Reliability Estimation of Mobile Agent System in MANET with Dynamic Topological and Environmental Conditions

Chandreyee Chowdhury
Dept. of Computer Sc. and Engg.
Jadavpur University
Kolkata, India
email: chandreyee.chowdhury@gmail.com

Sarmistha Neogy
Dept. of Computer Sc. and Engg.
Jadavpur University
Kolkata, India
email: sarmisthaneogy@gmail.com

***Abstract*—A mobile agent is an agent with the ability to migrate from one host to another where it can resume its execution. Mobile agents can be used in wireless and mobile network applications in order to save bandwidth and time. In this paper we consider reliability issues that need to be addressed before mobile agents can be used in a broad range of applications in Mobile Adhoc Network. We show how a Mobile Agent based System can be made more reliable despite the uncertainties introduced by underlying network environment. Adhoc network brings in new aspects to dependability because the characteristics of such network affect reliability of the services offered by the agent system. Here we propose an algorithm for estimating the task route reliability of a system of agents that is based on the conditions of the underlying network. The system consists of independent agent groups, each group corresponds to a particular application for which these are deployed. The complexity of mobile agent based system combined with the underlying dynamic topology of adhoc network drives us to estimate it using Monte Carlo simulation. Smooth Random Mobility Model is used to estimate node location at a particular time. Environmental factors like multipath propagation that affect the received signal power are also considered. The results achieved demonstrate the robustness of the proposed algorithm. This paper demonstrates a reliability estimation model for mobile agent based system in mobile adhoc network and shows that reliability is heavily dependent on the conditions of the network and on agent heterogeneity.**

Keywords- Mobile Ad hoc network; Monte-Carlo; Reliability; Mobility Model; Fault-tolerance;

I. INTRODUCTION

A mobile agent is a combination of software program and data, which migrates from a site to another site to perform tasks assigned by a user according to a static or dynamic route [1]. It can be viewed as a distributed abstraction layer that provides the concepts and mechanisms for mobility and communication [2]. An agent consists of three components: the program, which implements it, the execution state of the program and the data. A mobile agent may migrate in two ways namely weak migration and strong migration [3]. Weak migration occurs when only the code of the agent migrates to its destination, a strong migration occurs when the mobile agent carries out its migrations between different hosts while conserving its data, state and code. The platform is the environment of execution. The platform makes it possible to create mobile agents; it offers the necessary elements required by them to perform their tasks such as execution, migration towards other platforms and so on.

Typical benefits of using mobile agents include

- Bandwidth conservation: sending a complex query to the database server for processing.
- Reduced latency: a lightweight server can move closer to its clients
- Load balancing: loads may move from one machine to the other within a network etc.

The route of the mobile agent can be decided by its owner or it can decide its next hop destination on the fly.

Here, we assume the underlying network to be a Mobile Ad Hoc Network (MANET) that typically undergoes constant topology changes, which disrupt the flow of information over the existing paths. Mobile agents are nowadays used in MANETs for various purposes like service discovery [4], network discovery, automatic network reconfiguration etc.

Dependability of any computing system may be defined as the trustworthiness of the system, which allows reliance to be justifiably placed on the service it delivers [5]. It is an integrative concept that encompasses attributes like availability (readiness of usage) and reliability (continuity of correct service) [5]. In MANET, like in any other mobile distributed system, mobile nodes access information through wireless data communication at any time and everywhere (motion and location independence) [6]. Therefore, this environment itself introduces new features and aspects to dependability, affecting both availability and reliability of the services of distributed systems.

Hence the reliability of underlying network becomes a factor that may affect the performance, availability, and strategy of mobile agent systems [7] [8].

In this paper, we define a Mobile Agent-based System (MAS) to be a system consisting of a number of different groups of agents where each group accomplishes an independent task.

The connectivity between the nodes is calculated according to the two-ray model [9] for signal propagation reflecting multipath propagation effect of radio signals. The node movements are assumed to be smooth as is the case in most real life scenario. Smooth Random Mobility Model (SRMM) [10] is used for this purpose. We propose a randomized agent planning strategy where an agent selects a destination almost randomly giving preference to a list of nodes over the others and the routes are also updated dynamically, in order to incorporate node mobility, as agents roam in the network. We estimate the reliability of such a mobile agent based system using Monte Carlo simulation technique. This technique is used to avoid the typical computational complexity that may arise.

Some contemporary work in this area is discussed in Section II. Our work in reliability estimation is presented in details in the subsequent section (III). The simulation results of our reliability model are summarized in Section IV. Finally, Section V concludes with an indication of our future endeavor in this area.

II. Related Works

Reliability analysis of MAS in adhoc network is a complicated problem for which little attention has been paid. Most of the work done in this area is related to distributed systems and distributed applications. But as pointed out in [8], features like scalability and reliability becomes critical in challenging environment with wireless networks. However, the scalability/reliability issue of MAS has been highlighted in [11], although the work does not focus on MANET. We did not see any work that considers transient environmental effects (apart from node mobility) into the reliability calculation for MANET.

A. Reliability of Distributed Systems

Two reliability measures are introduced in [12], distributed program reliability and distributed system reliability. Here graph traversal is used in designing an efficient method to evaluate the proposed measures.

In [13], a unified algorithm is proposed to efficiently generate disjoint file spanning trees by cutting different links, and the distributed program reliability and distributed system reliability are computed based on a simple and consistent union operation on the probability space of the file spanning trees.

In [14], two algorithms are proposed for estimating the reliability of a distributed computing system with imperfect nodes. One is called symbolic method (SM), is based on a symbolic approach that consists of two passes of computation, and the other algorithm, called factoring method (FM), and employs a general factoring technique on both nodes and edges.

B. Mobile Ad Hoc Network

In [15], Toh et al. describes a MANET as a collection of two or more devices equipped with wireless communications and networking capability. This definition is expanded further by explaining the method by which their networking capability is realized. Like point to point radios, ad-hoc devices can communicate directly with other devices within their range. They may also communicate with those outside their range by using intermediate nodes to relay or forward the message to the destination node. This second capability, multi-hop communications without the need for network infrastructure is what makes MANET unique.

Research on ad-hoc networks generally focuses on the modification and creation of protocols in the network and transport layer, such as Transmission Control Protocol/Internet Protocol (TCP/IP) to accommodate the mobility of the nodes and make network performance more robust. In [16], Ye et al. proposed a deployment strategy to increase probability of a 'reliable path'. The increase in path reliability was accomplished through strategic node placement, limiting the application to instances where node mobility be directed. In [17], a protocol is proposed to accommodate the probabilistic reliability of a MANET but it does not explicitly measure network reliability.

