A Critical MANET Routing Protocol

Ashraf Elgohary[1], Tarek S. Sobh[2], Sayed A. Nouh[3], Mohamed Zaki[4]
[1,2]Information System Dept, Egyptian Armed Forces, Cairo, Egypt
[3,4]Computer and System Engineering Dept, Al-Azhar University,Cairo, Egypt
ashfik2000@yahoo.com,tarekbox2000@gmail.com
sayed.nouh07@gmail.com, azhar@eun.eg

Abstract

Critical MANET environments such as military battlefields and disaster recovery operations impose a number of requirements (such as the need for robustness and performance within a high mobility scenarios), and constraints (such as Hostile attacks, RF range and cost, battery limitations). Many studies proved that PUMA (Protocol for Unified Multicasting through Announcements) is superior compared to other multicast and core-based routing protocols like DCMP, MAODV and ODMRP where in addition to providing the lowest control overhead compared to ODMRP and MAODV, PUMA provides a very tight bound for the control overhead [1]. So we studied PUMA and its drawbacks while working in critical MANETs which leaded us to propose a novel routing protocol named Adaptive Secure Headship Following Induction Keeping (ASHFIK) to work ideally in critical MANETs scenarios.In this paper we discuss the structure of ASHFIK which uses a new mechanism called the Headship Mechanism which provides always a standby core to work if the original core is down. After studying performance analysis of ASHFIK compared to PUMA, we could conclude that ASHFIK is suitable for most critical MANETs scenarios.

Keywords: *Critical MANET, Ant based routing, ASHFIK, multicasting, Ant Colony Optimization, Headship Mechanism*.

1. Introduction

A MANET (Mobile Ad hoc Network) is a collection of wireless nodes (like laptops or PDAs) that can dynamically form a network to exchange information without using any pre-existing fixed network infrastructure. This is very useful in critical situations like military battlefield (sensitive information exchange) and disasters recovery (fire/safety/rescue operations).

There are some important characteristics when we choose the suitable routing protocol for critical MANETs like: **multicast, multipath** capabilities.

In real world critical MANETs (military, law enforcement, rescue operations), each group consists of members and group leader that move in most cases inside hostile environments.

Military deployments (for example) have a well defined chain of command. While not suggesting that communications must strictly follow that chain of command, a chain of command will always exists, and in general, the nodes are physically located according to that model. This has an effect on how ad hoc network topologies actually form both initially and subsequently throughout the operation [2].

The group leader has many essential characteristics all times like:

- Collecting data from the group members and any communications among group members must be done through him.
- Analyzing data and make decisions and pass them to group members.
- Ability to pass orders to one or many members of the group.
- Strong stability from failure or attacking, noticing that the group must has a leadership mechanism in case that the communication with the group leader is dead to select the next leader in command order to be the new group leader.

Although many core based routing protocols appeared in the last years, they usually concentrated upon improving the performance not to satisfy the above requirements for critical MANETs which meet several challenges like:

- Heterogeneous mobility (low and high velocity),
- Tactical and hostile areas (constrained areas that may be divided into sub areas and Enemy attacks are possible),
- Optimal paths (short paths are not always trusted paths),
- Obstacles (dealing with obstacles effects is a must),
- Units join and leave the scenario (either for damages or for other reasons),
- Group movement (group leader controls the behaviors of the other nodes).

We tried in our work to apply these requirements and to propose an efficient and dependable routing protocol for critical MANETs.

This paper is organized as following: section 2 illustrates the problem statement and our objectives, section 3 refers to some related works to the target of this paper, section 4 presents ASHFIK framework and its components, section 5 defines the simulation environment, section 6 explains dependability (availability, reliability) analysis, section 7 discusses performance analysis and finally section 8 notifies conclusions.

2. Problem Statement

Critical MANETs have many challenges must be considered when designing routing protocols like: high mobility, obstacles existence, limited resources and hostile environments (malicious nodes, active and passive attacks).

For a critical MANET example we assume a group consists of a number of persons (between twenty and fifty), which will move across an area in a hostile environment.

For our scenario where there is a small sized network with moving nodes at high speed and high demands on data delivery, a ***reactive*** protocol with flat ***architecture*** and multicast capabilities and a ***mesh-based*** structure is the best choice. The protocol should be able to send information through ***multiple paths*** to ensure the high throughput of the network [3]. In situations that demand a high quality of service, a protocol that ensures the quality of the network is of great importance.

The proposed protocol must also support multicasting (a source sends data to many destinations simultaneously), this service is important in critical MANETs due to the requirements for audio and video conferencing and sharing of text and images. Multicasting reduces the communication costs for applications that send the same data to multiple recipients and minimizes the link bandwidth consumption, delivery delay, sender and router processing [4].

In critical MANETs, when the core is down a process (core selection and migration) is done to select another core and inform the other nodes in the group. The problem here is how to select a trusted core since the group nodes depend on the trusted core to exchange sensitive data inside the critical MANET.

3. Related Work

Multicast protocols that work for mobile adhoc networks can be classified to: (i) tree-based like (MAODV [5]), (ii) mesh-based like (ODMRP [1], PUMA [6]) and (iii) hybrid like (MCEDAR [7]).

In this section we will explain the Protocol for Unified Multicasting through Announcements (PUMA) since it was superior when compared to MAODV and ODMRP [1, 6]. Also, we will discuss two examples of ant based multicast routing protocols which are used in MANETs.

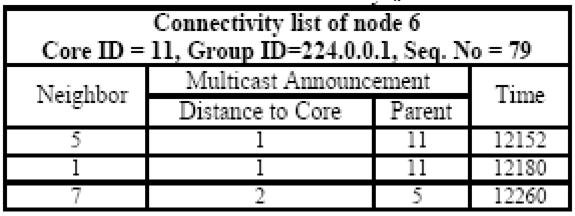
3.1. PUMA Protocol

The Protocol for Unified Multicasting through Accouchements (PUMA) [6] establishes and maintains a shared mesh for each multicast group without depending upon a unicast routing protocol.

In PUMA, any source can send multicast data to a multicast group without having to knowing the constituent members of the group. Moreover source does not require joining the group to dispatch the data. Also, PUMA is a receiver initiative approach where receivers join the multicast group using the address of a special core node without the need for flooding of control packets from the source of the group. It makes the use of dynamic cores (not pre-assigned).

When multicast announcement propagates through the network it establishes a connectivity list at every node in the network and helps nodes to build the mesh (Table 1). Each node uses core ID, group ID, sequence number, distance to core, parent as fields in multicast announcement. There may be multiple routes to the core. But if core is changed, all nodes have to rebuild their connectivity lists.

Table 1 - Connectivity list



Connectivity list of node 6 Core ID = 11, Group ID=224.0.0.1, Seq. No = 79			
Neighbor	Multicast Announcement		Time
	Distance to Core	Parent	
5	1	11	12152
1	1	11	12180
7	2	5	12260

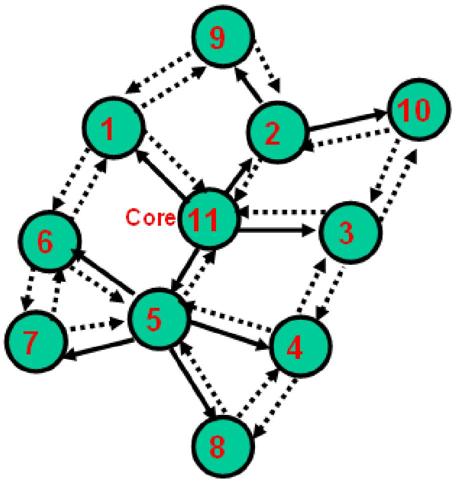


Fig. 1 - The propagation of multicast announcement.

