Wireless Pers Commun (2018) 99:371–401
https://doi.org/10.1007/s11277-017-5107-1

Mobility Aware Clustering Scheme with Bayesian-Evidence Trust Management for Public Key Infrastructure in Ad Hoc Networks

V. S. Janani[1] · **M. S. K. Manikandan**[1]

Published online: 13 December 2017

Abstract The autonomous nodes in Mobile Ad Hoc Networks (MANETs) are vulnerable to attacks ranging from passive to active, due to the dynamic mobility paradigm. Earlier, in order to handle this scalability issue; routing protocols were developed as a traditional solution that utilizes a flat organization to establish routes among the mobile nodes. However, the flat routing schemes present a static security restrains with single point dependency, which is inappropriate for MANET. Also, these security solutions failed to overcome certain drawbacks in managing a Public Key Infrastructure framework as: Nearest-Neighbour Search problem and cluster construction and maintenance complexities. Hence, an optimal hierarchical structure by overwhelming the drawbacks in the existing mechanisms should be regulated for solving the scalability problem in terms of topology control in MANET. In this paper, an efficient Hybrid Trust based Mobility Aware Clustering (HTMAC) strategy is proposed as the distributed trust based hierarchical solution for uncertain MANET system. Each node in the proposed scheme computes the trustworthiness of others with respect to the direct observations and recommendations, to develop a self-organized framework. To encounter challenges in node cooperation and security, hybrid trust computation is calculated in the proposed scheme. The originality of the proposed work combines clustering metrics with Voronoi hexagonal structure and Bayesian-Evidence trust management to predict the distributed security solution. Relevant simulation results demonstrate that our clustering model guarantees a secured and mobility-adaptive ad-hoc network with trustworthy mobile nodes. Thus the HTMAC scheme provides an efficient security solution that incorporate the promising features of clustering, Voronoi diagram and trust mechanisms.

Keywords Clustering · MANET · Mobility · Security · Public Key Infrastructure · Direct observations · Recommendation · Voronoi diagram · Hexagonal structure · Bayesian-Evidence trust management

✉ V. S. Janani
Jananivs1987@gmail.com

[1] Department of ECE, Thiagarajar College of Engineering, Madurai 625015, India

1 Introduction

With the rapid increase in wireless networks, in specific Mobile Ad Hoc Networks (MANETs), have gained significant attention in recent years. The MANET allows instantaneous communication with unrestricted mobile nodes in the absence of any predefined infrastructure. Hence, the MANETs are greatly put into practice for emergency communication in military and disaster management. The dynamic mobility paradigm of these networks makes a multi-hop framework, without any centralized maintenance. Unlike any fixed network, the autonomous nodes in MANET are vulnerable to attacks ranging from passive to active, owing to the differing topology. In other words, a MANET faces scalability issue due to their inherent mobility characteristics. On that account, it is difficult to provide a static security solution with single point dependency. Hence, a distributed security solution that can adapt the intrinsic features of MANET should be regulated.

The researches on distributed systems should therefore organize the mobile nodes with a hierarchical security solution to achieve performance guarantee in emergency applications. To manage these uncertain nodes, clustering has been widely applied as a typical hierarchical structure for solving the scalability problem in terms of topology control in MANETs [1–3]. A cluster configuration acts as a virtual backbone for routing, certificate management and key management with efficient spatial reuse, for deploying a Public Key Infrastructure (PKI) in MANET over a finite region [4, 5]. With the objective to co-ordinate and collaborate the dynamic nodes for establishing scalable system, the self-organization concept is integrated in the distributed clustering architecture. This eliminates the single point of failure of centralised methods and provides PKI system with self-organization, self- configuration and self-management attribute to adapt the changing network conditions.