C. Reliability of MANET

Due to the analytical complexity and computational cost of developing a closed-form solution, simulation methods, specifically Monte Carlo (MC) simulation are often used to analyze network reliability. In [18], an approach based on MC method is used to solve network reliability problems. In this case graph evolution models are used to increase the accuracy of the resultant approximation. In [19], a MC method is designed to estimate network reliability in the presence of uncertainty about the reliability of both links and nodes.

But little has been addressed on the reliability estimation of MANETs. In [20], analytical and MC-based methods are presented to determine the two-terminal reliability for the adhoc scenario. Here the existence of links was considered in a probabilistic manner to account for the unique features of the MANET. However, there remains a gap in understanding the exact link between a probability and a specific mobility profile for a node. In [21], MC-based methods are presented to determine the two-terminal reliability for the adhoc scenario. This work is an extension of that in [21], by including directly, mobility models in order to allow mobility parameters, such as maximum velocity, to be varied and therefore analyzed directly. The methods in this paper will now allow for the determination of reliability impacts under specific mobility considerations. As an example, one may consider the different reliability estimate when the same networking radios are used to create a network on two different types of vehicles. Here node mobility is simulated using Random Waypoint mobility model [22]. But this Random Waypoint model of mobility being a very simple one often results in unrealistic conclusions.

D. Reliability of Mobile Agents

Little attention has been given to the reliability analysis of MAS. In [23], two algorithms have been proposed for estimating the task route reliability of MAS depending on the conditions of the underlying computer network. In [24], which is an extension of the previous work, a third algorithm based on random walk generation is proposed. It is used for developing a random static planning strategy for mobile agents. However, in both the works the agents are assumed to be independent and the planning strategy seemed to be static. So this work does not address the scenario where agents can change their routes dynamically. Moreover, it does not address the issue of node mobility in between agent migrations.

In [1], a preliminary work has been done on estimating reliability of independent mobile agents roaming around the nodes of a MANET. The protocol considers independent agents only. Node and link failure due to mobility or other factors is predicted according to NHPP. Explicit node movement according to some mobility model is not considered. An agent may migrate to any node with equal probability. This may not be not realistic as some nodes may provide richer information for a particular agent deployed by some application. In [25], the MAS is assumed to be consisting of a number of agent groups demanding for a minimum link capacity. Thus, each agent group requires different channel capacity. Hence, different groups perceive different views of the network. In this scenario the reliability calculation shows that even with large number of heterogeneous agent groups with differing demands of link capacity, the MAS gradually reached a steady state.

III. Our Work

Though mobile agents are recently used in many applications of MANET, dependability analysis of such applications is not much explored. However, attributes like scalability, reliability and availability are affected by the dynamic network topology of MANET. However the scalability/reliability issue of MAS has been highlighted in [11], although the work does not focus on MANET. However, we have done some work on estimating reliability of wireless networks (in [26]), where nodes move according to some mobility model like Smooth Random Mobility Model [10]. But mobile agents are not considered in [26].

Moreover, we have done some preliminary work [1] [25] on agent reliability but it does not consider several issues that are considered in the present work.

A. Terminologies used in this paper

(V,E) the graph (G) representation of our network;
N no. of mobile nodes;
S our mobile agent based system;
M no. of mobile agents that constitutes S and are deployed in the network; thus, S= $\{m_1, ..m_i… m_M\}$
R_s reliability of S;
n no. of nodes successfully visited by an agent;
$\lambda_i(t)$ task route reliability of i^{th} agent in a step of simulation;
$\lambda(t)$ average reliability of all the agents;
L(t) an array of length NxN
$r_i(t)$ the probability that m_i is working correctly at time t that is the individual software reliability of m_i;
Gt,Gr transmitter and receiver gain respectively;
ht,hr height of the transmitting and receiving antenna;
d_{ij} the distance between nodes i and j
Q no. of simulation steps;

B. Problem Definition

In this paper, we assume that our mobile agent-based system (S) consists of M independent agents deployed by k owners that may move in the underlying MANET. The reliability of (S) is defined as the probability that (S) is operational during a period of time [2]. Consequently S is said to be fully operational if all its currently existing mobile agents are functional or operational [3], whereas it is fully down if all its currently existing mobile agents are fully non-operational. Moreover, (S) is said to be partially operational if some of its currently existing mobile agents are operational. Later, in Section III.C we define reliability of an individual agent in this context.

1) Modeling MANET: We model the underlying network as an undirected graph G= (V,E) where V is the set of mobile nodes and E is the set of edges among them. Let the network consist of N nodes, thus, |V|=N that may or may not be connected via bidirectional links (e). The following assumptions are made ([27] [28]):

1) The network graph has no parallel (or redundant) links or nodes.
2) The network graph has bi-directional links.
3) There are no self-loops or edges of the type (vj, vj).
4) The states of vertices and links are mutually statistically independent and can only take one of the two states: working or failed.

Initial locations of the nodes (v_is) are assumed to be provided. The mobility of nodes in MANET can be simulated using SRMM [10]. This model is like Random Waypoint Mobility Model [10] but more realistic as it prevents the nodes from taking sharp turns or making sudden stops.

To incorporate SRMM [10] a Poisson event determines the time instant of change in speed. A new speed is chosen from the interval $[0,V_{max}]$ where 0 and V_{max} are given higher preference and rest of the values are uniformly distributed. Once a target speed is chosen the current speed is changed according to the acceleration a(t), which is once again uniformly distributed in $[0, a_{max}]$. The values of V_{max} and a_{max} may be different for different users. For example, for vehicular traffic, these will have higher values than pedestrians. Thus, as in [10],

$$v_i(t):= v_i(t-\Delta t)+a_i(t)*\Delta t \tag{1}$$

A new target direction is chosen only when $v_i(t)=0$. We simulate here the *stop turn and go* [10] behavior. The target direction is uniformly distributed between $[-\pi/2, \pi/2]$ with $\pi/2$ and $-\pi/2$ having higher priorities [10]. At every time instant direction $(\Delta\varphi_i(t))$ changes incrementally $(\Delta\varphi_i(t))$ unless it attains the target direction. Thus, as in [10],

$$\varphi_i(t)= \varphi_i(t-\Delta t) + \Delta\varphi_i(t) \tag{2}$$

Now, using this speed at previous time instant, acceleration, and direction, we can estimate the position (x_i,y_i) of the node at $(t+\Delta t)$ as

$$x_i(t+\Delta t)=x_i(t)+\Delta t*v_i(t)*\cos\varphi_i(t)+0.5*a_i(t)*\cos\varphi_i(t)*\Delta t^2 \tag{3}$$

$$y_i(t+\Delta t)=y_i(t)+\Delta t*v_i(t)*\sin\varphi_i(t)+0.5*a_i(t)*\sin\varphi_i(t)*\Delta t^2 \quad (4)$$