When a receiver wishes to join a multicast group, it first determines whether it has received a multicast announcement for that group before. If the node knows the core, it starts transmitting multicast announcements and specifies the same core for the group. Otherwise it considers itself the core of the group and starts transmitting multicast announcements periodically to its neighbors stating itself as the core of the group. Node propagates multicast announcements based on the best multicast announcements it receives from its neighbors. A multicast announcement with higher core ID nullifies the announcement of a lower core ID. So, each connected component has only one core. If more than one receiver joins the group simultaneously, then the one with the highest ID becomes the core of the group.

As a rule, for the same core ID, only multicast announcements with the highest sequence number are considered valid. For the same core ID and sequence number, multicast announcements with smaller distances to the core are considered better. When all those fields are the same, the multicast announcement that arrived earlier is considered better.

From Table 1, Node 6 has three entries in its connectivity list for neighbors 5, 1, and 7. However it chooses the entry it receives from 5 as the best entry, because it has the shortest distance to core and has been received earlier that the one from node 1.

As shown in Fig. 1, If Node(6) wants to send a data packet to Node(2), it will choose its neighbor 5 to reach the core 11 then the core will forward the data packet to Node(2), The path will be 6-5-11-2**.**

PUMA protocol has many drawbacks with critical MANETs such as the method of selecting a core in PUMA [8] has a serious drawback that any node inside the group can be selected to be the core (there is no trusted core). Another drawback when a core is down is the selecting and migrating to a new core may cause a certain delay (even it is small delay) which is not accepted in critical MANETs like our scenario.

3.2. Ant Based MANET Multicast Routing

Ant Colony Optimization (ACO) is a famous swarm intelligence approach [9], initially proposed by Marco Dorigo in 1992 in his PhD thesis. He was aiming to search for an optimal path in a graph, by simulating the behavior of real ants seeking for the shortest path between their colony and a source of food. While real ants move, they put a chemical volatile substance called 'pheromone' and they select their next hop based on the amount of pheromone deposited on the path to the next node.

ACO was the inspiration for developing many routing protocols for MANETs, where ants are used as agents in which are divided into forward and backward ants. The sender to the neighbor nodes broadcasts the forward ants. The backward ants utilize the useful information like end-to-end delay, number of hops gathered by the forward ants on their trip from source to the destination.

However, there are two key issues that have not been solved yet: (i) The typical ACO algorithms (e.g., AntHocNet [10]) incorporate both reactive route setup and proactive route improvement/maintenance, and incur more control overhead than MANET protocols, (ii) ACO algorithms usually do not consider obstacles effects in routing protocol design. Although ACO routing protocols generally achieve higher throughput than MANET protocols, these two drawbacks make them less applicable for critical MANETs.

3.2.1. Multicast for Ad hoc Network with hybrid Swarm Intelligence protocol (MANHSI)

MANHSI [11] is an on demand multicast core based routing protocol that creates a multicast mesh shared by all the members within the group. Unlike other core-based protocols, MANHSI does not always depend on the shortest paths between the core and the group members to build group connectivity. Instead, each member who is not the core periodically deploys a small packet (FORWARAD ANT) that behaves like an ant to discover paths that include a better set of forwarding nodes producing a lower total cost of data forwarding noticing that the cost is considered on a per-node basis, not per-link.

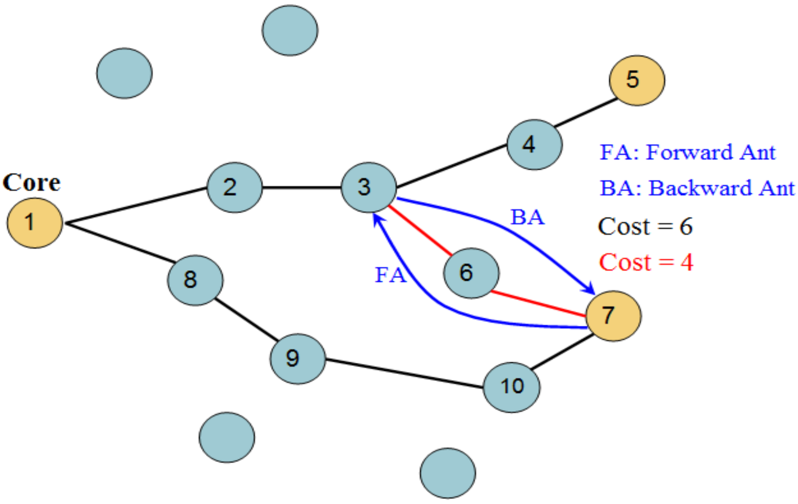


Fig. 2- Behavior of forward and backward ants in MANHSI.

As in Fig. 2, a FORWARD ANT deployed from the Node 7 choosing Node 6 as the next hop and discovering a forwarding Node 3, and at Node 3, the FORWARD ANT becoming a BACKWARD ANT and following the reverse path back to Node 7 while depositing pheromone along the way. This will reduce the total cost for sending data from Node 1 (core) to Node 5 and Node 7, from a total cost=6 (number of intermediate nodes) to a total cost=4.

We noticed that MANHSI uses the ACO concept to quickly and efficiently establish initial multicast connectivity paths and/or dynamically to improve the resulting connectivity rather than selecting trusted cores for critical MANETs.

3.2.2. Ant Based Adaptive Multicast Routing Protocol (AAMRP)

AAMRP is an ant agent based adaptive multicast protocol which combines between multicasting and broadcasting and dynamically organizes the group members into clusters where one of the group members is selected to be a cluster leader. Cluster leaders have two main functions: [12]

- They establish a sparse multicast structure among themselves and the source, and
- They adaptively use broadcasting to deliver the packets to other group members in their cluster.

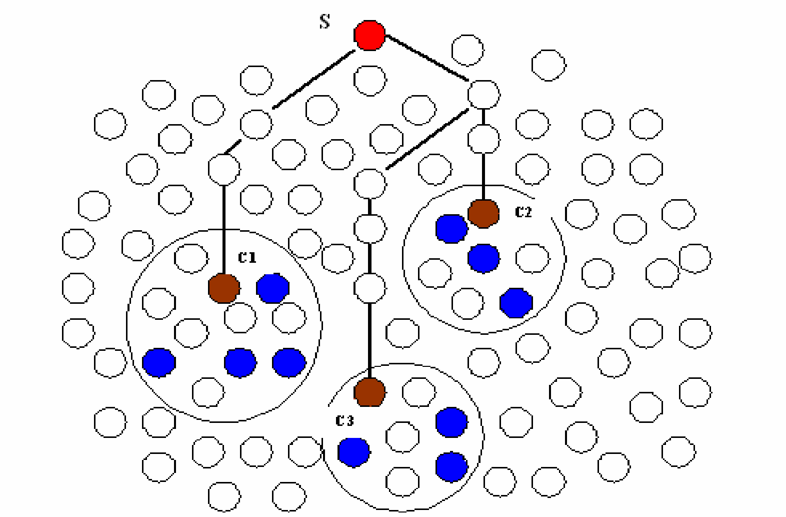


Fig. 3 - Multicast Architecture of AAMRP. [12]

As in Fig. 3, a multicast source **S** sends data to three clusters with cluster leaders **C1**, **C2**, **C3** respectively, then each cluster leader simply invokes an adaptive localized broadcast within its cluster to disseminate multicast packets received from the source **S**. This would decrease the consumed overhead while providing efficient data delivery.

