

Bio-Inspired Approach to Security and Routing in large scale highly mobile Wireless Networks

Aadith Raman¹, Ankur Khetrapal¹, Anvay Lonkar², Sandeep Dalmia¹, Rajni Jindal¹

¹Department of Computer Engineering

²Department of Information Technology
Delhi College of Engineering

ABSTRACT

As one of the most dynamic and gigantic instantiations of the Mobile Ad Hoc Networks, VANET's have attracted a lot of research towards the scalability of security aspects. Most results tend to point towards a digital signature based Public Key Encryption mechanism. Intermittent access to the fixed infrastructure, especially during the initial stages of deployment, makes certificate management a core issue in all such research activities. In this paper we propose swarm intelligence based temporal certificate revocation information dissemination mechanism for robust and efficient detection and certificate revocation of malicious (or anomalous) nodes. We use the Ant colony optimization as a metaheuristic optimization algorithm to find a globally optimal solution for distributing certificate revocation information to the nodes that need it most. Our simulation results show lower latencies for prohibition of malicious nodes from the network.

1.INTRODUCTION

1.1 Bio Inspired Algorithms

In ant societies and, more in general, in insect societies, the activities of the individuals, as well as of the society as a whole, are not regulated by any explicit form of centralized control. On the other hand, adaptive and robust behaviors transcending the behavioral repertoire of the single individual can be easily observed at society level. These complex global behaviors are the result of self-organizing dynamics driven by local interactions and communications among a number of relatively simple individuals. The simultaneous presence of these and other fascinating and unique characteristics have made ant societies an attractive and inspiring model for building new algorithms and new multi-agent systems. In the last decade, ant societies have been taken as a reference for an ever growing body of scientific work, mostly in the fields of robotics, operations research, and telecommunications.

1.2 Ant Colony Optimization

Among the different works inspired by ant colonies, the *Ant Colony Optimization metaheuristic* (ACO) is probably the most successful and popular one. The ACO metaheuristic is a multi-agent framework for combinatorial optimization whose main components are: a set of ant-like agents, the use of memory and of stochastic decisions, and strategies of collective and distributed learning. It finds its roots in the experimental observation of a specific foraging behavior of some ant colonies that, under appropriate conditions, are able to select the shortest path among few possible paths connecting their nest to a food site. The pheromone, a volatile chemical substance laid on the ground by the ants while walking and affecting in turn their moving decisions according to its local intensity, is the mediator of this behavior.

The ACO's synthesis was also motivated by the usually good performance shown by the algorithms (e.g., for several important combinatorial problems like the quadratic assignment, vehicle routing and job shop scheduling, ACO implementations have outperformed state-of-the-art algorithms). We have identified in dynamic problems in networked systems the most appropriate domain of application for the ACO ideas.

1.3 Application of ACO

ACO is characterized as a policy search strategy aimed at learning the distributed parameters (called pheromone variables in accordance with the biological metaphor) of the stochastic decision policy which is used by so-called ant agents to generate solutions. Each ant represents in practice an independent sequential decision process aimed at constructing a possibly feasible solution for the optimization problem at hand by using only information local to the decision step. Ants are repeatedly and concurrently generated in order to sample the solution set according to the current policy. The outcomes of the generated solutions are used to partially evaluate the current policy, spot the most promising search areas, and update the policy parameters in order to possibly focus the

search in those promising areas while keeping a satisfactory level of overall exploration.

This way of looking at ACO has facilitated to disclose the strict relationships between ACO and other well-known frameworks, like dynamic programming and reinforcement learning. In turn, this has favored reasoning on the general properties of ACO in terms of amount of complete state information which is used by the ACO's ants to take optimized decisions and to encode in pheromone variables memory of both the decisions that belonged to the sampled solutions and their quality.