The movement of the nodes is assumed to be bounded within a specified simulation area as in [10].The distance between a pair of nodes (d_{ij}) can be calculated as follows

$$d_{ij}(t) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (5)$$

The probability of link existence (P_{link}) not only depends on the distance between the nodes but is also very much dependent on the environmental factors. So, even when two nodes remain within the transmission range of each other, but due to factors like signal fading, shadowing, diffraction etc., the quality of transmission can degrade appreciably [29]. The average received power (p_r) is a function of the distance between the transmitter and the receiver. Here we take the two-ray model for radio propagation in order to show how the transmitted signal with power (p_t) suffers from multipath propagation while reaching the receiving end. Thus, $p_r(d)$ can be stated as mentioned below [9]:

$$p_r(d) = p_t G_t G_r \frac{h_t^2 h_r^2}{d^4} \quad (6)$$

In free space, the received power varies inversely to the square of the distance but here we have assumed the exponent to be 4 to indicate the presence of a medium.

2) Modeling Mobile Agent Based System: In this scenario we can think of a mobile agent as a token visiting one node to another in the network (if the nodes are connected) based on some strategy as needed by the underlying applications to accomplish its task.

An agent starts its journey from a given owner and moves from one node to another at its will. The owner provides a priority list to the agent, which contains a list of node ids that are most beneficial migration sites (for the application that deployed that particular agent). So, an agent will always try to visit those nodes from the priority list as its first preference. But this movement is successful if the two nodes are connected and there is no simultaneous transmission in the neighborhood of the intended destination. We assume that cases of collisions (if any) are taken care of by the underlying MAC protocol. So, we associate a probability with the movement to indicate transient characteristics of the environment, since, for example, the routing table may not be updated properly or the link quality may have degraded so much (due to increased noise level) that the agents are unable to migrate. Thus, if an agent residing at node A decides to move to node B (connected to A) then the agent successfully moves to B with probability p_{tr}. Here p_{tr} denotes the problem of unpredictable background noise level mentioned above. For example, noise level may increase due to heavy rainfall.

Let us suppose that at an instance t, the MANET consists of five nodes namely MN_A, MN_B, MN_C, MN_D and MN_E and their connectivity is as shown in Figure 1.The dotted line represents an erroneous link. We assume that all the nodes have appropriate host platform for the agents and the agents may update their migration policy on the fly. An agent x (say) residing at node A does the following:

1) It chooses its next destination almost randomly giving more preference to the nodes in the priority list. If that destination is not visited before and if there is a path then x moves to its new location with probability p_{tr}.

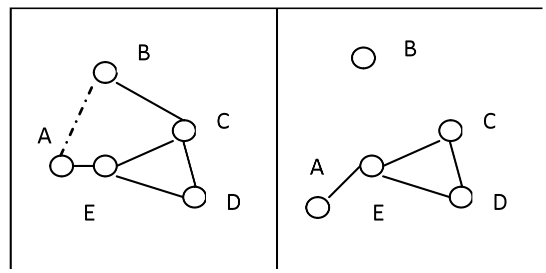


Figure 1. An instance of a network graph at instant t (left) and t+Δt (right) respectively

2) But when x attempts to move to MN_B at (t+Δt) time instant, the network graph changes (Figure 1) and MN_B becomes an isolated node, which is unreachable. It may also happen that the capacity of the link (from MN_A to MN_B) is lower than that needed by x. So for the underlying routing algorithm, a link exists between MN_A and MN_B but for agent x, the capacity of the link is not sufficient. So MN_B is unreachable for x.
3) So x will not be able to move to MN_B.
4) In the next time instant x may retry or try to choose its next destination randomly again.

This helps in the improvement in system performance. This is because of the fact that the agents themselves try to overcome the transient faults.

3) Modeling Agent Reliability:

In this scenario we study the reliability of MAS with respect to the network status and its conditions (for example, connectivity of the links, path loss probability etc.). We start with a dynamic planning strategy where each agent is expected to visit N (<=number of nodes in the network) nodes in the network to accomplish its task. Each group of agents starts its journey from a given node, which acts as its owner. We assume that a node can only own a single group of agents. In other words, a node can only host one application that will deploy a number of agents. Due to the constraints of mobile nodes (MN) such assumption is not absurd at all.

We have taken the failure probability (P) of the mobile nodes (P_{Node}) to be a variable of Weibull distribution [21].

Now reliability of MAS (R_s) can be defined as

$$R_s = \{R_{MAS} | R_{MANET}\} \quad (7)$$

Here reliability of MANET (R_{MANET}) can be treated as an accumulative factor of ($1-P_{Node}$) and P_{Link}. P_{Link} can be treated as a combination of P (p_r is at an acceptable level) and the mobility model. Here p_r denotes the received power at node j after traversing distance d_{ij} from sender node i.

Here we calculate individual agent reliability on the underlying MANET as follows:

If an agent can successfully visit M nodes out of N(desired) then it has accomplished M/N portion of its task. Thus, reliability in this case will be M/N.

But if the application requires all N nodes to be visited in order to fully accomplish the task and in all other cases the task will not be considered to be done, reliability calculation will be modified as:

If an agent can successfully visit all N nodes desired then it has accomplished its task. Thus, reliability in this case will be 1. In all other cases it will be 0.

Above definitions of agent reliability works only if there is no software failure of the agent (assumed to follow Weibull distribution [21]).

Now, the probability that the MAS is operational i.e., reliability of MAS (R_{MAS}) can be calculated as the mean of reliability of all its components, that is, the agents in this system.

$$R_{MAS} = \frac{\sum \{Agent\ \mathrm{Re}\, liabilities\}}{No.ofAgents} \quad (8)$$

Finally to calculate R_s in equation 7 an algorithm is proposed in this paper in the next section.