We noticed that at the leader election phase of AAMRP, the joining node elects itself as the cluster leader for its k-hop neighborhood, if it cannot still find any cluster leader in its vicinity, after the discovery phase. This means that any node could be a cluster leader which is inappropriate for critical MANETs (like our scenario).

4. The Proposed ASHFIK Routing Protocol

The term ASHFIK (Adaptive Secure Headship Following Induction Keeping) is proposed to make the routing protocol always maintain the group communications and ***adapts*** the case of core node failure to ***secure keep*** using the ***headship following induction*** (like headship mechanism in military scenarios) and finally there will be always trusted core nodes ready to take place the original core if it down.

4.1. ASHFIK Framework

Since a major requirement in critical MANETs is to overcome the obstacles existence, we need to choose a mobility model that can work well within obstacles environment.

MANET environment may contain unpredictable obstacles, such as mountains, lakes, buildings, or regions without any hosts, impeding or blocking message relay. The obstacles restrict not only the nodes movement but may obstruct the effective transmission paths between nodes [13].

The obstacles like rivers or lakes affect the node movement only but do not reduce the effective transmission range of nodes, while the obstacles such as mountains restrict both node movement and effective transmission paths between nodes.

At 2009, Papageorgiou [14] proposed a very good mobility model called Mission Critical Mobility (MCM). The main characteristics of the MCM model with which the real-life properties of movement in such environments are captured, are the presence of physical obstacles that affects both the node movement and the signal propagation.

We use the TerGen java program [15] to build obstacles environment.

The output from above is Obstacles.txt file which is used as input for Mission Critical Mobility (MCM) to build different mobility scenarios.

ASHFIK agent includes as a major part, the Headship Mechanism which is explained in the next section.

We create a trace analyzer program to analyze the output trace files and build statistics graphs and the final ASHFIK framework is shown in Fig. 4.

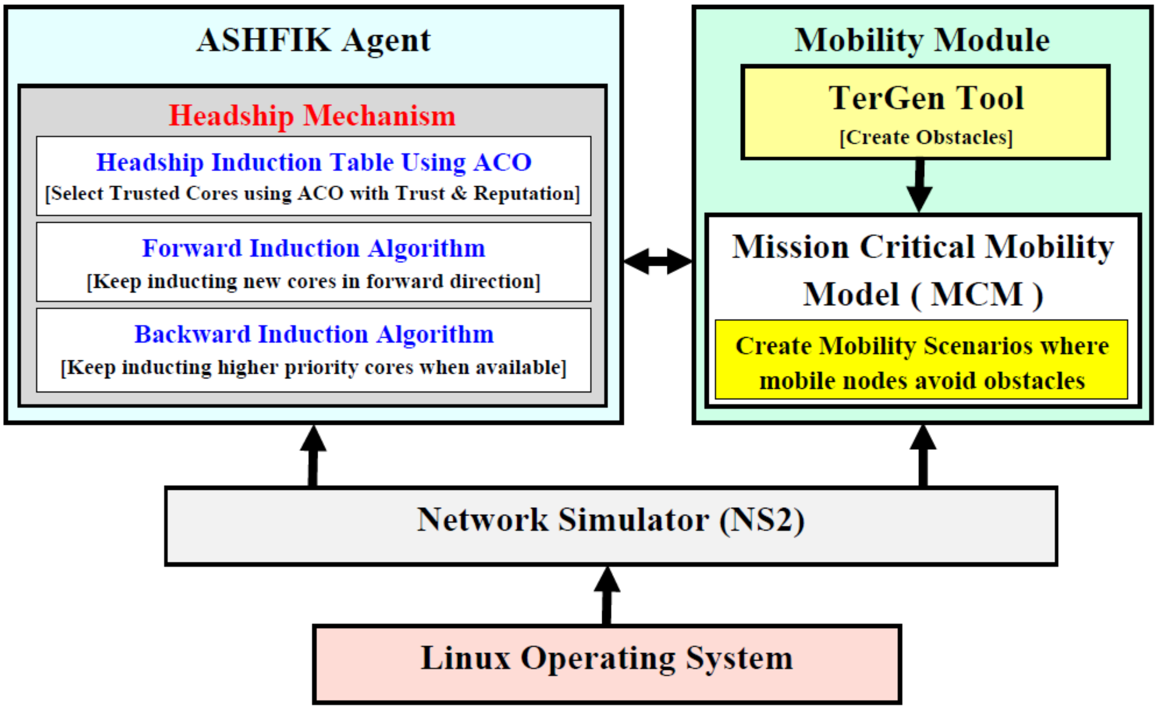

ASHFIK Agent
Headship Mechanism
Headship Induction Table Using ACO
[Select Trusted Cores using ACO with Trust & Reputation]
Forward Induction Algorithm
[Keep inducting new cores in forward direction]
Backward Induction Algorithm
[Keep inducting higher priority cores when available]
Mobility Module
TerGen Tool
[Create Obstacles]
Mission Critical Mobility Model (MCM)
Create Mobility Scenarios where mobile nodes avoid obstacles
Network Simulator (NS2)
Linux Operating System

Fig. 4 - ASHFIK Framework

4.2. The Headship Mechanism

The most important part in ASHFIK is the Headship Mechanism which is responsible selecting a specific number of nodes inside the group to and arranging them with highest priorities to be trusted cores and ready to take place the current core if it is down.

The Headship Mechanism consists of two algorithms: the forward induction keeping and backward induction algorithms (will be discussed later).

Also the Headship Mechanism uses two variables:

- **-the headship_id:** a number is assigned to identify the node's headship priority.
- **-the max_headship_id**: a number defines the total number of nodes allowed to be trusted cores.

4.2.1. Headship Induction Table Using ACO with Trust and Reputation

Félix Gómez [16] described trust and reputation models used in Mobile ad-hoc networks (MANETs) like Robust Reputation System (RRS) [17] and Pervasive Trust Management (PTM) [18] which were approved as a main approach to improve efficiency and security matters in MANETs.

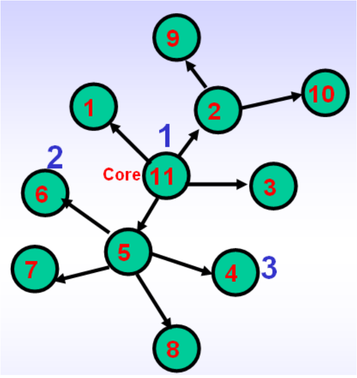
From ASHFIK point of view, the definition of trust is the probability with which an agent will performa particular action assigned to him even if this will affect upon his own action. Equally, the definition of reputation is the expectation about an agent's behavior based on information about it or observations of its past behavior.

In the trust based protocols the behavior of each node in the network is observed by the other nodes of the network and a trust index is developed about the node under observation. This trust index developed by a node about other nodes of the network can either be due to its own observation (first hand trust) or may be due to the opinion given by the other nodes (second hand trust) [19].