1.4 ACO for Routing

The first half of the paper is devoted to the study of the application of ACO to problems of routing in networked systems. This class of problems has been identified in the project as the most appropriate for the application of the multi-agent, distributed, and adaptive nature of the ACO architecture. The algorithms cover a wide spectrum of possible types of network by providing a best-effort service in ad hoc networks. The algorithms for wired networks have been extensively tested by simulation studies and compared to state-of-the-art algorithms. The observed experimental performance is excellent, especially for the case of mobile networks: our algorithms always perform comparably or much better than the state-of-the-art competitors. In the report, we try to understand the rationale behind the brilliant performance obtained and the good level of popularity reached by our system. More in general, we discuss the reasons of the general efficacy of the ACO approach for network routing problems compared to the characteristics of more classical approaches. Moving further, we also informally define Ant based Routing Protocol (ARP), a multi-agent framework explicitly integrating learning components into the ACO's design in order to define a general and in a sense futuristic architecture for autonomic network control.

1.5 ACO for Certificate Revocation

ACO, being an efficient distributed multi-agent algorithm, has been very effective as an information propagation algorithm. Handling security for mobile and sparse infrastructure networks like Vehicular Ad Hoc Networks

(VANETS) pose a very interesting area of application for such algorithms. Specifically, disseminating information regarding the maliciousness of nodes in a network can be offloaded from a central trusted third party to the nodes themselves by effective utilization of such algorithms. In this part of the project we implement a scenario for such applications of ACO. Proposing a pheromone drop mechanism we enable nodes to make their neighbors aware of malicious nodes in the vicinity – without using any third party. Challenges include aggregation of such pheromones to make revocation decisions. We also propose metrics to evaluate algorithms for such an application. All these mechanism and related decision making processes are termed as a single Ant Based – Distributed Certificate Revocation Algorithm (AB-DCRP). We compare this algorithm to another proposed distributed certificate revocation scheme.

The rest of the paper is organized as follows: Section 2: describes the basics of the swarm intelligence approach; Section 3 describes our algorithm; Section 4 provides our simulation results and we conclude in Section 5.

2. SWARM INTELLIGENCE

The Swarm Intelligence Paradigm comprises of the ant-based colony optimization algorithm that is widely used to solve various dynamic combinatorial optimization problems. The basic principle behind these algorithms is a concept called *stigmergy* - communicating through the environment. As an example, ants drop pheromones to communicate with other ants that might be following their trails. This principle is widely used in routing for networks. Figure 1 demonstrates how ants use this pheromone deposit paradigm in real time to find food.

In our approach vehicular nodes interacting with each other send out ants depositing information (i.e. pheromone) regarding the maliciousness of other nodes. Then, a node establishes the authenticity of another node following a pheromone regulation process and hence aggregating all information available to the network taking an informed decision in an optimized manner.

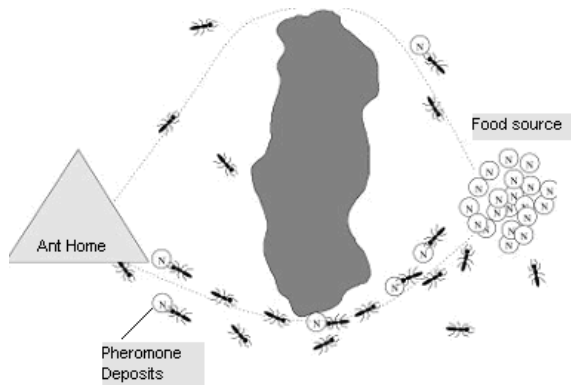


Fig1. Ants using pheromone deposits to find food

3. ANT BASED ALGORITHMS

3.1 ANT ROUTING PROTOCOL (ARP)

The basic idea behind ACO algorithms for routing is the acquisition of routing information through sampling of paths using small control packets, which are called ants. The ants are generated concurrently and independently by the nodes, with the task to test a path to an assigned destination. An ant going from source node s to destination node d collects information about the quality of the path (e.g. end-to-end delay, number of hops, etc.), and uses this on its way back from d to s to update the routing information at the intermediate nodes. Ants always sample complete paths, so that routing information can be updated in a pure Monte Carlo way, without relying on bootstrapping information from one node to the other.

