Exploiting Mobility for Trust Propagation in Mobile Ad Hoc Networks

Ningning Cheng, Kannan Govindan and Prasant Mohapatra
Department of Computer Science
University of California, Davis, CA 95616
Email: {chengni, gkannan, prasant}@cs.ucdavis.edu

***Abstract*—Trust plays an important role in protecting the security of mobile ad hoc networks. The node mobility brings challenge to trust propagation, where traditional graph-based propagation methods are difficult to apply. In this paper, we propose a trust establishment and propagation scheme by exploiting natural mobility of mobile ad hoc network nodes. We estimate the moving state of neighboring nodes, and select the group of nodes with higher movement diversity as the next hop. In this way, we expedite trust propagation with the help of mobility instead of treating it as a hurdle. Two classes of mobility models are considered: random direction model and cluster-based model. Extensive experiments are conducted on trust propagation performance using our scheme. Results show that the detection rate of highly untrustworthy node is improved by about 400% in random direction mobility model and about 300% in cluster-based mobility model compared to static case. Further, we observe that trust convergence time and communication overhead varies dramatically in different mobility models. Based on these observations, we discuss the usage of different mobility models and metrics in different trust applications in mobile ad hoc networks.**

I. Introduction

Mobile ad hoc networks introduce the possibility of an environment where multitudes of heterogeneous entities will participate in collaborative applications. Tactical networks in battle field and ad hoc networks in event detections are few to name. The goal of such a network-centric operation is to share information about an event so that best decisions can be made. The quality of information assessment received from unknown nodes and decision making can be enhanced using the notion of node level trust in the network. In a distributed environment generally every node assess the trust of its neighbours based on direct interactions. Direct trust evaluation is difficult to achieve in a large scale distributed network because the nodes may not have direct experience with far away nodes. In these networks direct trust generally gets propagated and used as a recommended trust in the rest of the network nodes which are far away, using the trust propagation phenomenon [1]–[6].

Widely followed approaches in the literature to propagate trust can be classified into two categories: Push-Aggregate based approach and Pull-Aggregate based approach [7]. In the Push-Aggregate based trust propagation process a participant node who has evaluated the trust of the target node and also willing to share the evaluated trust information, will push its evaluated trust value to other participants. Any participant node can collect its interest of trust information, and aggregate them in case the information is received through a different propagation path. This is basically flooding based approach and suffers from high communication over heads.

In the Pull-Aggregate based scheme a trust requesting node will find the trust information provider for the target node, and pull the trust information and aggregate different trust value from different providers. These schemes suffer from dynamic topology and the process of finding the trust information provider is a non-trivial problem in a mobile network. Further, if the propagation path is not the shortest path then the trust uncertainty and the propagation delay will be high.

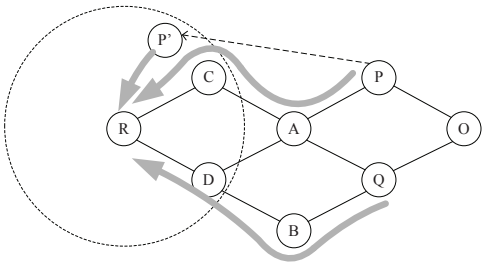

Fig. 1. Illustration of trust propagation

In this paper without using the Push-Aggregate and Pull-Aggregate schemes we exploit mobility in propagating the trust information. Due to mobility, node can meet more neighbors face to face (for example, in Fig. 1 if node P moves to P' within R's communication range). We exploit this phenomenon and propose a mobility-assisted trust propagation scheme, which can be adapted to both dense or sparse ad hoc networks.

Motivations for our scheme are:

1) Mobility can reduce the overhead in trust propagation. Using mobility we can avoid periodical flooding of trust information packets in the network. A node can only propagate the trust information to its direct neighbors, and can still achieve a better propagation rate because the neighborhood can change continuously due to mobility.
2) In traditional recommendation models, the trust information will get degraded when it passes through the nodes. However, mobility will give a chance for face to face meeting and hence can reduce the uncertainty in recommendation systems.
3) Mobility can generate more disjoint recommendation paths. Disjoint recommendation paths provide trust information from different sources which are independent.