C. Steps of Reliability calculation of mobile agent with dynamic route

1) SRMM is used to simulate the effect of node mobility.
2) The probability of the existence of a link is calculated according to equation 6 to cover multipath propagation effect of radio signals.
3) Breadth First Search (BFS) is used iteratively to identify the connected components (clusters) of the network and are given unique identifiers (cluster id).
4) A mobile agent prefers to select a destination, which is not visited before, from the priority list. If it finds a route (that is if the source and destination share the same cluster id) then it moves with a certain probability and the process continues otherwise the process halts.
5) Individual node failure is also considered and Weibull distribution [21] is used to simulate the same. Weibull distribution takes two parameters, scale and shape. We have given the values in such a way that as time passes on the probability of failure also increases.
6) Finally, Monte Carlo method of simulation is used to find the overall reliability.

1) Input parameters: M (number of independent mobile agents in the system), The initial state of the network (node position, location, speed of the nodes)

2) Detailed Steps:

1. Initialize n (that is the number of mobile nodes successfully visited by an agent) to 0 and a source for the mobile agent.
2. List of vertices along with their initial positions is given.
3. The priority list for each agent group is also formed and kept with the owners.
4. i. To simulate the effect of node mobility create E', a subset of VXV with the same using SRMM as follows.
    a. The $v_i(t)$ and $\varphi_i(t)$ are calculated using equation (1) and (2) respectively.
    b. The position of each MN is updated for the next time increment by equation (3) and (4).
    c. Distance between each pair of nodes is calculated using equation (5) and E' is populated according to equation (6).

   ii. Some nodes may also fail because of software/hardware failure or become disconnected from the network according to NHPP distribution. Node failure can be simulated by deleting the edges e from E' further that are incident on the failed node $v \in V$.
5. According to Weibull distribution we find individual software reliability r_i for an agent i.
6. BFS is used unless all connected subgraphs are assigned a proper cluster id. Thus, an isolated node is also a cluster.
7. The agents perform their job on this modified graph.
    a. An agent will prefer to choose a node to be its next destination if it is in its priority list and is not visited already. All other nodes (not there in the priority list) are equally likely destinations.
    b. If that destination falls in the same cluster as it is now residing, the agent moves to the new destination with probability p that represents the instantaneous background noise level in the network. If it succeeds, n is incremented by 1.
    c. Despite several attempts that an agent may make, if an agent fails to move to its next destination (say $node_i$), then,
        i. the agent tries to move to other destinations as needed by the application.
8. Repeat steps 3 to 6 until all nodes are visited or the new destination falls in a different cluster.

9. Calculate $$\lambda_i(t) = \frac{n}{N} \quad (9)$$

Here the value of n depends heavily on the conditions of the underlying network.

10. Reset the value of n.
11. Repeat steps 5-9 for all agents (k) in the system.
12. Calculate $$\lambda(t) = \frac{1}{k}\sum_{i=1}^{k} \lambda_i(t) r_i \quad (10)$$
13. Repeat steps 3 to 11 Q (simulation steps) times.
14. Calculate node reliability $$\frac{1}{Q}\sum_{q=1}^{Q} \lambda(q,t) \quad (11)$$

It is to be mentioned that step 4 is repeated for every move of the mobile agent. Since in a typical adhoc scenario we cannot assume the nodes to be static during the entire tour of the mobile agents so after every single move the entire network configuration (hence the effect of node mobility) is recalculated. Moreover in this case E' does not have to be a subset of E because with time some nodes may also move closer to the other nodes and thus, creating a link between them.

If an agent fails to move because of background noise level, then it may retry depending on the amount of delay that the respective application can tolerate.

Here we have assumed that in order to accomplish a task the agents need to visit all the nodes in the network. So we have N as the denominator in equation 9. But we can change this parameter and our algorithm will still work if lesser number of nodes is needed to be visited. We have also assumed that the agent can always retract to its owner.

It may be seen in practice that in a network some nodes have rich information and the agents tend to move to those nodes as their next destination over the other. That is why, we prioritize the nodes by providing a priority list rather than randomly selecting the next destination in step 7 of the algorithm. Here we feed the priority list from owners but the agents may also learn about such rich nodes from their experience and may share this information also with the others using some multiagent communication scheme like the blackboard model [30]. A mobile agent may leave a message for another agent at one of the N hosts. Whenever the dependent agent comes to that host it will receive that message and act accordingly. So, the node priorities can also be modified on the fly. This is a possible application of learning [31] in this system.

TABLE I. PRIORITY LIST OF THE AGENTS

Agent Id	Priority List
Agent 1	*MN_2,MN_4*
Agent 2	*MN_1, MN_3*
Agent 3	*MN_4*
Agent 4	*MN_1, MN_2*

D. An Example

We have taken an instance where there are ten nodes in the network. Four mobile agents are deployed by four different owners and they start their journey from their owners. Agents 1, 2, 3 and 4 start their journey from nodes MN_1, MN_2, MN_3 and MN_4 respectively and roam around the network to accomplish its task. Thus, an application (for example, service discovery) running on MN_1 deploys agent 1. Our job is to find the number of nodes that are successfully visited by these agents, which indicates the progress of its task (how many services the agents discover for a MANET) and consequently the reliability of the agent group will be calculated. Average reliability of all groups taken over a certain time period for a number of simulations represents the reliability of the MAS despite the uncertainties of MANET. So, for reliability calculation we are giving equal priority to all nodes. However, our migration policy gives some nodes higher weight over the others (step 7a in the algorithm) indicating the fact that all destinations are not equally likely. The agents are fed with a given priority list by their respective owners as shown in TableI. For example, visiting nodes MN_2 and MN_4 will be most beneficial for agent 1 and so on.

The nodes are taken close enough (Figure 2) so that they form an almost connected network. As shown in Figure 3a, MN_9 is isolated from the MANET initially. But eventually it finds MN_{10} within its range and hence can connect itself to the network (Figures 3b, c and d). This strategy of node distribution sounds realistic as the nodes in a MANET may not remain connected to each other always due to individual node movement and environmental characteristics.