ARP is a reactive multipath algorithm, designed along the principles of ACO routing. As in other proactive routing protocols, ARP does not maintain routes to all possible destinations at all times (like the original ACO algorithms for wired networks), but only sets up paths when they are needed at the start of a data session. This is done in a reactive route setup phase, where ant agents called reactive forward ants are launched by the source in order to find multiple paths to the destination, and backward ants return to the source to set up the paths. According to the common practice in ACO algorithms, the paths are set up in the form of pheromone tables indicating their respective quality. After the route setup, data packets are routed stochastically over the different paths following these pheromone tables

3.1.1 Reactive Path Setup

When a source node s starts a communication session with a destination node d , and it does not have routing information for d available, it broadcasts a reactive forward ant F_k . Due to this initial broadcasting, each neighbor of s receives a replica of the reactive forward ant F_k . The process of creating set of replicas from the same original ant is known as ant generation. The task of each ant is to find a path connecting s and d . At each node, an ant is either unicast or broadcast, according to whether or not the current node has routing information for d . The routing information of a node i is represented in its pheromone table T^i . The entry T^i_{nd} of this table is the pheromone value indicating the estimated goodness of going from i over neighbor n to reach destination d . If pheromone information is available, the ant will choose its next hop n with the probability P_{nd} .

If no pheromone information is available for d , the ant is broadcast. Due to this broadcasting, ants can proliferate quickly over the network, following different paths to the destination (although ants which have reached a maximum number of hops, related to the network diameter, are deleted). When a node receives several ants of the same generation, it will compare the path travelled by each ant to that of the previously received ants of this generation: only if its number of hops and travel time are both within an acceptance factor a of that of the best ant of the generation, it will forward the ant. Using this policy, overhead is limited by removing ants which follow bad paths, while there is still the possibility to find multiple good paths. However, it does have as an effect that the ant which arrives first in a node is let through, while subsequent ants meet with selection criteria set by the best of the ants preceding them, which means that they have higher chances of being rejected. This can lead to "kite-shaped" paths, as shown in Figure 2 (a). In order to obtain a mesh of sufficiently disjoint multiple paths as shown in Figure 2(b), which provides much better protection in case of link failures, we also consider in the selection policy the first hop taken by the ant. If this first hop is different from those taken by previously accepted ants, we apply a higher (less restrictive) acceptance factor.

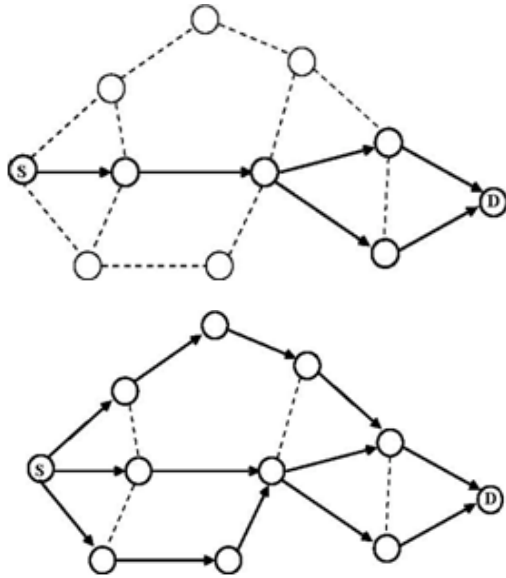


Fig2. Kite shaped paths in (a) and set of disjoint paths in (b)

7.3.2 Stochastic data routing

Once the paths have been setup, nodes in ARP forward data stochastically. According to this strategy, we do not have to choose a priori how many paths to use: their number will be automatically selected in function of their quality. The probabilistic routing strategy leads to data load spreading according to the estimated quality of the paths.

7.3.3 Link Failures

In ARP, each node tries to maintain an updated view of its immediate neighbors at each moment, in order to detect link failures as quickly as possible, before they can lead to transmission errors and packet loss. When a neighbor is assumed to have disappeared, the node takes a number of actions. In the first place, it removes the neighbor from its neighbor list and all the associated entries from its routing table. Then the node broadcasts a link failure notification message

3.2 ANT-BASED DISTRIBUTED CERTIFICATE REVOCATION PROTOCOL (AB-DCRP)

In our model of a VANET security system we assume the certifying authority responsible for generation and distribution of digital signatures for all nodes entering the network. This CA could

be the manufacturer, an international consortium or a government body.