Moreover, it is necessary that the distributed solution should provide the required security services and guarantees an invulnerable system, for co-operative information sharing in MANET. This can be practicable only if all the nodes behave in a trustworthy manner. In recent years, trust, has been suggested as an effective mechanism to encounter network challenges for constructing an optimized secured distributed self-organized system [6, 7]. The trust in MANET is the subjective appraisal on the behaviour of a node by its neighbouring nodes. It reflects the belief as well as expectations on the reliability of information sends by any node. Nevertheless, there are several drawbacks in establishing a distributed trust based PKI communication system to cluster the ad hoc networks. Some of them are:

- The traditional clustering techniques usually assume that the position of nodes is known accurately, which is impractical in MANET.
- The typical clustering techniques, in which the nodes are grouped depending on the physical position, requires additional computation steps whenever mobility changes. These clustering of uncertain nodes increases the complexities as well as the communication and computation cost for MANET.
- The clustering methodologies in MANET are generally designed to handle node-valued data and thus cannot manage uncertainty of data, due to the dynamic mobility of nodes.
- The information sharing among cluster nodes immensely depend on the location of mobile nodes. Therefore, the computation of distance between such nodes is significant for any co-operative communication, which is complicated in conventional clustering techniques.
- It is hard to establish a complete resilient system with underlying self-organized trust based clustering with frequent link failures.

Therefore, it is clear that the drawbacks of the long-established clustering techniques should be minimized in order to make the PKI based security viable for node to node security deployment. On this pursuit, the proposed research work concentrated in developing a distributed security solution for self-organized PKI framework, which quantifies nodes behaviour in the form of trust.

The most dominant problem in distributed trust computation is how to combine the individual observations from multiple nodes to calculate the trust of any node. The primary objective of the proposed research work is to adapt the active topology with a hybrid trust computation model. This hybrid trust establishment follow the self organizing property that no trusted third party involved in the trust computation among the participating nodes. This is achieved by combining the direct observations and the recommendations obtained from the neighbouring nodes. The direct trust is evaluated and verified using Bayesian theory with beta distribution functions. Whereas, the Dempster–Shafer (DS) evidence Theory combines the multiple observations called recommendations, in order to calculate the indirect trust. Here the observations are considered as evidences which can also be in the form of misbehaviour (variation from expected trust behaviour).

The proposed clustering model guarantees better performance by improving the dynamic re-configurability, scalability and security. An optimal solution for various cluster inherent issues such as: size of the cluster, probability of node being clustered, node residence time, cluster age, cluster overhead and rate of control messages is achieved with the proposed model. A header is elected for each cluster to secure the intra-cluster and inter-cluster communication considering all the network functionalities and node mobility-instigated events. The popular geometric structure, Voronoi diagram, is used in the proposed clustering methodology to overcome the neighbour search problem in finding the number of nearest neighbouring nodes and reducing their expected distance computations [8, 9]. Unlike the conventional circular shape of clustering, a hexagonal shape is presented to partition the MANET area into adjacent and non-overlapping group of nodes with improved spatial reuse [10].

This paper is structured as follows. In Sect. 2, the relevant works on clustering and trust in MANET are described. Section 3 narrates the motivation of the research work. Section 4 the proposed trust management scheme followed by the detailed describtion of proposed clustering methodology with cluster formation and header section in Sect. 5. A misbehaviour verification mechanism to detect and revoke selfish or mischievous nodes is dscribed in Sect. 6. Section 7 presents the mobility adaptiveness of the proposed scheme followed by the system model in Sect. 8. The performance evaluation and simulations is illustrated in Sect. 9 and the concluding remarks appear in Sect. 10.

2 Related Works

Over the past several years there has been a significant amount of researches regarding clustering protocols and their issues in a PKI based MANET security system as done in [1–3]. This section considers the literature on MANET where the set of nodes need to form stable clusters to maintain scalability during secure communication and in case of link failures. It is difficult to provide a complete security to mobile networks due to its wireless connectivity, dynamic topology, and infrastructure-less features. Here, a brief outlook on the existing approaches for clustering; Trust and its importance and application of Voronoi

diagram in ad hoc networks that form the supporting framework to enhance the security in MANET are specified.