Although, few previous researchers have considered mobility in trust propagation [8], the mobility pattern is restricted, and the moving trajectory is planned aprior. Nodes need to move intentionally in a controlled pattern in order to achieve better trust convergence, which means extra moving overhead. In addition, some nodes in ad hoc networks may not follow the rules defined in the protocol and may move arbitrarily. In this paper, we do not assume a fixed moving pattern as considered in the existing literature.

The basic idea of our approach is described as follows. Every node makes trust evaluation on its neighbours based on both direct observation and trust recommendations from other neighbours. These trust informations will be carried

along when the node is moving. Whenever this node meets new neighbors, it will communicate the trust information to those nodes which have high direct trust and have more chance to meet other new nodes. This selective forwarding approach is based on the relative direction and relative speed between the sender and receiver. It intents to balance the tradeoff between flooding overhead and trust propagation time. This way the trust information will be propagated to the rest of the network nodes. We propose a systematic approach for the trust propagation. We conduct extensive analysis using different mobility and trust propagation metrics. Two mobility models are considered for performance evaluation: Random direction model and cluster based mobility models [4]. We show that nodes can obtain trust information with higher quality by our selective forwarding approach in mobile environments.

The rest of the paper is organized as follows. Section II revives some of the related work. Section III explains the trust properties, the proposed trust system architecture and design objectives. Functionalities of various components of our proposed trust system are explained in Section IV, Section V and Section VI. Extensive analysis of impact of various mobility components on trust propagation performance are provided in Section. VII. Concluding remarks are given in Section VIII.

II. Related work

Trust propagation in the mobile wireless networks using the concept of small world [9] is proposed in [10]. This approach can be used only in certain specified self-organized ad hoc networks. There have been substantial works on trust propagation based on recommendation [11]–[13] which are basically flooding approaches. Propagation of the security credentials such as cryptography keys, trust information by exploiting mobility is analyzed in [14], where nodes exchange trust information as soon as they are connected. The performance of this strategy depends on the mobility pattern, density of the nodes and other related parameters. Trust propagation based on spreading activation models is proposed in [15], [16]. Spreading activation is a method for searching trust values or any intended values of nodes in the networks. Our approach considerably differs from the above stated as we use selective forwarding for trust propagation by exploiting mobility. There have been other work in analyzing the impact of mobility on network performance including network security, connectivity, capacity [17]–[19]. We can classify in a broader sense the mobility models considered in literature into two classes: Nature moving models and Controlled moving models. The commonly used nature walking models include independent entity mobility models (e.g., random direction, random way point). In controlled moving models moving trajectory of mobile nodes are predefined. Mobile nodes follow certain patterns such as town hall model [8]. We analyze propagation of trust in these two categories of mobility models.

III. Exploiting mobility for trust propagation

A. Definitions and properties of trust

Based on [1] we define the trust of node i on node j as the node i's expectation on future behavior of node j. We consider two main properties of trust that can be used in trust propagations: Transitivity and Composability [20], [21]. Transitivity implies that trust can be passed along a path of trusting users. If A trusts B and B trusts C, it can be inferred that A trusts C at a certain level. Composability implies that trust information available from different paths can be composed together (aggregation) for inferring the final trust.

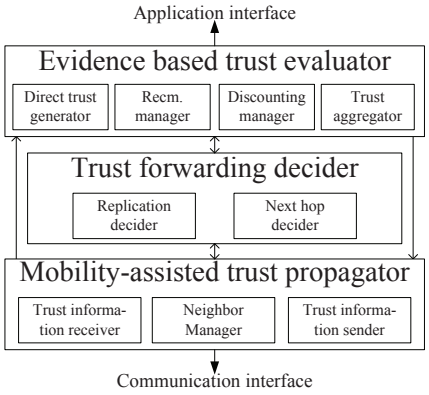

Fig. 2. System architecture

B. System architecture overview

Fig 2 shows the system architecture of our mobility assisted trust propagation framework. It consists of three blocks: *Evidence-based trust evaluator*, *Mobility-assisted trust propagator* and *Trust forwarding decider*.

Evidence-based trust evaluator interacts with the upper application layer (such as sensor network monitoring, tactical network decision making or secure routing). It has four sub blocks: *Direct trust generator*, *Recommendation manager*, *Trust discounting manager* and *Trust aggregator*. *Direct trust generator* collects evidence by observing its direct neighbors and evaluates their direct trust performance. *Recommendation manager* collects recommendation trust from its neighbors. *Trust discounting manager* ages the historical opinions by a discount factor, to distinguish old trust opinion from new trust opinion and gives high weight to the fresh trust opinion. *Trust aggregator* aggregates both the direct trust and recommendation trust to obtain a single trust value.