Once a node becomes the part of a Vehicular network it monitors the performance (including authenticity and honesty) of other nodes using all the available sensor data. We assume this to be done by a black box [6]. Till the time it continues to be in touch with the RSI, it transmits this information back to the CA and makes it responsible for handling non-repudiation and banishing of any nodes from the network.

The problem we are trying to tackle is in the scenario when connection to the RSI is not available. This kind of a situation will occur in plenty for VANET's, especially during the early years of deployment and in rural areas. In this case a node will not be able to perform the following two jobs:

1. Request for the public key of a node that it wants to communicate with
2. Transfer information regarding the maliciousness of a node that it has detected.

Ant based request for certificates has already been proposed in [7] for MANET's and can be easily adapted for highly mobile networks like VANETS.

In addition to looking for certificates, ants can also be sent out to distribute information regarding honesty (or dishonesty) of particular nodes. Then each node, based on all information gathered through these ants, can take a decision regarding temporal revocation of certificates (till the time it comes in contact with the RSI and offloads this information to the CA) of that node. This decision can be made in a manner described i. In the next sub-section we describe a mechanism for disseminating maliciousness information.

3.2.1 Ants for maliciousness information dissemination

3.2.1.1 Defining Terms

Evidence: a metric defined as the amount of evidence one node has regarding the maliciousness of another node. Also called Revocation Evidence value.

Evidence, E_{ij} can be defined as the amount of evidence that i has against j .

Revocation Threshold: A reference value used in actually revocation of certificates. If the sum of all evidence against a node goes above the Revocation threshold, his certificate is revoked.

Forward Threshold: A reference value used in making a decision on whether a received pheromone should be forwarded to other nodes.

Revocation Table: A table maintained by each node containing the evidence value against each of its neighbors as suggested by its other neighbors and itself.

3.2.1.2 Algorithm Overview

At the core, this algorithm is essentially a reactive evidence distribution mechanism. Pheromones (pieces of evidence) are dropped by a node in two scenarios:

- It has revoked the certificate of another node. This again is possible in two ways
 - Either it has detected from its sensor data that a neighboring node has become malicious or
 - The sum of evidence provided by all its registered nodes (having an entry in its revocation table) against a certain node has gone above the revocation threshold
- When it updates the evidence of a node, if the evidence against this particular node by some node X is above a threshold value, it sends a pheromone to node X .

The algorithm consists of the following modules:

Setting it up

Each node within the network consists of a certificate revocation table. This table consists of a row and column for all neighbors as identified by the node. Whenever a node receives a pheromone regarding a node not present in the table, a row and column is appended to it. There is also a mechanism to remove neighbors whose values have not been edited for some threshold time period.

This table contain revocation evidence values which, as explained above, tell us the amount of evidence node i has against node j in the entry E_{ij} . E_{ij} is set to an initial value depending on the certificates issued and current environment of the node in question.

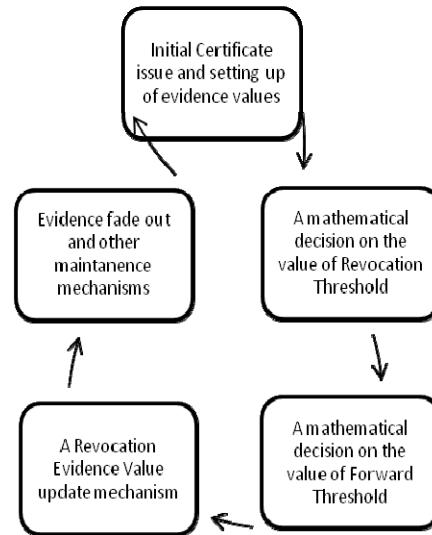


Fig 3. AB-DRP Algorithm Overview

Revocation Thresholds

The revocation quotient threshold (R_T) is a configurable parameter. A typical value would be 0.5 (majority vote). If $R_j > R_T$, key j becomes untrustworthy, i.e., messages signed with this public key are disregarded. A vehicle that has accumulated enough accusations against an attacker to reach the threshold and disregard messages from it is called hereafter a *warned vehicle*.