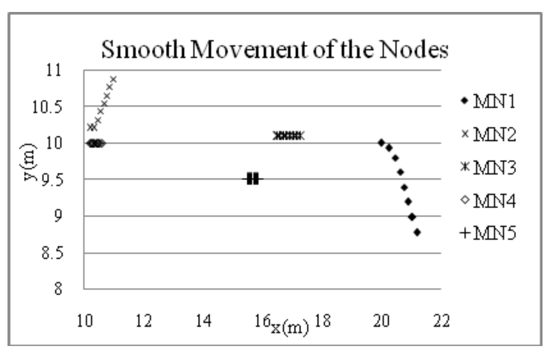
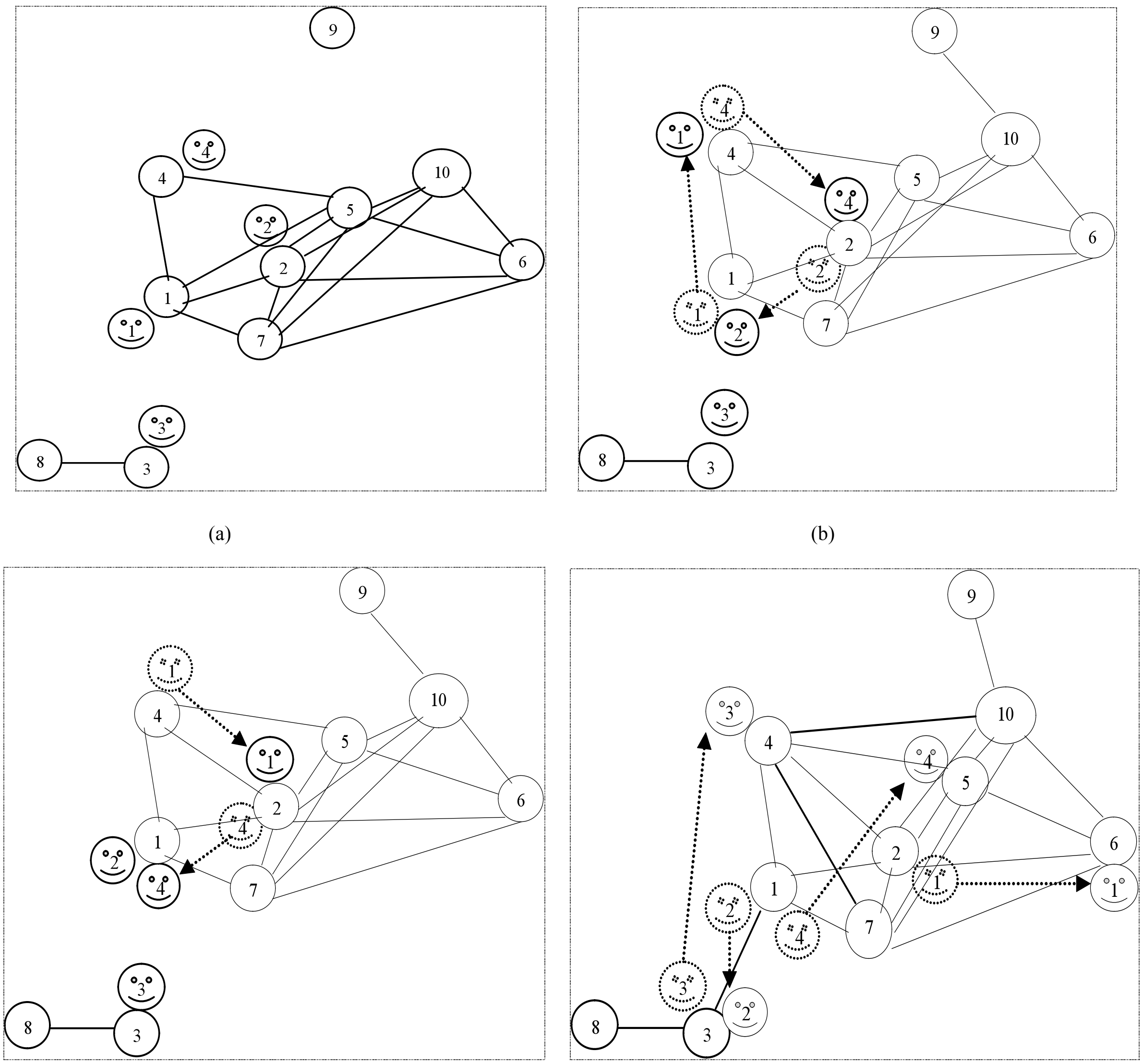


Figure 2. Movement of the nodes according to SRMM

Every 3 seconds the positions of the nodes are updated according to SRMM. The simulation is carried out for 30 seconds and the positions of the different nodes are given in Figure 2. The smooth movement of the nodes is obvious from the figure itself. Connectivity of the nodes is calculated according to the Two-ray model. For convenience we have only shown four nodes to be deploying agents. The network topology at 4 successive time instants is shown in Figure 3(a, b, c and d). Agents are also shown in Figure 3 by callouts along with a numeral to indicate agent ids. The dotted ones (callouts) represent the starting position and the bold ones (callouts) represent end point of their journey at that time instant.

Figure 3(a) indicates a disconnected network graph for the MANET with an isolated node (MN_9) and two components (clusters). Nodes, MN_3 and MN_8 form one



(a) (b) (c) (d)

Figure 3.

a. Network graph at time instant t=t0 and the position of the agents

b. Network graph at time instant t=t0+Δt and the position of the agents

c. Network graph at time instant t=t0+2Δt and the position of the agents

d. Network graph at time instant t=t0+3Δt and the position of the agents cluster and all other nodes (except MN_3, MN_8 and MN_9) fall into a different cluster. Here any agent can move to any destination it wants to within its cluster. The agents start their journey in such a scenario.

While the nodes move and form a network configuration as shown in Figure 3(b), the agents also start migrating in the network. The network connectivity is slightly changed here as MN_9 now comes within the transmission range of MN_{10} and hence becomes connected to one of the clusters. So our MANET now contains two clusters, (one containing MN_3 and MN_8 and the other containing the rest). Since agent1 gives highest priority to MN_2 and MN_4 over the others so, agent1 first visits MN_4. For similar reasons, agent2 visits MN_1 (from MN_2) and agent4 visits MN_2 (from MN_4) respectively. But agent3 cannot migrate successfully as node MN_4, the highly beneficial migration site for agent3, lies in a different cluster.

Network connectivity changes a little in the next 3 seconds as indicated in Figure 3c. So, agents 1 and 4 make successful migrations to their highly preferred destinations such as MN_2 (from MN_4) and MN_1 (from MN_2) respectively. However, MN_3, a highly beneficial migration site for agent2 falls in a different cluster than MN_1 (where agent2 currently resides). Consequently, agent2 cannot make any migration but stays at MN_1. Moreover due to transient characteristics, the link between nodes MN_3 and MN_8 becomes erroneous. As a result agent3 makes an unsuccessful attempt (step 7b in the algorithm) to migrate to MN_8 (from MN_3) but stays at MN_3. As the agents are sent with a given probability, even if nodes fall in the same cluster, an agent may not be able to make a successful migration. This scenario indicates the notable effect of transient errors on the performance of MAS.