Mobility-assisted trust propagator consists of three sub blocks *trust information receiver*, *Neighbor manager* and *trust information sender*. *Trust information receiver* gets the trust information from neighbor nodes' Evidence-based trust evaluator to propagate this information to other neighbor node. The neighbor node for trust propagation will be decided by the *Trust forwarding decider*. *Neighbor manager* collects neighbor node's information such as ID and mobile attributes, and passes them to *trust forwarding decider* as algorithm inputs.

Trust forwarding decider consists of sub blocks *replication decider* and *next hop decider*. *Replication decider* decides how many replication of trust messages to be disseminated, and *next hop decider* decides which neighbors to send trust message, *trust information sender* takes these algorithm output and disseminate the trust massage. This architecture is maintained in each node, in order to balance the tradeoff between flooding overhead and trust propagation time, and also to reduce the degradation of trust information in trust propagation.

C. Design objectives and metrics

The overall target of our design is to provide trust information within certain accuracy about any node. We design our system with the following objectives. First design objective is to achieve good *Malicious detection rate*, which is the number of attackers recognized by trust requesters. Less *Trust propagation time* is another objective because trust information should be propagated before expires, this objective is especially important when trust is used in delay critical applications. The third objective is to reduce *Trust propagation overhead*, the number of trust messages used in propagation,

which is also an important metric in mobile environment. Reducing the *Uncertainty of trust information* is considered as the final objective in trust propagation since it reflects the quality of the trust information.

In the following sections we explain in detail about the various blocks of our proposed system in Fig. 2.

IV. Evidence-based trust evaluator

We use Bayesian inference based reputation system for direct trust evaluation. First, evidence based trust evaluator in a node distributively monitors a target neighbor's behavior and quantify it as *evidence* (α and β). The notations of trust information and their descriptions are shown in Table I.

TABLE I
NOTATIONS IN TRUST EVALUATED

Field	Description	Value
α	evidence of good behavior count	Integer
β	evidence of bad behavior count	Integer
u	uncertainty of trust information	$12Var(beta(\alpha,\beta))$
b	belief of trust information	$\alpha/(\alpha+\beta)(1-u)$
d	disbelief of trust information	$\beta/(\alpha+\beta)(1-u)$
$\Gamma()$	gamma function	–

Each node uses a *direct observation buffer* to keep *event evidences* α and β. A *recommendation buffer* is used to keep the second hand recommendations from neighbors. The buffer can be updated time to time to guarantee a fresh trust value.

Direct trust generator: For the *event evidence* collection, nodes count neighbors' good behavior α and bad behavior β respectively. For instance in routing problem α could be the number of successful packets forwarded by the target node to proper destination as per the routing table and β could be the number of packets dropped or wrongly forwarded by it. These two values will be used as parameter of a beta distribution as follows:

$$P(t) = beta(\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} t^{\alpha-1}(1-t)^{\beta-1} \qquad (1)$$

Now the trust opinion will be generated as a triplet $< b, d, u >$, where b stands for node's belief that the neighbor will behave good in next time, d is the node's disbelief for a good behavior and u is the uncertainty of the opinion. The expectation of the beta distribution ($\alpha/(\alpha+\beta)$), is used to derive b and d. The variance ($Var(t) = \alpha\beta/(\alpha+\beta)^2(\alpha+\beta+1)$), is used to derive u. For brevity we refer [8] for detailed calculation of b, d and u.

Recommendation manager: The indirect recommendation trust is processed based on the transitivity property of the trust. Recommendation manager receives the recommended trust value from the Mobility-assisted trust propagator.

Trust discounting manager: The trust value is aged by time. Due to mobility, ad hoc networks usually have dynamic, sometimes rapidly changing topology. The evidence in the network will decay when the nodes move out of the communication range. Therefore the older recommendations need to be aged. In order to differentiate an older recommender from a newer recommender, an aging factor ρ is used. Here ρ is a real number ranges in $(0, 1)$. In our experiment, the recommendation trust value is updated time to time as follows: $b_{new} = \rho b_{old}$, and $d_{new} = \rho d_{old}$.