Forward Thresholds

An *initially warned vehicle* is a warned vehicle that disregards all bogus messages, because it has already reached the warned state before receiving the first message from M (because it has received enough accusations to reach the revocation threshold). To sustain the concept of initially we introduce the concept of forward thresholds. If on update of a revocation table, the evidence that i has against j in the table of x is greater than a specific value called the forward threshold, x will send a pheromone update to i . This threshold is again a configurable parameter, typically set to 0.37.

Pheromone Sequence Number	ID of revoked node (or affected node in case of forward pheromones)	ID of revokee node (or sender node in case of forwarded pheromones)	S / M Revoked by self or by metric	Revocation Evidence Value	R/F Revoked pheromone or forwarded pheromone
---------------------------	---	---	------------------------------------	---------------------------	--

Fig 4. Pheromone Packet Format

Evidence Update Mechanism

Pheromones being exchanged between nodes travel through the network and update evidence information being stored in revocation tables. Evidence update is based on a simple non-linear reinforcement-learning rule.

$$EU[n] = \frac{EU[x-1] + dE}{1 + dE}$$

Where dE can be defined as

$$dE = \frac{k}{f(\text{Evidence Value passed in pheromone})}$$

Where k is a constant and $f(x)$ is an increasing function of x .

Evidence Fade Out Mechanism

As time passes and nodes interact with newer nodes, evidence claimed against certain nodes becomes stale and as such staleness must be considered while taking a revocation decision.

To incorporate this factor, a cron job is set to update the evidence value. The update can be as simple as

$$EU' = \mu EU$$

Where μ is the evidence update ratio.

Pheromone Packet format

- Pheromone Sequence number is a unique identifier for each pheromone that is passed into the network.
- The universal identification number of the revoked node
- The universal identification number of the revokee node
- S/M : Whether the revocation was done by data analysis of sensor data (denoted by S) or by achievement of revocation threshold (denoted by M). In case of forward pheromones, the value as carried by initial revocation pheromone is copied.
- Revocation Evidence Value is the evidence value generated by the revokee node in case of S or the value achieved by accumulated evidence in case of M.

- R/F: Whether this pheromone was dropped in the case of revocation or in the case of achievement of forward threshold.

4. SIMULATION

4.1 Simulation Environment

We assume a 1000x1000 grid of a map on which nodes move randomly, emulating a vehicular network environment (VANETS). We have further assumed an urban city-like scenario where the vehicle density will be on the higher side. These nodes try and communicate with each other based on another random set of connections of TCP or UDP. We use a simple maliciousness model to inject malicious behaviour into the system. A node marks one of its neighbours as malicious based on this maliciousness model. At this point AB-DRP comes into the action and drops pheromones. We vary the following parameters and evaluate the said metrics for these scenarios. All simulations are run 20 times over and averaged.

4.2 Simulation Parameters

Vehicle Density: Defined as the number of vehicles per km, we vary this parameter by varying the number of nodes in the same 1000x1000 grid. This parameter also enables us to consider time of the day as the vehicle density varies depending on peak and no-peak hours.

Average Speed: The average speed with which the nodes move in the simulation environment decide the average active communication time i.e. the average amount of time two nodes will stay in communication range of each other. This is an important parameter in our case as the pheromone dropping only affects the nodes within the communication range of the node that has just identified a malicious node. Hence intuitively, more the average speed, more the active time, more the number of nodes receiving the pheromone containing revocation information, greater the number of initially aware nodes.

Transmission Range: As is the case with average speed, higher transmission ranges imply greater coverage for the pheromone and

more effective is our algorithm. We evaluate our algorithm for various values of the transmission range.

4.3 Simulation Metrics

Percentage of Initially Aware Nodes: An initially aware node is one who is aware of the maliciousness of a node before it comes into direct contact of it. So, as this metric we evaluate such nodes as a percentage of nodes who find out about the maliciousness of a node only after they come into direct contact with that node.