Finally in the next 3 seconds the collection of nodes form a connected graph as MN_3 comes within the transmission range of MN_1. Now the agents can migrate to any other node with a certain probability (step 7 of our algorithm). Thus, agents 1 and 4 migrate to MN_6 (from MN_2) and MN_5 (from MN_1) respectively. Agents 2 and 3 also finally find their most beneficial migration sites (MN_3 for agent 2 and MN_4 for agent 3) reachable and attempt to make successful migrations.

In this way, the simulation is continued and the nodes in the MANET continued to form different network configurations affecting agent migrations. The value for received power is taken to be 16dBm. In the calculation the antenna gains are taken to be 2.2dBi, the height is taken to be 2m and the transmitting power is taken to be 20dBm [32].

At the end of the 10th second, agents 1, 2 and 4 finish migration to 9 nodes each (including their owners) out of all 10 nodes in the MANET accomplishing (9/10 that is) 90% of their tasks each. However agent3 was only able to cover 7 out of 10 nodes (70%) because its owner MN_3 was disconnected from most nodes of MANET for a while, thus, accomplishing only 70% of its task. This scenario shows the effect of MANET configuration on the performance of MAS. Thus, the overall reliability of MAS comes out to be (3*0.9+0.7)/4=0.85 that is 85%. If another simulation run is carried out for the same amount of the time, then the overall reliability comes out to be 0.825. If we use Monte Carlo simulation for a number of times (Q=100 onwards) the overall reliability tends to converge to 0.53 (as shown in Table II). Thus, with a MANET of 10 nodes moving according to SRMM, the MAS where the agents almost randomly choose their neighbor and migrate, will be 53% reliable.

TABLE II. VARIATION OF RELIABILITY WITH NO. OF MONTE CARLO SIMULATION STEPS

Q	Reliability
10	*0.539*
100	*0.5315*
500	*0.5319*
1000	*0.5314*
2000	*0.5312*
10000	*0.5319*

IV. EXPERIMENTAL RESULTS

The simulation is carried out in Java and it can run in any platform. The initial positions of the MNs are given along with their initial speeds and the maximum acceleration that can be attained by them. All agents of the same group start from the same node, that node are designated to be the owner. The maximum allowable speed and acceleration of the MNs are read from a file. These values are needed by SRMM. The simulation time is taken to be 1 hour. For the rest of the experiments, the number of nodes is taken to be 40 unless stated otherwise. The other parameters like received power, antenna gains are kept the same as mentioned in the example (Section IIID). Unless otherwise stated the number of agents is taken to be 30 and the number of groups is taken to be 4. In MANET due to environmental factors like diffraction, fading along with asynchronies in movement pattern, some nodes become isolated from the network. Some of the nodes may rejoin and some remain disconnected from the network. So, to start with we have taken such a scenario (of MANET) in terms of initial node positions and respective speeds.

With four (4) groups and a total of 30 mobile agents, if we increase the MANET size, the reliability is found to drop eventually as shown in Figure 4. This result is in concurrence with the one we get in [1]. Here at every step we add approximately 10 nodes but almost none of them remain within the transmission range of any of the disconnected components of the existing MANET. This is not also possible in a MANET with an appreciable diameter. So the number of successful agent migration reduces as more nodes become unreachable for an agent. Consequently at each step there is no drastic change in network connectivity as can be observed in [1], just the size of some disconnected components increase. This results in the gradual fall in reliability with increasing N.

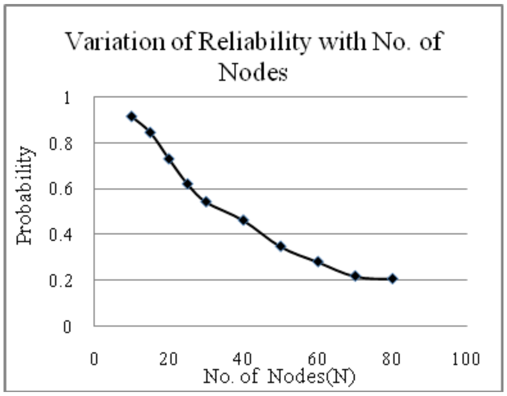


Figure 4. Variation of reliability with increasing network size

Now we look into the matter in more details for MANET with fast moving nodes. The maximum acceleration of the nodes is varied to yield different standard deviations for a given mean. When the average of all the maximum acceleration that a node can attain is 0.75, we plotted the reliability value for standard deviation = 0.1, 0.2, 0.3, 0.4 and 1. Similar things have been done for average value of 1.5 and 3 as shown in Figure 5. In most cases for a given standard deviation, higher mean implies lower reliability. So, this indicates the fact that when all nodes have the same variance in speed, if the overall MANET nodes are slower then obviously, the nodes will remain crowded implying higher reliability. On the contrary, for a given mean, higher the standard deviation, lesser will be the reliability. This indicates that when all the nodes move with comparable speed (lower standard deviation), for example, group movement in disaster relief or military operations, overall reliability improves. But when some nodes lag behind the others, reliability of MAS would get hampered as the MANET breaks into a number of clusters.

The above mentioned conclusion is valid irrespective of the MANET nodes being slower or faster. Thus, in Figure 6, the points at the peak of the curve yield lower standard deviation. But if the mean goes even higher, that is for faster MANET, the reliability of MAS reduces as shown in Figure 6.

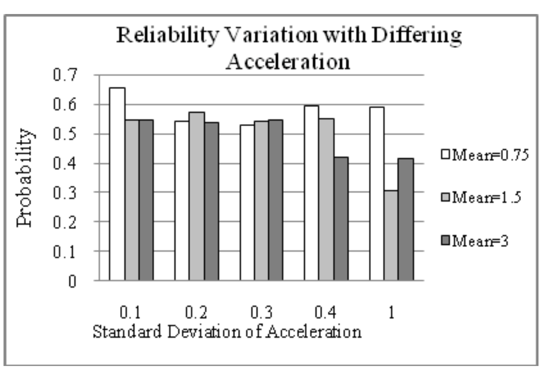


Figure 5. Variation of reliability with greater variation of node accelerations

The effect of background noise is observed in Figure 7. As the environment becomes noisier, such as urban areas, interference is higher. So the receiver would not be able to decode the signal if the received signal power is low. Thus, a weak signal having signal power of 8dBm could not be decoded in crowded areas. But for environment with lower interference, such as highways or countryside, the transmission range increases, enabling weaker signals (having power of 8-15dBm) to be detected and decoded properly. Hence network connectivity improves making MAS more reliable.