In real critical MANETs the Commander (group leader) chooses the following members in leadership according to many conditions like (previous experience, good communication resources, etc.) while the method of creating the headship induction table in ASHFIK is based on the concept of integration of ant colony optimization (ACO) with trust and reputation models, where the leader ant (group leader) collects trust and reputation information about the other surrounding nodes.

For our work, two fundamental parameters (trust value and reputation value) are used. The nodes with highest trusted values are used to build the headship induction table and the high value of reputation of a node signifies that the node is trusted and is more reliable for data communication purposes. As a node shows signs of misbehavior, its reputation decreases, which affect its quality-of-security (QSec), thereby disabling the malicious nodes from gaining access to the network [20].

At the beginning, the group leader assigns to himself the headship_id = 1, and increments the max_headship_id by 1.Then the group leader will prepare the headship following induction table which contains the nodes following him in command based upon the trust values mentioned before, then floods the network with this table.



Node id	headship_id	max_headship_id
11	1	3
6	2	3
4	3	3

Fig. 5 - The Headship Induction Table.

As shown in Fig. 5, Headship Induction Table tells that, Node (11) is chosen to be the primary group core, and Node(6) and Node(4) are ready to take the group leadership respectively if Node(11) is failed.

4.2.2. Forward Induction Keeping Algorithm

As shown in Fig. 6, the current headship core floods the network with data, if the group nodes do not receive any packets from the headship core for N= 3 seconds, the forward induction keeping algorithm increments the headship_id value by 1 and gets its corresponding node id from the headship induction table.

Then, it inducts this node as a new headship core if this node is available, otherwise incrementing the headship_id value again and test the next trusted node. If the headship_id reached the max_headship_id, the headship_id is reset to one.

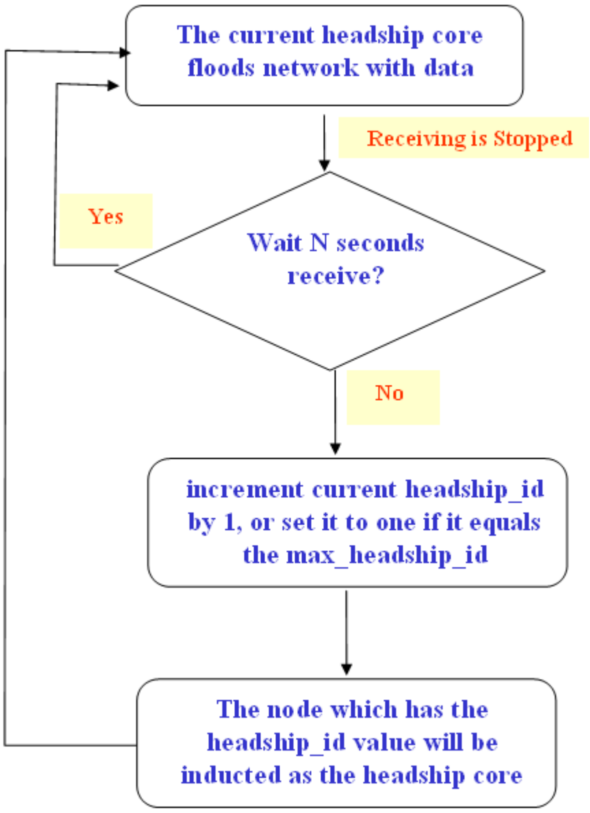

The current headship core floods network with data
Receiving is Stopped
Yes
Wait N seconds receive?
No
increment current headship_id by 1, or set it to one if it equals the max_headship_id
The node which has the headship_id value will be inducted as the headship core

Fig. 6 - Forward Induction keeping.

4.2.3. Backward Induction Algorithm

As shown in Fig. 7, at any time, if a superior headship core node gets connection again with the group (where N= 3 seconds), the backward induction algorithm forces the group to induct this old core to be the current headship core. If two or more superior trusted cores enter again the group, the backward induction algorithm inducts the node with smallest headship_id is as the new headship core.

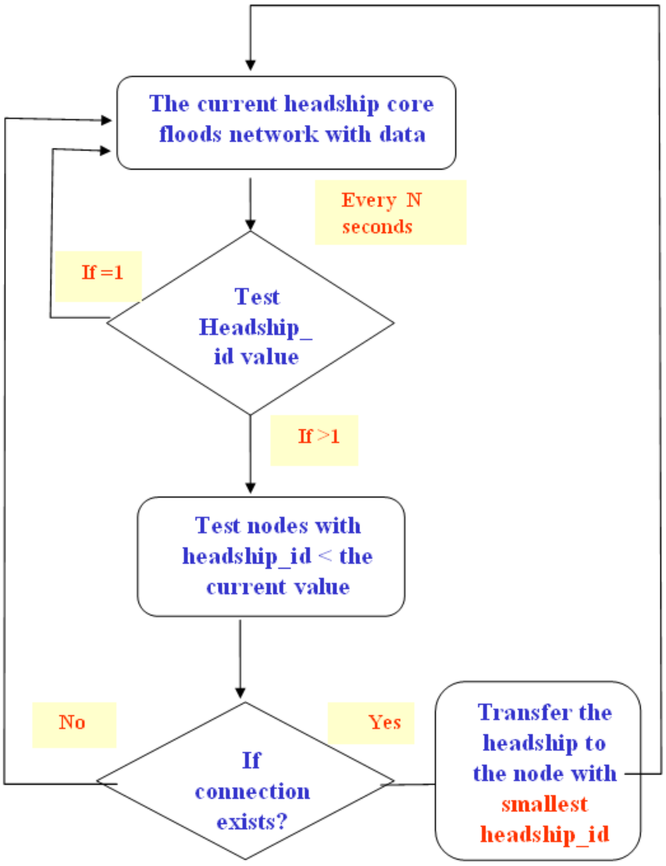

The current headship core floods network with data
Every N seconds
If =1
Test Headship_ id value
If >1
Test nodes with headship_id < the current value
No
Yes
If connection exists?
Transfer the headship to the node with smallest headship_id

Fig. 7 - Backward Induction.

4.3 MANET Scenario under Using ASHFIK

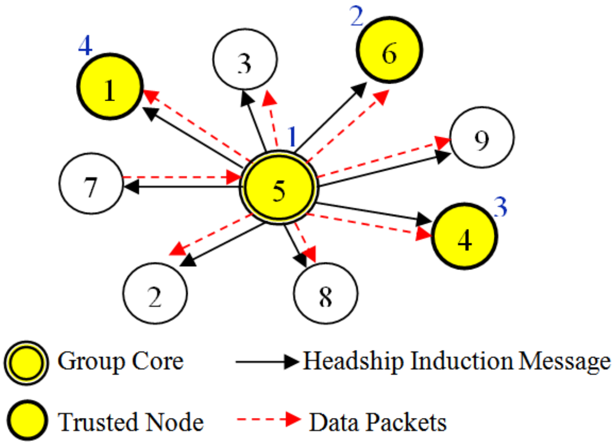


Fig. 8: - ASHFIK Messages Flow

Fig. 8 explains ASHFIK messages flow where the group core sends headship induction (HI) messages to inform the nodes with the trusted nodes that will take place the group core when needed (Nodes 6,4,1 are trusted nodes with priority values 2,3,4 respectively). Node 7 wants to send data to the group nodes so it will send data packets to the group core (Node 5) which will forward the data packets using multicasting approach to the other group nodes.