Trust aggregator: This module aggregates first hand evidence and second hand evidence when node has both direct trust evaluation and trust recommendation about another node. The recommendation evidence is aggregated by the ratio of the recommender's recommendation trust degree. For example in Fig 1 assume that node P has first hand evidence $< \alpha_O^P, \beta_O^P >$ on node O. P's one hop neighbor Q also has evidence $< \alpha_O^Q, \beta_O^Q >$ on node O and send this evidence to P as recommendation. In this case, P aggregates Q's evidence into a new evidence based on Dempster-Shafer belief theory [1] as follows:

$$\alpha_{new} = \alpha_O^P + \frac{2\alpha_Q^P\alpha_O^Q}{(\beta_Q^P+2)(\alpha_O^Q+\beta_O^Q+2)+2\alpha_Q^P}$$

$$\beta_{new} = \beta_O^P + \frac{2\alpha_Q^P\beta_O^Q}{(\beta_Q^P+2)(\alpha_O^Q+\beta_O^Q+2)+2\alpha_Q^P}$$

where $< \alpha_Q^P, \beta_Q^P >$ is the evidence of P on Q. In other words, when multiple recommendation nodes exist, a trust requester node can aggregate the recommendation values from recommenders weighted by their own trust value.

After the newly aggregated *event evidence* is calculated, the recommender's *recommendation evidence* is then updated using Algorithm 1:

Algorithm 1: Recommendation evidence update for recommender j

```
For recommender j in node i
if α_new/(α_new + β_new) > 0.5
   if α_j/(α_j + β_j) > 0.5      α_j ++
   else                          β_j ++
else
   if α_j/(α_j + β_j) > 0.5      β_j ++
   else                          α_j ++
```

V. Mobility-assisted trust propagator

Mobility-assisted trust propagator has three subcomponents: *trust information receiver*, *trust information sender*, and *neighbor manager*. The functionality of these three subcomponents are given below:

Trust information receiver: This module receives trust information from neighbor nodes on target node. It then filters the redundant information based on the sender, target node and message arrival time t. Here the redundant information could be multiple similar information or a small portion of the information which is completely contradicting with the rest of the information. These redundant information will be deleted without any processing. The filtered information will then be passed to *Evidence based trust evaluator* for trust aggregation or recommendation discount.

Neighbor manager: It records neighbor's information including the neighbor's ID, velocity ($\vec{v}(t)$) and position. The position of the node can be determined using some of the infrastructureless localization techniques e.g., trilaterization. Every node is assumed to observe the neigbour for certain duration to collect trust evidences. By making multiple measurement of position during this evidence collection time window, a node can determine the velocity of the neighbour including speed and directions. These measurements are made continuously and *Neighbor manager* refreshes the neighbor's mobility attributes every time and sends them to *trust forwarding decider*.

Trust information sender: This module obtains the aggregated trust value from the Evidence based trust evaluator and also the number of replication as well as the ID of the next hop node from the Trust forwarding decider. Using this information the *Trust information sender* disseminates the trust information.

VI. Trust forwarding decider

TABLE II
NOTATIONS IN TRUST FORWARDING DECIDER

Field	Description
$\vec{v}_i$	node i velocity vector
v_i	node i's speed
$RS_{i,j}$	relative speed between node i and node j
$RD_{i,j}$	relative direction between node i and node j
N	the total number of network nodes
t_0	trust message generation time
T_{expire}	time when the trust information is expired in trust propagation

Replication decider: For trust propagation, a mobile node can transmit trust messages whenever it meets a new neighbor. The problem with this straightforward method is that the network is overwhelmed by trust messages and hence causes large overhead. What makes it even worse is that trust providers may not have an idea whether the trust information has reached the requester or not. The trust provider may keep disseminating the trust information even after the trust information has been used. In our trust model *replication decider* limits the time of trust information propagation. We use T_{expire} to denote that the trust information is expired and can no longer valid.

The replication decision algorithm works as follows. When i receives a trust message at time t_i which was originally generated at time t_0 and if $t_i - t_0 > T_{expire}$, then node i will drop the message. Otherwise it will propagate the trust information.

