2010.

[29] J. Vaithiyanathan and B. S. Manokar, "Robust Fault Tolerant AOMDV Routing Mechanism in MANET," Trends in Computer Science, Engineering and Information Technology Communications in Computer and Information Science , vol. 204, p. 142–150, 2011.

[30] J. Vaithiyanathan and B. S. Manokar, "Robust Fault Tolerant AOMDV Routing Mechanism in MANET," Trends in Computer Science, Engineering and Information Technology Communications in Computer and Information Science , vol. 204, p. 142–150, 2011.

[31] J. Huang and Y. Liu, "MOEAQ: A QoS-Aware Multicast Routing algorithm for MANET," Expert Systems with Applications, vol. 37, p. 1391–1399, 2010.

[32] S. Qazi, R. Raad, Y. Mu, and W. Susilo, "Securing DSR against wormhole attacks in multirate ad hoc networks," Journal of Network and Computer Applications, vol. 36, no. 2, p. 582–592, 2013.