Average time to being warned: This metric signifies the amount of time that has lapsed between the time a node declares itself as malicious and another node declares itself as initially aware, an average value of that.

4.4 Results and Interpretations for AB-DRP

4.4.1 Effect of Vehicle Density

For low values of vehicle density we can see that the percentage of initially warned vehicles is low. This can be attributed to low pheromone drops as the node population is less dense. Hence the Ant Based algorithm works less efficiently as opposed to a generic revocation algorithm. As the density of nodes increases, there is a steep jump in the percentage of initially warned vehicles and it soon overtakes the percentage value of the original revocation algorithm. Hence we can conclude that in a highly dense scenario, as expected for urban traffic, our algorithm will work more effectively than a non-ant based algorithm.

Just as the percentage of initially warned vehicles is low at lower densities for AB-DRP, the amount of time it takes for the maliciousness information to reach a non-initially warned vehicle is high. This can again be attributed to low pheromone drop percentage in low density areas. From the graph, we can see that this value decreases to reach an acceptable and more favorable value with increasing densities. As the density value crosses 5 vehicles/km mark, our algorithm becomes more efficient than the earlier algorithm. This value is approximately the same vehicle density value that is required for the percentage of initially warned vehicles metric requires to perform better than the non-ant based algorithm.

From the above discussion we can conclude that our algorithm works on an average better

than the original DRP in a scenario where the vehicle density is higher than 5 vehicles per kilometer.

4.4.2 Effect of Average Speed

We evaluated our algorithm for increasing average speed of the vehicles in the system. For this analysis we assumed the vehicle density to be above the 5 vehicles per kilometre threshold. From the graph below we can observe that our algorithm works better than the non-ant based algorithm for most values of average speed. Their performance becomes nearly equal for very high as the link active times for the nodes go down. As a result of this the pheromones being dropped do not reach the more appropriate nodes – they have already moved out of the malicious nodes communication range.

Similar behavior is observed for average time to being warned for increasing average speed. For lower and average speeds, our algorithm works much better than the non-ant one becoming as bad for high speeds.

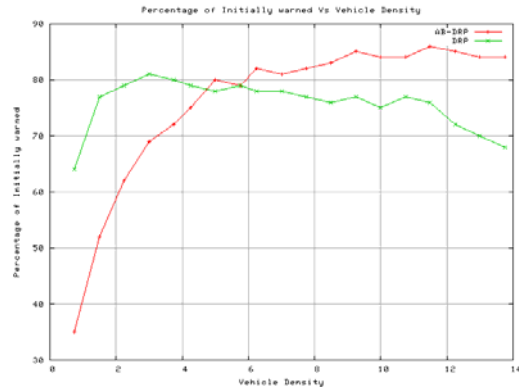
4.2.3 Effect of Transmission Range

As we increase the transmission range of the nodes, their coverage area increases and so does the number of nodes receiving pheromones being dropped by a node. As a result, the algorithm performs better for higher values of transmission range. This behaviour is shown in the following graph.

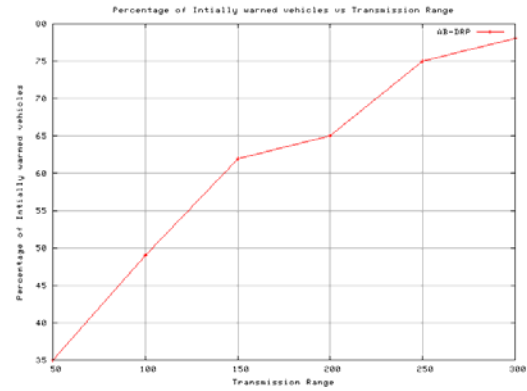
5 CONCLUSION AND FUTURE WORK

We have studied several bio-inspired algorithms to evaluate their applicability to the problems of networked systems, with special emphasis on security. Choosing Ant Colony Optimization as our algorithm of choice we have simulated its application to two specific problems of routing and certificate revocation in highly mobile and large scale networks (like VANETS). We have demonstrated its effectiveness over standard algorithms while also evaluating limits posed on the system by this algorithm.