The algorithm of the proposed protocol can be described as follows:

```
Step 1: At the beginning, the group core identify trusted alternative cores,
Step 2: Inform the group nodes with the trusted alternative cores information,
Step 3: While the group core is alive {
            Receive data packets from senders and forward them to receivers
  }
            Else {
                  Select a trusted node to work as the group core,
            Send control messages to inform the group nodes with the new core information,
            If any time, the original core is back alive it takes the headship again.
      }
```

4.4. Group Division and Union using ASHFIK

Due to the nature of critical MANETs, for many circumstances (like meeting obstacles) the group nodes are divided into subgroups which may be isolated from the group core.

ASHFIK can deal well with this situation, each subgroup will search about the next trusted core using the forward induction algorithm until they meet a core with higher headship_id, and then the backward induction algorithm is used for subgroups union.

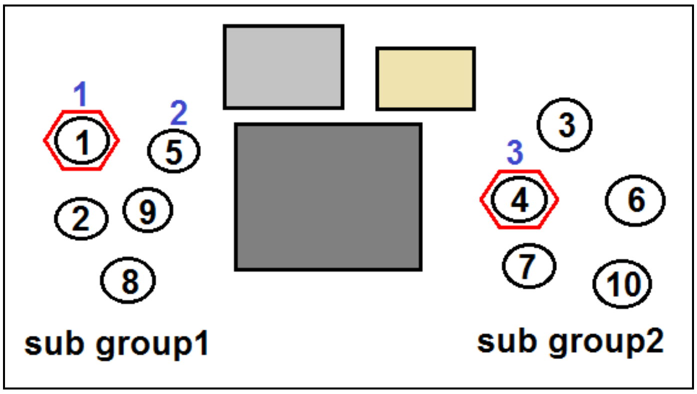


Fig. 9 - Group division using ASHFIK.

As shown in Fig. 9, a group of 10 mobile nodes meets 3 obstacles which cause the division of the group into two subgroups. The members of sub group2 are isolated from the group core (Node 1 with headship_id =1) so they search among them for the next trusted core which will be (Node 4 with headship_id =3).

When they can communicate with Node 1, the backward induction algorithm is used to unite sub group2 with sub group1.This makes ASHFIK very useful for critical MANETs situations.

5. Simulation Environment

We used the network simulator NS2.35 under CYGWIN for testing ASHFIK routing protocol. The results were visualized using NAM animator.

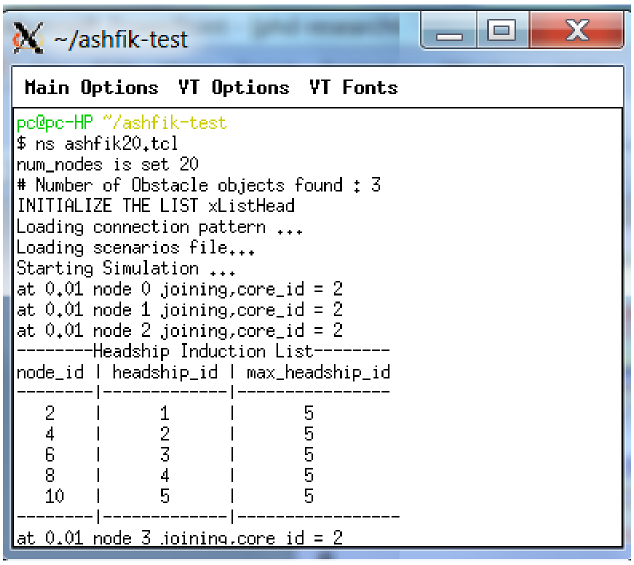
5.1. ASHFIK Simulation with NS2

We tested ASHFIK relative to varying the group size starting with 5,10,20,50,100 mobile nodes are moving in area 500X500 meters and area includes three large obstacles. Table 2 lists the values of the common parameters used in all the experiments.

Table 2 Simulation parameters

Parameter	**Value**
Nodes	5,10,20,50,100
Simulation time	150 sec
Mobility Model	Mission Critical Mobility model
Packet size	512 bytes
Simulation area	500 m X 500 m
No. of obstacles	3

As shown in Fig. 10, a group of size 20 mobile nodes and 5 trusted nodes, under testing. The Headship Induction List is constructed at the beginning of the simulation.

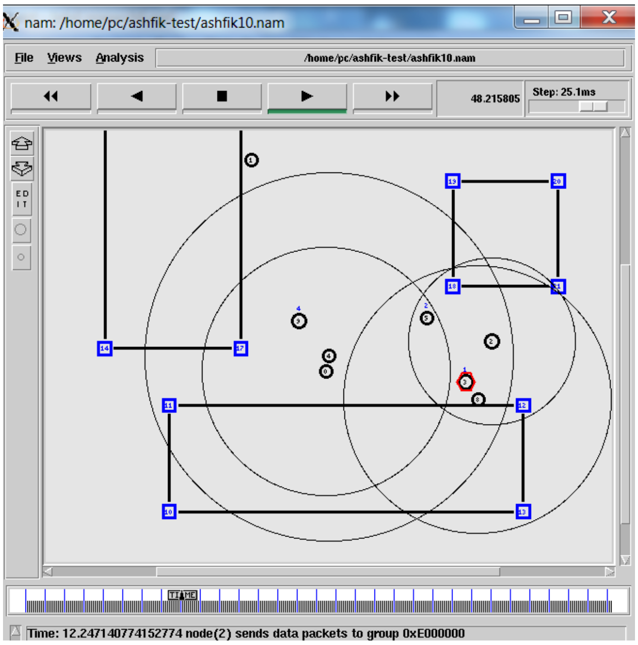


```
pc@pc-HP ~/ashfik-test
$ ns ashfik20.tcl
num_nodes is set 20
# Number of Obstacle objects found : 3
INITIALIZE THE LIST xListHead
Loading connection pattern ...
Loading scenarios file...
Starting Simulation ...
at 0.01 node 0 joining,core_id = 2
at 0.01 node 1 joining,core_id = 2
at 0.01 node 2 joining,core_id = 2
--------Headship Induction List--------
node_id | headship_id | max_headship_id
--------|-------------|----------------
   2    |      1      |       5
   4    |      2      |       5
   6    |      3      |       5
   8    |      4      |       5
   10   |      5      |       5
--------|-------------|-----------------
at 0.01 node 3 joining,core_id = 2
```

Fig. 10 - NS2 layout.

5.2. ASHFIK Animation with NAM

As shown in Fig. 11, a Critical MANET consists of 10 mobile nodes moving in an obstacle environment is visualized with NAM. The current core (group leader) is surrounded by a red hexagon.



Time: 12.2471407741527 74 node(2) sends data packets to group 0xE000000

Fig. 11 - A group of 10 nodes presented in NAM.

We noticed that the group nodes are very clever in avoiding obstacles using ASHFIK with MCM mobility model.

6. Dependability (Reliability and Availability) Analysis

Dependability of a MANET may be defined as the trustworthiness of the MANET, which allows reliance to be justifiably placed on the service it delivers. It is an integrative concept that combines attributes like availability (readiness of correct usage) and reliability (continuity of correct service). [21]

The proposed ASHFIK ensures at any time (using the Headship Mechanism) the existence of **available**, **reliable** core for group communications using the forward and backward induction algorithms.