We have seen that if node movements are allowed only at the beginning before the mobile agents start their task route, then performance of the algorithm does not vary appreciably with number of mobile agents deployed in the system. But if the situation is made more realistic by allowing node movements in between agent migration then reliability of MAS varies with its size as shown in Figure 8. As far as the number of agent groups remains fixed (heterogeneity), the increasing size of MAS (in terms of M) does not seem to affect reliability greatly. But if the heterogeneity among agents increases, even for a fixed size of MAS, reliability improves and slowly reaches a stable

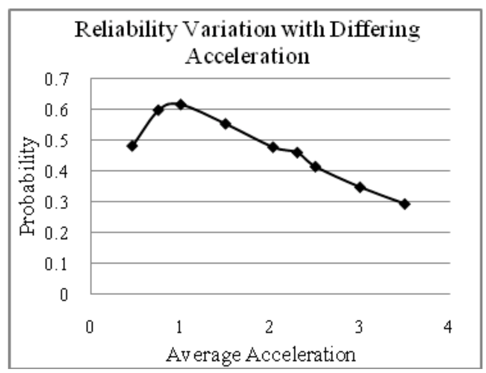


Figure 6. Variation of reliability with faster nodes

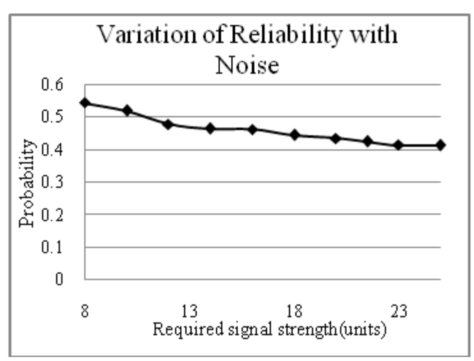


Figure 7. Reliability Variation with Noise

state. This result is significant as it shows that a large number of applications deploying different types of agents (having different migration pattern) does not hamper the reliability of MAS. Rather they cover the different parts of the network in a better manner and can better exploit the denser portion of MANET. So, an increasing number of heterogeneous agents yield better performance than a single group of homogeneous agents of comparable size. This is because the homogeneous agents have similar migration pattern, they start from the same region of MANET and tend to face similar connectivity problems.

Let us now concentrate in the migration pattern of the agents. As we know, every agent is provided with a preferred list of migration sites (priority list of the agents) by their owner. Longer the priority list wider will be the agent's scope to choose its next destination. But still, the probability of successful agent migration remains highly dependent on the position and connectivity of the next destination. Hence as shown in Figure 9, only a little improvement can be observed for longer priority list.

Keeping all parameters fixed if we increase the simulation time, the MANET diameter increases, thus, decreasing the overall reliability of MAS. But after some time the network connectivity somewhat stabilizes, thus, reducing any further the rate of change of reliability with time and the system enters a somewhat stable state (shown in Figure10).

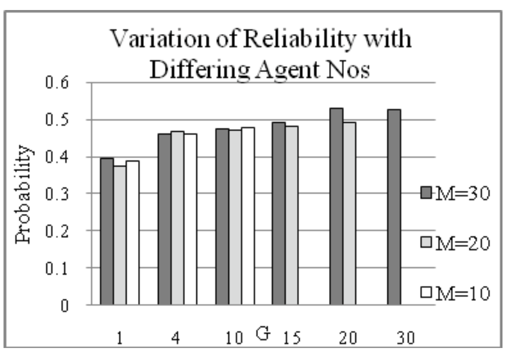


Figure 8. Reliability variation with increasing no. of agents and agent groups

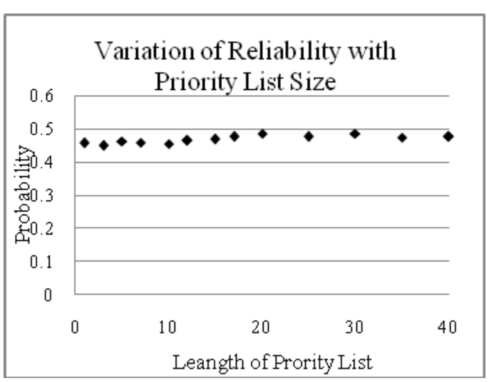


Figure 9. Effect of varying priority list size on MAS reliability

V. CONCLUSION

In this paper, a scalable approach to estimate the reliability of a mobile agent based system for MANET is presented. The reliability calculation depends heavily on the conditions of MANET, like area covered by a node, size of MANET and of course, node mobility.

A starting point for the agents is provided. However the agents are not fed with a given route, rather a list of preferred migration sites are mentioned, which is quite practical. The agents show slightly better reliability if more nodes are designated as preferred migration sites, that is, the agent's scope becomes a little wider.

SRMM is used to simulate the movement of the nodes. The protocol is validated and results are shown in Section IV. It can be observed that for a faster MANET only if all the nodes move with comparable speeds then MAS is found to be appreciably reliable. Higher background noise is also found to hamper the reliability of MAS.

As can be seen, reliability improves heavily if the agent set is sufficiently heterogeneous, despite the dynamics and uncertainties associated with MANET. This work does not consider agents with differing QoS requirements for migration.

We are planning to include (i) mobile agents designed specifically for an application like service discovery and (ii) security characteristics to design a fully dependable system.

REFERENCE

[1] C. Chowdhury and S. Neogy, "Estimating Reliability of Mobile Agent System for Mobile Ad hoc Networks", Proc. 3rd International Conference on Dependability, DEPEND 2010, pp. 45-50, 2010.

[2] J. Cao, X. Feng , J. Lu, and S. K. Das, "Mailbox-Based Scheme for Designing Mobile Agent Communications", Computer, Vol. 35 n.9, pp. 54-60, September 2002.

[3] N. Migas, W.J. Buchanan, and K. McArtney, "Migration of mobile agents in ad-hoc, Wireless Networks", Proc. 11th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems, pp. 530 – 535, 2004.

[4] R.T. Meier, J. Dunkel, Y. Kakuda, and T. Ohta, "Mobile agents for service discovery in ad hoc networks", Proc. 22nd International Conference on Advanced Information Networking and Applications, pp. 114-121, 2008.