Next hop decider: Mobility patterns and components can have different influences on trust propagation. If we consider the moving direction, a mobile node moving towards the trust requester has higher chance to meet the requester, hence propagates the trust information in quicker time. If we consider moving speed, a node with a different moving speed can propagate trust information to more nodes whom trust provider will never meet while moving its in own trajectory.

Based on this observation, we propose to use selective trust forwarding algorithm in *next hop decider* block. Selective trust forwarding algorithm aims to select the next hop neighbors based on the relative speed and relative direction. The relative direction between i and j is represented by $RD_{ij} = \frac{\vec{v}_i \vec{v}_j}{|\vec{v}_i||\vec{v}_j|}$ which is a cosine of angle between two nodes mobility trajectory. The relative speed between two nodes i and j is represented by $RS_{ij} = \frac{\{min|\vec{v}_i|,|\vec{v}_j|\}}{\{max|\vec{v}_i|,|\vec{v}_j|\}}$. The approximate value of $\vec{v}_i$ can be calculated by observing the neighbor for while and then employing the localization techniques.

We filter the next hop neighbors based on the relative speed and directions. As per our definition the nodes with high RD and RS values are likely to move in the similar trajectory, therefore, less useful for trust propagations. Using the RD and RS *Next hop decider* runs a selective trust forwarding algorithm as given in Algorithm. 2 to select the forwarding neighbor.

Algorithm 2: The selective trust forwarding algorithm running on node i

```
for mobile node j within i's communication range
calculate RD_{i,j}
  if RD_{i,j} < 0
    send trust information from recommendation buffer to node j
  else
    calculate RS_{i,j}
      if RD_{i,j} × RS_{i,j} < 0.5
        send trust information from recommendation buffer to node j
```

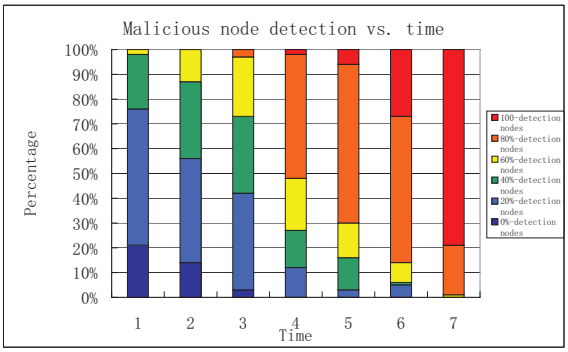

Fig. 3. Malicious node detection in random walk model

VII. Simulation and Evaluation

We conduct extensive simulations to evaluate the performance of our trust system using malicious node detection, propagation time, propagation overhead and trust uncertainty as performance metrics.

A. Mobility environment settings

For illustration purpose we consider two mobility models: random direction model and clustering based group model. The most significant difference between these two models is the speed correlation in mobile nodes. We generate the *speed correlation* based on the average $RS \times RD$ between any two pair of the nodes. We define the network's *speed correlation* is 0, if average $(RS \times RD)$ is a negative value, and *speed correlation* equals to average of $(RS \times RD)$ if it is a positive value.

B. Malicious node detection

First we analyze the malicious detection rate in random direction mobility model. Nodes exchange trust value at the beginning of each epoch. We assume 5% of the nodes are malicious and placed at random locations with high β evidences and low α evidences. Malicious node detection rate is defined as out of 100 nodes how many nodes detect few or all five nodes with high β value and low α value. Epoch 1 represents a static scenario, where nodes start to move. The trust information is processed and transmitted based on the mobility-assisted trust system proposed in Section IV, V and VI. The results obtained for malicious node detection is shown in Fig 3. In Fig 3 at each time epoch we have shown the number of nodes detecting and their malicious node detection rate using various color bars. At each epoch, we classify nodes into six categories based on their malicious node detection rate: nodes detecting 0% malicious nodes are called as 0% detection nodes, nodes detecting 20% malicious nodes are called as 20%-detection nodes and so on. We run the simulations 100 times and average the performance. In Fig. 3 at epoch 1 we can see that 20% nodes are not detecting any malicious nodes (dark blue color bar). However, at epoch 7 we can see that all nodes can recognize at least few of the malicious nodes and 80% nodes detects all five malicious nodes (100% malicious detection) due to proposed trust propagation. We can infer that malicious detection rate improved over 400% in epoch 7 compared to epoch 1 in case of random direction model. The reason for this performance improvement is that mobility reduces long distance recommendations. Long distance recommendation can generate trust information degradations which will make the trust information less useful.