2.1 Clustering Methodologies

Clustering in MANET has become a well-known hierarchical structure to improve the efficiency in a dynamic network. The idea of clustering was first applied for routing in ad hoc networks, for example Cluster Based Routing Protocol (CBRP) proposed [11]. Most of the traditional clustering techniques for wireless networks were based on the metrics such as: dominating set, cost, energy efficiency, load balancing, battery power, weighting factor etc. These clustering metrics set out to be a virtual backbone for routing protocols. In latter year, security based clustering models were proposed as two-hop acknowledgment (2ACK) protocol presented by [12]. To handle the mobility of ad hoc nodes, clusters are formed with respect to their physical position and closeness to the other mobile nodes. Generally, these clustering methods are categorized into Cluster-Head (CH) and non-CH-based clustering methods. In both the methodologies, the groups are reconstructed whenever nodes change their position.

Based on the diameter of the cluster, the protocols in the CH-based clustering can be further classified into one-hop and multi-hop clustering (k-hop) [13]. The Cluster Members (CMs) are assumed to be at one-hop distance from its corresponding CH in the former methodology. Whereas, in k-hop clustering, due to the random movement of nodes, the hop distance between the header node and its members are restricted at k-hop. To the down side, the cluster maintenance is extremely complex and problematic in the k-hop clustering, under high mobility environment.

In a high-density network, these conventional clustering methods probably create a huge number of overlapping clusters that make increase in CH intensity. Consequently, these overlapping groups enclose the same set of cluster members with two or more CHs. Therefore, the mobility of a single node can possibly result in the reconstruction of several clusters. This mobility issue and its consequences in cluster construction and maintenance when applied for MANET is studied in [14]. Recently some clustering algorithms were presented to MANET with location and neighbour based as the primary clustering metrics [15]. A mobility prediction based clustering algorithm was introduced by [16] to solve the relative node movement issue in MANET. The advantages of clustering techniques extended their application in the area of ad hoc networks as presented in [17–20]. The most common problem in all these existing clustering methodologies is the nearest neighbour search in which the distance metric computation is significant.

2.2 Trust in Clustering

The security solution are introduced to provide the security services and to revoke attackers. Trust, in recent years, is considered as a critical aspect in the design of a secure distributed system. The nodes in the network setup a trust relationship among themselves by evaluating the trust value. Trust-based security schemes have been studied as an attacker detection technique in MANET [21–25].

In a mobility uncertain network like MANET, the degree of uncertainty (rate to which a node cannot predict accurately whether its neighbouring node is trustable or not) is considered to formulate trust as done by [26]. The DS theory has been used in multi agent systems, sensor networks and intrusion detection systems to predict uncertainty [27]. A Cluster based Trust-aware Routing Protocol (CBTRP) to protect forwarded packets from

malicious attacks was proposed in [28]. Trust-based security systems are also presented in various network architectures [29]. To overcome the pitfalls of traditional security systems with no expectations, the uncertainty reasoning is considered as probabilistic technique in MANET where mobility adaptive characterization is of greater importance.

2.3 Voronoi Diagrams in Clustering

In most of the clustering methodologies, the distance between the nodes is measured with Euclidean distance calculation during cluster formation as presented by [30]. However, this method works well only for specific distance function [31].

A Voronoi geometric structure has been introduced to handle this distance calculation problem [32]. The Voronoi structure provides information on the nearest neighbour search queries in uncertain plane [33]. To increase the network capacity, spatial reuse techniques have been widely applied for wireless services. This is achieved by dividing the network area into congruent clusters. VD is applied as the space decomposition method to evaluate the distribution probability of distance between nodes in MANET with an assumption that nodes are distributed independently in the polygonal cluster. These remarkable features of Voronoi diagrams increased its contribution in various wireless ad hoc applications as in [34, 35].

A hexagonal geometric distribution of nodes was introduced by [36]. This partitioning technique has shown to increase the network capacity and throughput of the network. It was proven the regular hexagons have flexibility to be partitioned into smaller hexagonal shapes and grouped together to form larger ones as stated by [37]. Withal, these clustering models increases the control overhead of cluster construction, maintenance and cluster head selection. The comparison of different clustering, trust and Voronoi diagrams mentioned in the literature work are given in the Table 1 given below.