6.1. Reliability Analysis

It can be defined as the between the number of nodes that receive the sources transmission to total number of nodes in the group.

$$\textbf{Reliability} = \frac{\textbf{Number of nodes that receive the source's transmission}}{\textbf{Total number of the group nodes}}$$

Experiment:

Changing the group size (5, 10, 20, 50,100 nodes)

Result: (Fig. 12)

The reliability of ASHFIK is high even when increasing the group size and number of senders.

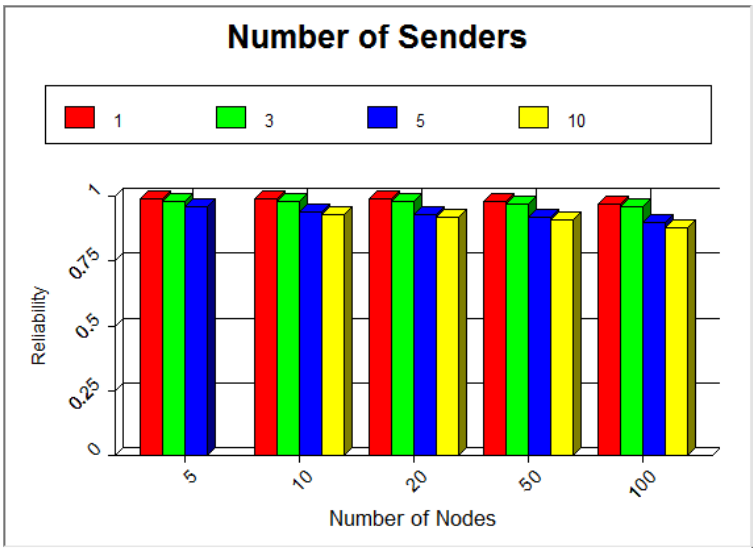


Fig. 12 - ASHFIK Reliability as group size increases.

Note that, as increasing the number of senders and due to the mobility of the nodes the reliability is decreasing with small affordable value.

6.2. Availability Analysis

It can be defined as the ratio of the expected value of the uptime of a system to the aggregate of the expected values of up and down time.

$$A = \frac{E[\text{Uptime}]}{E[\text{Uptime}] + E[\text{Downtime}]}$$

The downtime is the time when a node (or more) in the group lost the connection with group core until the connection is back or the next core in the headship induction list takes place the original core.

Experiment:

Changing the group size (10, 20, 50,100 nodes) and the simulation period is 150 seconds.

Result: (Fig. 13)

The reliability of ASHFIK is high even when increasing the group size.

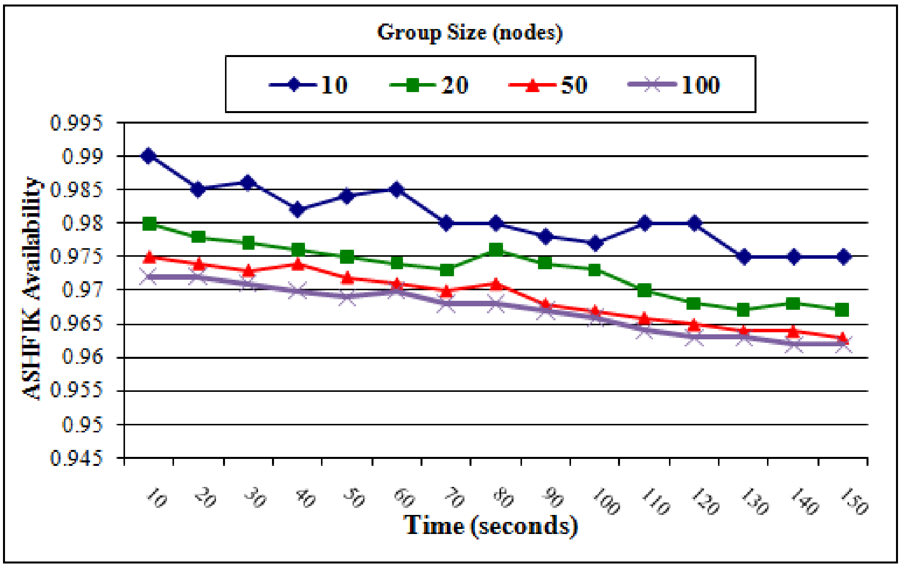


Fig. 13 - ASHFIK Availability as group size increases.

Note that as the group size is small the nodes are closed to each and the group core can reach all nodes, while as the group size increases the nodes are stretched and the probability that some nodes are isolated from the group core increases.

7. Performance Analysis

We decided to compare between our proposed protocol ASHFIK and PUMA, AAMRP, MANSHI protocols as they all work for MANETs and they are multicast and ACO based protocols (specifically AAMRP, MANSHI).

7.1. Packet Delivery Ratio (PDR)

We tested the Packet Delivery Ratio (PDR) of ASHFIK under several circumstances as shown below giving that:

$$\mathbf{PDR} = \frac{\textbf{total packets received}}{\textbf{total packets sent}} \times \mathbf{100}$$

Experiment (1):

Changing the group size (5, 10, 20, 50, 100 nodes) and the mobility (speed) is 4 m/sec.

Result: (Fig. 14)

ASHFIK is better than others.

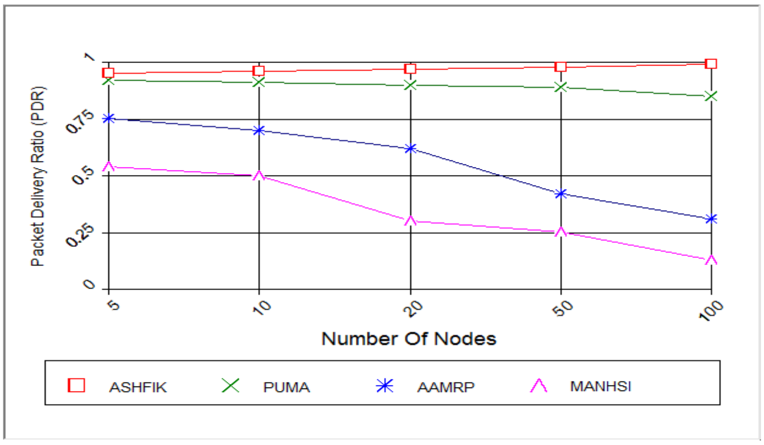


Fig. 14 - Packet Delivery Ratio (PDR) with respect to group size.

Experiment (2):

Changing the mobility (2, 4, 6, 8, 10 m/sec) and the group size is 20 mobile nodes.

Result: (Fig. 15)

ASHFIK is better than others.

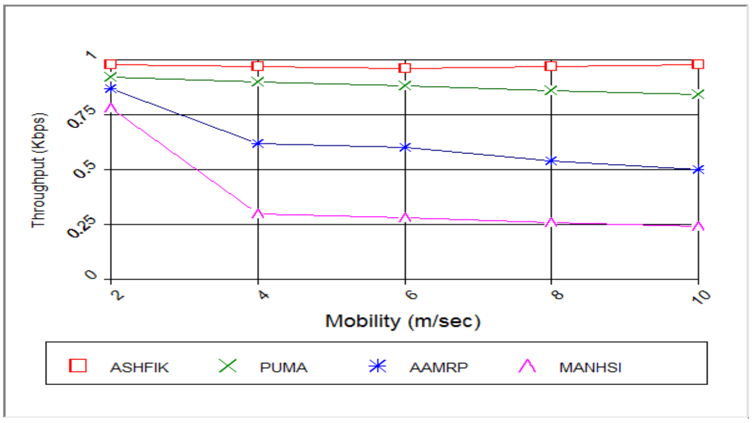


Fig. 15 – Packet Delivery Ratio (PDR) with respect to mobility.