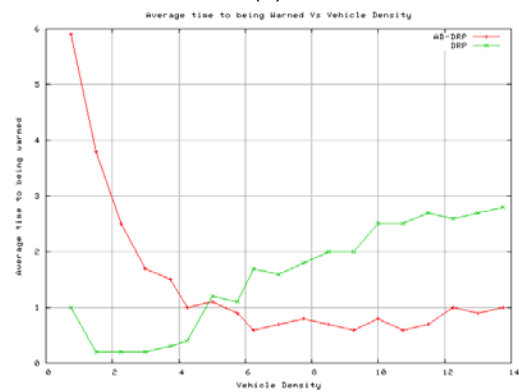
ACO being more of a design principle than an algorithm provides for several more areas of application in networked systems itself. Future work could involve dynamic revocation thresholds (R_T) as per vehicle status or priority. Also other biological algorithms can be used to implement the same.



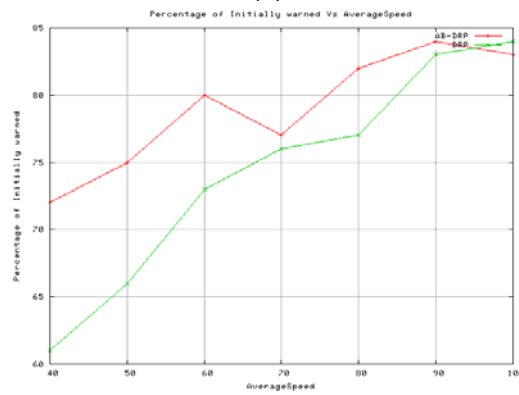
(a)



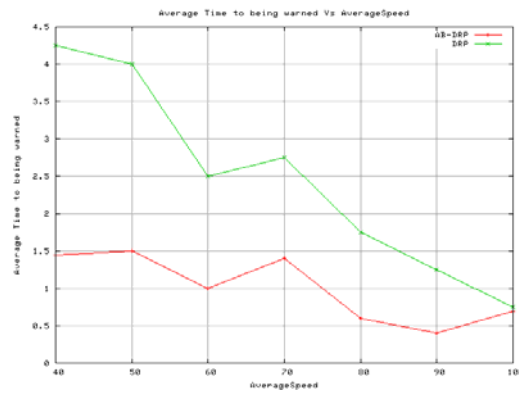
(e)



(b)



(c)



(d)

Fig 5 (a) Percentage of Initially warned Vs Vehicle Density (b) Average time to being warned Vs Vehicle Density (c) Percentage of Initially warned Vs Average Speed (d) Average time to being warned Vs Average Speed (e) Percentage of Initially warned Vs Transmission Range

REFERENCES

- [1] M. Raya, A. Aziz and J-P Hubaux. Efficient Secure Aggregation in VANETs. In VANET'06, September 29, 2006.
- [2] K. Plobl, T. Nowey, C. Mletzko. Towards a Security Architecture for Vehicular Ad Hoc Networks. Proceedings of the First International Conference on Availability, Reliability and Security (ARES'06).
- [3] M. Raya and J-P Hubaux. The Security of Vanets. In VANET'05, September 2, 2005.
- [4] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence – From Natural to Artificial Systems*. Oxford University Press, New York, 1999.
- [5] Kassabalidis, I.; El-Sharkawi, M.A.; Marks, R.J., II; Arabshahi, P.; Gray, A.A. Swarm intelligence for routing in communication networks. Global Telecommunications Conference, 2001. GLOBECOM apos;01. IEEE Volume 6, Issue , 2001 Page(s):3613 - 3617 vol.6
- [6] Golle, P.; Greene, D. H.; Staddon, J. Detecting and correcting malicious data in VANETs. Proceedings of the First ACM Workshop on Vehicular Ad Hoc Networks; 2004 October 1; Philadelphia; PA. NY: ACM; 2004; 29-37.
- [7] Tao Jiang John S. Baras Ant-based Adaptive Trust Evidence Distribution in MANET*
- [8] M. Raya, A. Aziz and J-P Hubaux. Certificate Revocation in Vehicular Networks