3 Motivation of the Proposed Work

As discussed above, the merits and demerits of all the existing mechanisms are compared in order to choose and combine the best suited mechanisms to deploy PKI security framework in MANET. Owing to the absence of topology, providing a fully distributed self-organizing clustering framework security to the mobile nodes in MANET is difficult to achieve. An efficient solution for this hurdle should incorporate the promising features of all those mechanisms; clustering, trust and Voronoi diagram predominantly for managing the PKI framework, which is still unresolved. Such an optimal security solution is proposed in this paper for providing clustering security in MANET by overwhelming the following drawbacks in the existing mechanisms.

- The traditional clustering techniques face Nearest-Neighbor Search (NNS) problem for which no exact solution is determined yet.
- Numerous CHs and cluster gateways are needed to cover mobile nodes and inter-cluster connection respectively in a highly overlapping cluster structure, which increases the cluster construction complexities.
- Whenever the cluster membership changes, the overhead and computation complexities for cluster re-construction get increased in traditional clustering methodologies.

Table 1 Comparison of clustering, trust and voronoi mechanisms

References	Context in use	Performance metrics	Advantages	Disadvantages
Clustering methodologies				
Jiang et al. [11]	The clustering approach is done to minimize the traffic in route discovery. Route acquisition delay is reduced with local repair process	Clustering is evaluated with packet delivery ratio and routing overhead with respect to network size	Increased packet delivery ratio. Reduced route rediscovery traffic. Reduce route acquisition delay	Suggest solution to use only uni-directional links. Two level hierarchical model is scalable to an extend only. Increased overhead bytes per packet
Liu et al. [12]	Misbehaving links are detected by acknowledgments transmitted over consecutive nodes between the source and destination	Two-hop acknowledgment packets are used in the routing path to detect routing misbehaviour	Reduces the additional routing overhead	Acknowledgment is provided only for a fraction of packets with increased misbehaviour detection delay
Ni et al. [16]	A mobility-based clustering scheme is presented	The cluster head is selected based on Doppler shift that arise from the HELLO packets	Mobility aware clusters with high cluster stability	The cluster members move at relatively low speed with overlapping clusters
Chinara and Rath [13], Agarwal and Motwani [14], Anupama and Sathyanarayana [15], Sucasas. et al. [20]	A comprehensive survey of clustering schemes for MANETs based on their objectives, the cluster heads election criteria and mobility is presented			
Trust in clustering				
Zouridaki et al. [21]	Past experiences and current behaviour are combined to estimate trust using Bayesian approach	Trust is calculated as the probability factor	No single point failure	No precise trust measurements
Chen and Venkataramanan [22]	Describes how Dempster–Shafer Theory (DST) is applied to distributed intrusion detection in ad hoc networks	Degrees of belief about a hypothesis can be obtained from subjective probabilities and these beliefs can be combined together	DST has advantages over uncertainty in intrusion analysis to deal with prior probabilities for all events and the ability to combine beliefs from multiple sources	Doesn't consider DST in a trust related context The evidence collected from neighbours may be unreliable