To analyze performance in a cluster mobility model we place 100 nodes into 10 cluster cells uniformly random in a

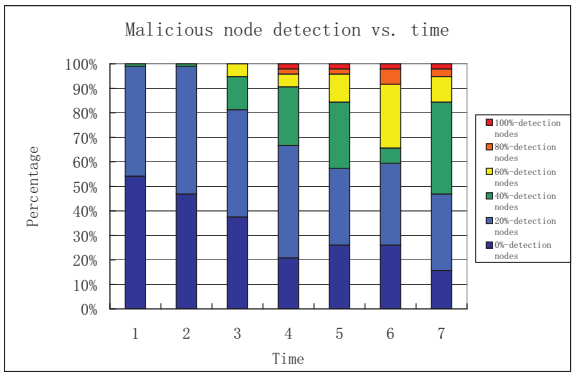

Fig. 4. Malicious node detection in clustering model

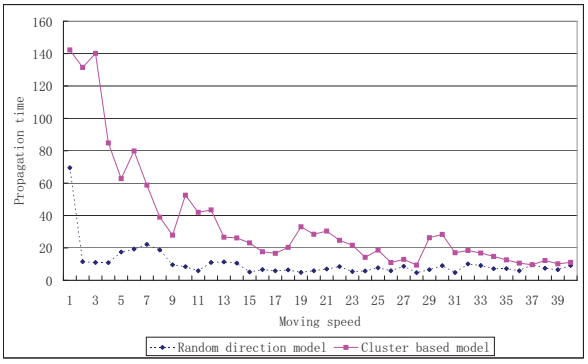

Fig. 5. Propagation time comparison between different mobility models

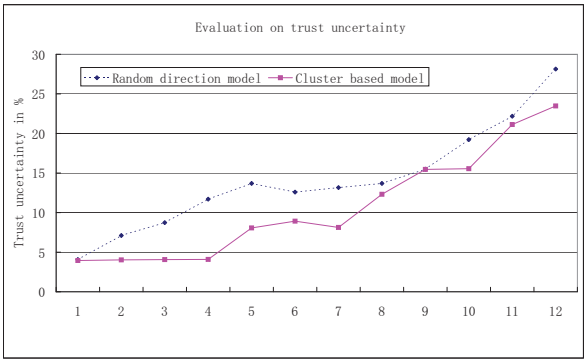

Fig. 6. Trust uncertainty in different mobility models

100×100 area. Each cluster cell has on an average 10 nodes. Like the previous setup, time is divided into 7 epochs and epoch 1 represents static scenario. In each epoch, nodes move randomly inside its own cluster cell by choosing the random speed ranges in $[0, 5]$ units. Less speed range is chosen in this setup to contain the nodes in their respective cluster cells. In each cluster cell, a cluster head is elected which can move out of the cluster to exchange trust information with other cluster members. After the trust information exchanges, all the cluster head go back to their original cluster and update trust information with their cluster members. In the result, we see that the malicious detection rate improved over 300% in epoch 7 compared to epoch 1 (static setup).

From these two experiments, we observe that both random direction and cluster based mobility models significantly improve the trust propagation performance compared to static deployment. For malicious node detection, the random direction model converges faster than cluster based mobility model with a higher malicious detection rate. The reason for the better performance improvement with random mobility model is that the nodes can move randomly all over the deployment region and hence the trust propagation as well as malicious node detection rates are higher and faster. However, in cluster based mobility models the mobility pattern is constrained in the cluster cell and hence nodes have only restricted local knowledge through the cluster head.

C. Propagation time in different mobility model

In Fig 5, we compare the trust propagation time in both the mobility models under consideration. In each model, nodes move at a speed ranges in $[v_{avg} - v_{var}, v_{avg} + v_{var}]$. We change the values of v_{avg} in both the models from 1 to 40 units randomly and set $v_{var} = \frac{v_{avg}}{4}$. We randomly pick node ID-16 as the trust requester, and node ID-85 as the trust provider. The trust propagation stops as soon as node ID-16 receives the trust message generated from node ID-85. The comparison of the trust propagation time is given in Fig 5. This result shows that in most cases, the trust propagation time is at least twice in cluster based model than in random direction model.