7.2. Throughput

Throughput refers to how much data can be transferred from the source to the receiver(s) in a given amount of time:

$$\textbf{Throughput} = \frac{\textbf{Number of packets sent}}{\textbf{Time Taken}}$$

Experiment:

Changing the group size (5, 10, 20, 50, 100 nodes)

Result: (Fig. 16)

ASHFIK is better than others.

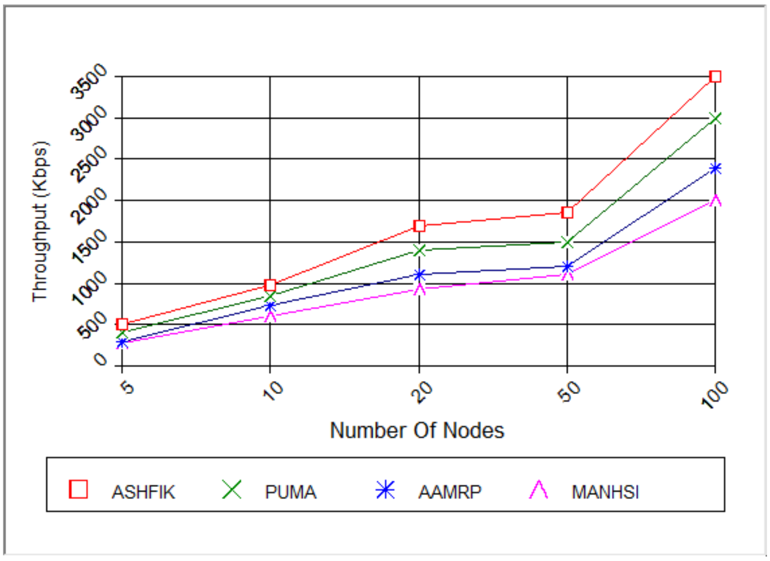


Fig. 16 - Throughput for ASHFIK vs. others.

7.3. Total Overhead

The total packets transmitted is the sum of control packets + data packets

$$\textbf{Total overhead} = \frac{\textbf{total packets transmitted}}{\textbf{data packets delivered}}$$

Experiment:

Changing the group size (5, 10, 20, 50, 100 nodes)

Result: (Fig. 17)

ASHFIK is better than others.

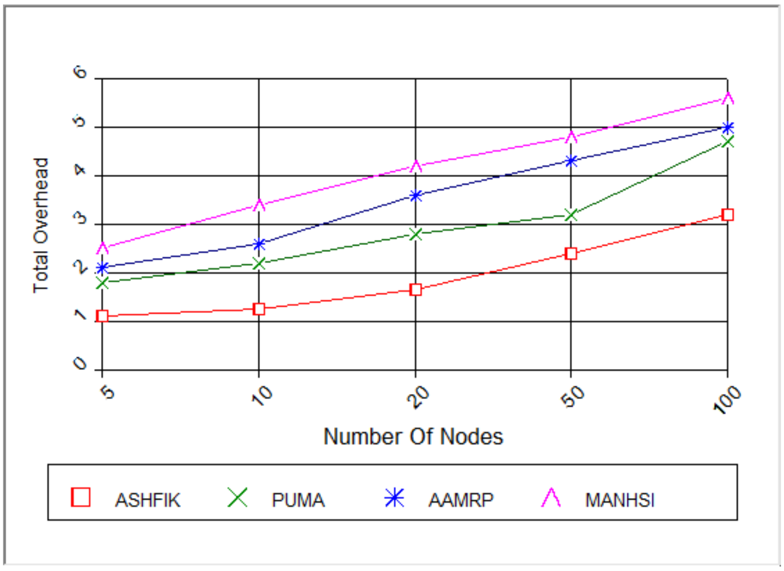


Fig. 17 - Total Overhead for ASHFIK vs. others

7.4. Average End-to-End Delay (second)

It is defined as the average time taken for a data packet to be transmitted across a MANET from source to destination.

If Tr is receive Time and Ts is sent Time then, for each packet: D = (Tr –Ts) and

$$\textbf{Avg. EED} = \frac{\textbf{SUM (D)}}{\textbf{Number of packets sent}}$$

<u>*Experiment:*</u>
Changing the group size (5, 10, 20, 50, 100 nodes)

<u>Result: </u>(Fig. 18)
ASHFIK is better than others.

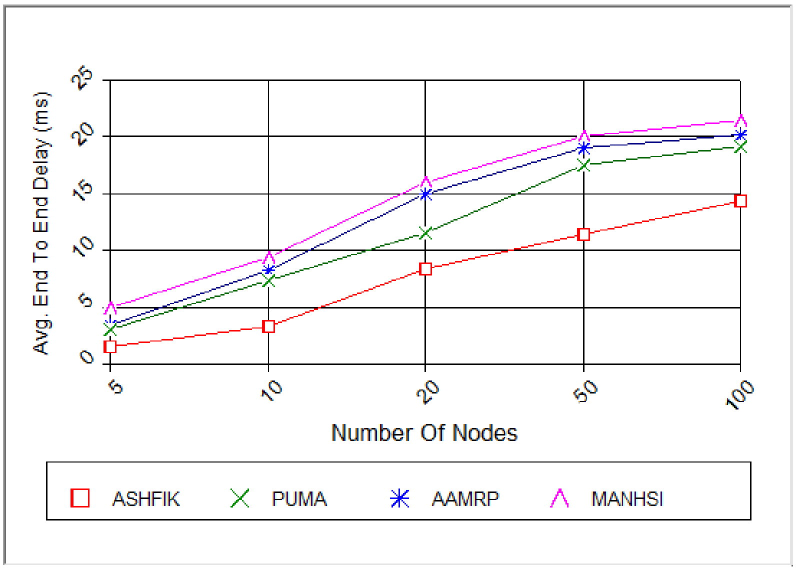


Fig. 18 - Average End TO End Delay for ASHFIK vs. others.

7.5. Normalized Routing Load (NRL)

It is defined as the ratio of total no. of data packets received to the total no. of routing packets received:

$$\textbf{NRL} = \frac{\textbf{Number of data packets received}}{\textbf{Number of routing packets received}}$$

<u>*Experiment:*</u>
Changing the group size (5, 10, 20, 50, 100 nodes)

<u>Result: </u>(Fig. 19)
ASHFIK is better than others.