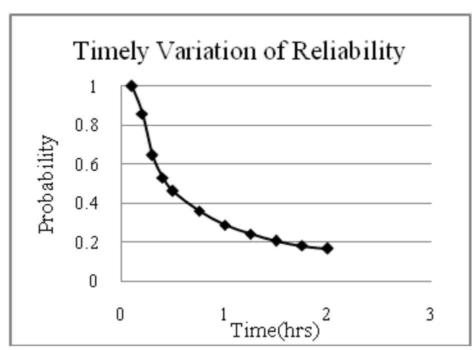


Figure 10. Timely variation of reliability

[5] R. H. Laamanen, T. Alonko, and K. Raatikainen, "Dependability issues in mobile distributed system", Proc. of the Pacific Rim International Symposium on Dependable Computing, pp. 7-14, 1999.

[6] A. Avizienis, J. Laprie, and B. Randell, "Fundamental concepts of dependability", LAAS-CNRS, Technical Report N01145, Apr. 2001.

[7] A.L. Murphy and G. P. Picco, "Reliable communication for highly mobile agents", Autonomous Agents and Multi-Agent Systems, Vol. 5, Issue 1, pp. 81-100, 2002.

[8] O. Urra, S. Ilarri, and E. Mena, "Agents jumping in the air:dream or reality", Proc. 10th International Work-Conference on Artificial Neural Networks, IWANN'09, Special Session on Practical Applications of Agents and Multi-Agent Systems, pp. 627–634, 2009.

[9] M. Rooryck, "Modelling multiple path propagation-Application to a two ray model", in the journal of L'Onde Electrique , Vol. 63, pp. 30-34, Aug.-Sept. 1983.

[10] C. Bettstetter, "Smooth is better than sharp: a random mobility model for simulation of wireless networks", Proc. 4th ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, pp. 19-25, 2001.

[11] S. Ilarri, R. Trillo, and E. Mena, "SPRINGS: A scalable platform for highly mobile agents in distributed computing environments", Proc. 4th International WoWMoM 2006, Workshop on Mobile Distributed Computing (MDC 2006), pp. 633–637, 2006.

[12] C. S. Raghavendra, V.K.P. Kumar, and S. Hariri, "Reliability analysis in distributed systems", in the IEEE Transactions on Computing, Vol. 37, Issue 3, pp.352–358, 1988.

[13] D.J. Chen, and T. H. Huang, "Reliability analysis of distributed systems based on a fast reliability algorithm", in the IEEE Transactions on Parallel Distributed Systems, Vol. 3, Issue 2, pp.139–154, 1992.

[14] M.S. Lin, D.J. Chen, and M.S. Horng, "The reliability analysis of distributed computing systems with imperfect nodes", The Computer Journal, Vol. 42, Issue 2, pp. 129–141, 1999.

[15] C-K Toh, "Ad Hoc Mobile Wireless Networks", Prentice Hall PTR © 2002.

[16] Z. Ye, S. V. Krishnamurthy, and S. K. A Tripathi, "Routing Framework For Providing Robustness to Node Failures in Mobile Ad Hoc Networks",in Ad Hoc Networks, Vol. 2, No. 1, pp. 87-107, 2004.

[17] M. Abolhasan, T. Wysocki, and E. Dutkiewicz, "A Review of Routing Protocols for Mobile Ad Hoc Networks", in Ad Hoc Networks, Vol. 2, No. 1, pp.1-22, January 2004.

[18] T.Elperin, I. Gertsbakh, and M. Lomonosov, "Estimation of network reliability using graph evolution models", in the IEEE Transactions on Reliability, Vol. 40. No. 5. pp 572-581, 1991.

[19] M. Marseguerra, E. Zio, P. Luca, and D. W. Coit, "Optimal Design of Reliable Network Systems in Presence of Uncertainty", in the IEEE Transactions on Reliability, Vol. 54, No. 2, June 2005.

[20] J. L. Cook and J. E Ramirez-Marquez, "Two-terminal reliability analyses for a mobile ad-hoc wireless network", in Reliability Engineering and System Safety, Vol. 92, Issue 6, pp. 821-829, June 2007.

[21] J. L. Cook and J. E. Ramirez-Marquez, "Mobility and reliability modeling for a mobile ad-hoc network", IIE Transactions, Vol. 41, Issue 1, pp. 23 – 31, 2009.

[22] C. Bettstetter and C. Wagner, "The spatial node distribution of the random waypoint mobility model", Proc 1st German Workshop on Mobile Ad hoc Networks (WMAN), pp. 41-58, 2002.

[23] M. Daoud and Q. H. Mahmoud, "Reliability estimation of mobile agent systems using the Monte Carlo approach", Proc. 19th IEEE AINA Workshop on Information Networking and Applications, pp. 185–188, 2005.

[24] M. Daoud and Q. H. Mahmoud, "Monte Carlo simulation-based algorithms for estimating the reliability of mobile agent-based systems" Journal of Network and Computer Applications, pp. 19–31, 2008.

[25] C. Chowdhury and S. Neogy, "Reliability Estimate of Mobile Agent Based System for QoS MANET Applications", Proc. Annual Reliability and Availability Symposium 2011, RAMS 2011, pp.1-6, 2011.

[26] C. Chowdhury and S. Neogy, "Reliability estimation of fault-tolerant wireless and mobile networks", Proc. 3rd International Conference on Communication Theory, Reliability, and Quality of Service, CTRQ 2010, pp. 67-72 2010.

[27] M. L. Shooman, "Reliability of computer systems and networks: fault tolerance, analysis, and design". New York: Wiley; 2002.

[28] C. Srivaree-ratana, A. Konak, and A. E. Smith, "Estimation of all-terminal network reliability using an artificial neural network", Computers and Operations Research, Vol. 29, pp. 849–868, 2002.

[29] H. Taub and D. L. Schilling, "Principles of Communication Systems", McGraw-Hill, 1986.

[30] S. Li and F. Hu, "Communication between the RoboCup agents based on the blackboard model and observer pattern", Proc. 5th international conference on Wireless Communications, Networking and Mobile Computing, pp. 5007-5011, 2009.

[31] R. S. Sutton and A. G. Barto, "Reinforcement Learning:An Introduction", MIT Press, Cambridge, MA,1998.

[32] W. Stallings, "Wireless Communications and Networks", Pearson, Second Edition, 2009.