We also measure the average speed correlation between nodes in the given setup. We observe that the average speed correlation between nodes in the case of cluster based mobility model is 0.23 and 0 in the case of random direction model. Speed correlation considers the average relative speed of all pair of nodes in the network. It reflects how similar the moving traces are. Therefore, the lesser the speed correlation is, the more we can exploit the mobility in trust propagation. Random direction model achieve better convergence performance due to this reason.

D. Trust uncertainty in different mobility model

In this experiment, we compare the trust uncertainty due to trust propagation in two mobility models. We have randomly generated 12 network topologies for the two mobility models. We set node ID-16 as trust information provider and node $ID-85$ as trust information requester in this experiment, too. The trust uncertainty is calculated as normalized variance of Beta distribution given in Eq. (1) [8]. From Fig 6, we can see that although the propagation time is longer in group based model, the trust uncertainty is lesser compared to the random direction model. This result matches with the fact that in cluster based model, a node has more direct neighbors making one hop recommendations. Since uncertainty is reduced when the number of evidences (both first hand evidence and second hand evidence) increases, the cluster based model performs better than random direction model in this experiment.

E. Propagation overhead in different mobility model

In order to evaluate the trust propagation overhead, we observe the number of packet exchanges in propagating the trust from the trust provider to the trust requester. The trust propagation time starts when the node ID-85 dissipates the trust information and ends when the requester node ID-16 receives the information. We show the total message overhead in Fig 7 for both the mobility models. The result shows that the number of packets exchanged (over head) in random direction model is almost twice as compared to cluster based mobility model. Random direction mobility nodes have more chance to meet new neighbors, so nodes need to exchange more packets. However, cluster based mobility model can take full advantage of our proposed selective trust forwarding algorithm as the speed correlation is high in cluster based mobility model.

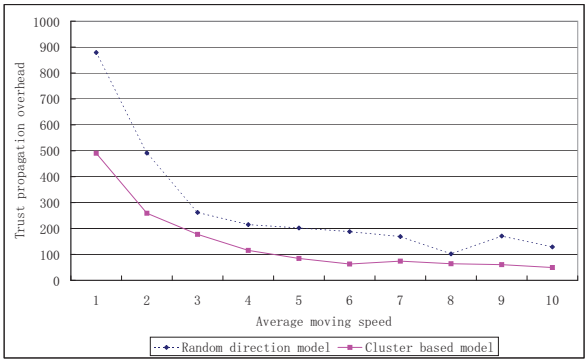

Fig. 7. Trust propagation overhead in different mobility models

Fig. 8. Walking distance comparison

Therefore, the number of neighbor nodes filtered out is high in cluster mobility models and hence less packet exchange.

From this experiment, we can infer that albeit longer propagation time, correlated mobility model has less overhead and uncertainty.

F. Walking distance comparison in different mobility model

In this experiment, we compare the average walking distance in two mobility models. Average walking distance is defined as the average distance traveled by each node during the trust propagation. From Fig 8, we can observe that random direction model has average walking distance more than 33% as compared to cluster based model. Combining the previous results, we can infer that the trust propagation time is reduced in random direction walking model due to a longer average walking distance by all nodes.

VIII. Discussions and Conclusion

In this paper, we have proposed a mobility assisted trust system architecture and performed extensive simulations using two mobility models. Results show that the average malicious detection rate improved by about 400% in random direction model and by about 300% in cluster based model. Furthermore, the convergence process is faster in a random direction model by about 100% compared to that of cluster based model. On the other hand, with respect to trust uncertainty and propagation overhead, the cluster based model outperforms the random direction model. In the end, we concludes that the cluster based model is a better choice in resource limited applications where delay is not a critical issue. If the goal is to reduce trust uncertainty in a trust application, cluster based moving model is more suitable for trust propagation. Random direction model is more suitable for the scenarios where delay is a critical metric in trust information propagation. We consider detailed performance evaluation using theoretical approaches and experimental setup for future work.

Acknowledgment

This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official polices, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copy right notation here on.

References

[1] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Reputation-Based Framework for High Integrity Sensor Networks," in *ACM Transactions on Sensor Networks, TOSN*, vol. 2845, pp. 66–77, 2008.