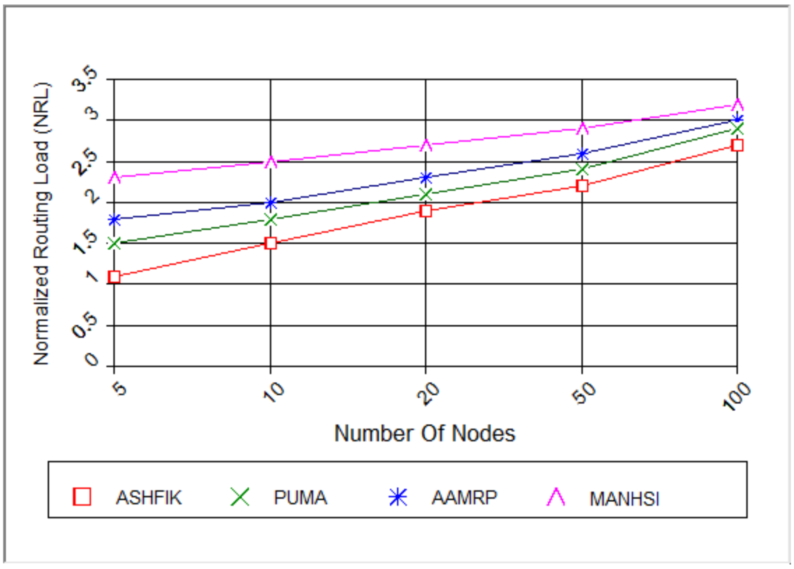


Fig. 19 - Normalized Routing Load (NRL) for ASHFIK vs. others.

8. Conclusions

Now, from the above performance analysis we can conclude that ASHFIK is much better than PUMA for our critical MANET scenario. The Headship Mechanism which is included in our proposed ASHFIK protocol was very useful in enhancing the ASHFIK performance comparing with PUMA noticing that ASHFIK overcomes the lack of security in PUMA [8]. Also, ASHFIK is very reliable in obstacles environment and ASHFIK overcomes the drawbacks in PUMA so it is very suitable for Critical MANETs.

The feature of group division and union in ASHFIK is useful for critical MANETs scenarios, where some of the group nodes are isolated from the group for a period of time. In such a situation the isolated sub group (using Forward and Backward induction algorithms) searches for the next trusted core inside it and considers it the current core until the sub group returns back to the main group.

During our work we concentrated upon how to make ASHFIK satisfy critical MANETs requirements like high mobility, obstacles avoidance and trusted group communications. ASHFIK can be tested in future under heavy transmission of stream audio and video files and observing statistics. Also, ASHFIK can be extended in future to work with other kinds of networks like satellite and cellular (4G technology) networks.

References

[1] S. Sumathy, Beegala Yuvaraj, and E Sri Harsha,"Analysis of Multicast Routing Protocols: Puma and Odmrp", *International Journal of Modern Engineering Research (IJMER), Vol.2, Issue.6, Nov-Dec. 2012 pp-4613-4621.*

[2] Peter Holliday," SWARMM - A Mobility Modeling Tool For Tactical Military Networks" *, Military Communications Conference, 2008. MILCOM 2008, E-ISBN: 978-1-4244-2677-5, IEEE.*

[3] Rohit Jain, Abhinav Mehta, and Vinay Somani, "Performance Evaluation of Fault Tolerance Protocols in MANET", *International Journal of Computer Applications (IJCA), Volume 61- No.2, 2013.*

[4] Sumathy S , Sri Harsha E , and Yuvaraj Beegala," Survey of Genetic Based Approach For Multicast Routing In MANET", *International Journal of Engineering and Technology (IJET), Vol. 4,No 6, Jan 2013.*

[5] Rajneesh Gujral, Sanjeev Rana, and Amrita Chaudhary, "Study and Comparison of Mesh and Tree-Based Multicast Routing Protocols for MANETs", *International Journal of Latest Trends in Engineering and Technology (IJLTET), Vol. 1 Issue 2 July 2012.*

[6] Ravindra Vaishampayan and J.J. Garcia-Luna-Aceves, "Efficient and Robust Multicast Routing in Mobile Ad Hoc Networks", *In Proceedings of 2004 IEEE International Conference on Mobile Ad-hoc and Sensor Systems.*

[7] Pavan Pichka, H.Santhi, N.Jaisankar, and Devi Priya," A Comprehensive Study of Existing Multicast Routing Protocols Used In Mobile Ad Hoc Networks ", *International Journal of Engineering Science and Technology (IJEST), Vol. 4, No.05, May 2012, pp. 2058-2071.*

[8] A.Amuthan, D.Nagamani Abirami,"Multicast Security Attacks And Its COUNTER Measures For PUMA Protocol", *Int. J. Comp. Tech. Appl., Vol 2 (3), pages: 594-600 ,2011.*

[9] M. Dorigo , C. Blum, "Ant colony optimization theory: A survey", *Theoretical Computer Science (Elsevier) , vol. 344, no. 2-3, pp 243–278, 2005.*

[10] G. D. Caro, F. Ducatelle, and L. Gambardella, "AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks", *European Transactions on Telecommunications, vol. 16, no. 5, 2005.*

[11] Zeyad M. Alfawaer, GuiWei Hua, and Noraziah Ahmed, "A Novel Multicast Routing Protocol for Mobile Ad Hoc Networks", *American Journal of Applied Sciences 4 (5): 333-338, 2007, ISSN 1546-9239*

[12] A. Sabari, K.Duraiswamy, "Ant Based Adaptive Multicast Routing Protocol (AAMRP) for Mobile Ad Hoc Networks ", *International Journal of Computer Science and Information Security (IJCSIS), Vol.6, No. 2, 2009.*

[13] Shailender Gupta, Chirag Kumar, C. K. Nagpal, and Bharat Bhushan, "Performance Evaluation of MANET in Realistic Environment", *I.J.Modern Education and Computer Science, 2012, Vol.7, pp.57-64.*

[14] C. Papageorgiou, K. Birkos, T. Dagiuklas, S. Kotsopoulos, "Simulating Mission Critical Mobile Ad Hoc Networks", *12-th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM) - Tenerife, Canary Islands, Spain 2009.*

[15] Jardosh, A.P.: TERGEN, *http://moment.cs.ucsb.edu/mobility/ (2005-06-12).*

[16] Félix Gómez Mármol, Gregorio Martínez Pérez," Towards pre-standardization of trust and reputation models for distributed and heterogeneous systems", *Elsevier Computer Standards & Interfaces Volume 32, Issue 4, June 2010, Pages 185–196.*

[17] S. Buchegger, J.Y. Le Boudec, "A Robust Reputation System for P2P and Mobile Adhoc Networks", *Proceedings of the Second Workshop on the Economics of Peer-to-Peer Systems. Cambridge MA, USA, Jun 2004.*

[18] F. Almenárez, A. Marín, C. Campo, C. García," PTM: a Pervasive Trust Management Model for Dynamic Open Environments, Privacy and Trust". *First Workshop on Pervasive Security andTrust, Boston,USA,Aug. 2004.*

[19] Shailender Gupta and Chander Kumar, "An Intelligent Efficient Secure Routing Protocol for MANET ", *International Journal of Future Generation Communication and Networking Vol. 6, No. 1, February, 2013.*

[20] Dhurandher, S.K. , Misra, S. , Obaidat, M.S. , Gupta, N. "QDV: A Quality-of-Security-Based Distance Vector Routing Protocol for Wireless Sensor Networks Using Ant olony Optimization" *IEEE International Conference on Wireless & Mobile Computing, Networking & Communication. WiMob 2008, Page(s): 598 – 602*

[21] Chandreyee Chowdhury, Sarmistha Neogy, "Reliability Estimation of Mobile Agent System in MANET with Dynamic Topological and Environmental Conditions", *International Journal on Advances in Networks and Services, vol 4 no 1 & 2, year 2011.*