[2] M. Virendra, M. Jadliwala, M. Chandrasekaran and S. Upadhyaya, "Quantifying Trust in Mobile Ad-Hoc Networks," in *Proceedings of the IEEE International Conference on Integration of Knowledge Intensive Multi-Agent Systems, KIMAS 2005*, pp. 65–71, 2005.

[3] Y. Sun, Z. Han, W. Yu and K. J. R. Liu, "A Trust Evaluation Framework in Distributed Networks: Vulnerability Analysis and Defense Against Attacks," in *IEEE INFOCOM*, pp. 230–236, 2006.

[4] T. Camp, J. Boleng and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research," in *Wireless Communication and Mobile Computing*, vol. 2, pp. 483–502, 2002.

[5] L. Eschenauer, V. D. Gligor, and J. Baras, "On Trust Establishment in Mobile Ad-hoc Networks," in *Proceedings of Security Protocols Workshop*, vol. 2845, pp. 47–66, April 2002.

[6] H. Zhu, F. Bao, R.H. Deng, "Computing of Trust in Wireless Networks," in *Proceedings of IEEE 60th Vehicular Technology Conference*, pp. 2621–2624, 2004.

[7] "Tight Bounds on Information Dissemination in Sparse Mobile Networks." http://arxiv.org/abs/1101.4609, 2011.

[8] F. Li and J. Wu, "Mobility Reduces Uncertainty in MANETs," in *Proceedings of the The 26th IEEE International Conference on Computer Communications: INFOCOM*, pp. 1946–1954, 2007.

[9] E. Gray, J. marc Seigneur, Y. Chen, and C. Jensen, "Trust propagation in small worlds," in *In Proc. of 1st Int. Conf. on Trust Management (iTrust03)*, pp. 239–254, 2003.

[10] H. Zhu, F. Bao, R.H. Deng, "Computing of trust in wireless networks," in *Proceedings of IEEE 60th Vehicular Technology Conference*, pp. 2621–2624, 2004.

[11] F. E. Walter, S. Battiston and F. Schweitzer, "A model of a trust-based recommendation system on a social network," vol. 16, pp. 57–74, Feb. 2008.

[12] R. Andersen, C. Borgs, J. Chayes, U. Feige, A. Flaxman, A. Kalai, V. Mirrokni, and M. Tennenholtz, "Trust-based recommendation systems: an axiomatic approach," in *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pp. 199–208, 2008.

[13] J. O'Donovan and B. Smyth, "Trust in recommender systems," in *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, pp. 167–174, 2005.

[14] S. Capkun, J. P. Hubaux and L. Buttyan, "Mobility Helps Security in Ad Hoc Networks," in *The 4th ACM international symposium on Mobile ad hoc networking, Mobihoc'03*, pp. 46–56, 2003.

[15] C.-N. Ziegler and G. Lausen, "Spreading activation models for trust propagation," in *Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE'04)*, pp. 83–97, 2004.

[16] C. N. Ziegler, and G Lausen, "Propagation Models for Trust and Distrust in Social Networks," *Information Systems Frontiers*, vol. 7, no. 4-5, pp. 337–358, 2005.

[17] T. K. Madsen and F. H. P. Fitzek and R. Prasad, "Impact of Different Mobility Models on Connectivity Probability of a Wireless Ad Hoc Network," in *Proceedings of International Workshop on Wireless Ad Hoc Networks*, 2004.

[18] A. Jindal and K. Psounis, "Fundamental Mobility Properties for Realistic Performance Analysis of Intermittently Connected Mobile Networks," in *Proceedings of IEEE PerCom Workshop on Intermittently Connected Mobile Ad Hoc Networks (ICMAN)*, pp. 59–64, 2007.

[19] T. Spyropoulos, A. Jindal and K. Psounis, "An analytical Study of Fundamental Mobility Properties for Encounter-based Protocols," *Int. J. Auton. Adapt. Commun. Syst.*, vol. 1, no. 1, pp. 4–40, 2008.

[20] J. A. Golbeck, "Computing and applying trust in web-based social networks," in *University of Maryland at College Park (2005)*.

[21] K. Nordheimer, T. Schulze, and D. Veit, "Trustworthiness in networks: A simulation approach for approximating local trust and distrust values," in *IFIP International Conference on Trust Management-2010*, pp. 157–171.