Table 1 continued

References	Context in use	Performance metrics	Advantages	Disadvantages
Sun et al. [23]	Trust is calculated based on packet forwarding behaviour	Trust is calculated as entropy metrics with range [0, 1]	Can be applied to any wireless networks	Trust is instantaneously calculated based on individual nodes
Jhaveri and Patel [24]	A trust-model is integrated with an attack discovery technique	AODV routing protocol and node's historical behaviours are used for trust management	Earlier detection and elimination of adversaries	No trade-off between security levels and energy consumption
Safa et al. [28]	Organizes the network into disjoint clusters and elects cluster head with the most qualified and trustworthy nodes	Cluster-based trust-aware routing protocol	Ensures the trustworthiness of by replacing malicious cluster heads	Load balance clustering is a dynamic optimization problem
Voronoi diagrams in clustering				
Lee et al. [30]	Presents an optimized UK-means algorithm, which generalises the k-means algorithm to handle uncertainty	The uncertain location is described by a probability density function (pdf)	Tremendously reduces the execution time of UK-means algorithm to the traditional clustering algorithm	The proposed methodology works only for a specific form of distance function
Nichols and Michalowicz [31]	Distance statistics for mobile ad-hoc wireless network have focused on the three-dimensional spatial cases	Average inter-node distance. Average number of neighbouring network nodes and distance distribution	High network reliability quantified with distance distribution	Distribution is performed with Euclidean distance
Kao et al. [32]	Propose pruning techniques that are based on Voronoi diagrams to reduce the number of expected distance calculations	R-tree index to organize the uncertain nodes	Reduces the computation of expected distances between uncertain objects and cluster head	The complexity of the UK-means is not reduced by the proposed pruning techniques
Xie et al. [33]	Voronoi diagram is used for uncertain spatial data for evaluating nearest-neighbour queries	*Uncertain-Voronoi diagram* divides the data space into disjoint partitions	Support probabilistic nearest-neighbour queries execution	It is computationally infeasible to create and store UV partitions

Table 1 continued

References	Context in use	Performance metrics	Advantages	Disadvantages
Elwin et al. [34]	Finds the Voronoi neighbours directly from inter-object distances, before assigning coordinates	Distances between objects assign coordinates to each neighbour	Effectiveness in the presence of noise	Increased computational complexity
Fan et al. [36]	The probability density function of the distance between two nodes is derived using space decomposition method. The node degree is calculated with a simple path loss model	Probability distribution of the distance between nodes	Efficient node degree distribution and maximum flow capacity of the network	Limitation with multi hop networks
Tong et al. [37]	A probabilistic distance-based model is presented for Nodal Distance Distribution over a finite network	Considers network coverage and nodal spatial distribution	Extended to the networks with shape of one or multiple arbitrary polygons	Trust metrics are not considered as functions of the distances among interfering nodes

Hence, the objective of this paper is to propose an efficient Hybrid Trust based Mobility Aware Clustering (HTMAC) strategy is proposed as the distributed hierarchical solution for uncertain MANET system, in order to secure and to reduce the complexities in the cluster construction. To partition the uncertain nodes of MANET, the Voronoi based clustering is performed in hexagon structured polygon to reduce region overlapping drawbacks that occur in traditional clustering shapes. In this research work, the self-organized framework is developed, which quantifies node's behaviour in the form of trust. To encounter challenges in node cooperation and security, hybrid trust computation is calculated in the proposed HTMAC scheme, where cluster heads are selected with high trust degree. The originality of the proposed work combines different metrics for clustering with Voronoi structure and Bayesian-Evidence trust management to predict the distributed security solution.

Each node estimate its neighbour's trustability based on hybrid trust which combines the direct observations and indirect recommendations. The evidence or direct observations are computed with Bayesian theorem with Beta distribution function. To compute the indirect trust and to cope with the uncertainty, DS theorem is used to provide a statistical measurement of belief degree about a node from multiple neighbours. This hybrid trust computation is proven to adapt the dynamic mobility of MANET nodes.

4 Proposed Trust Management

The distributed trust framework to adapt the active topology and to secure MANET is described in this section.

4.1 Distributed Hybrid Trust Model

Based on a hybrid method (i.e., the aggregation of direct and indirect trust factors or components), the distributed trust is computed. The direct trust is constructed by direct observations on sensing the neighbouring nodes. Whereas, the indirect trust is made by recommendations from the one-hop neighbours. Unlike, in centralized trust calculation, each node computes its own trust value on its neighbour. The trust computation of trustor 'm' on trustee '$T_{m,n}$', by hybrid mechanism as shown in Fig. 1, is calculated as follows in (1)

$$T_{m,n} = (1-\mu)T^{D}_{m,n} + \mu T^{ID}_{m,n} \tag{1}$$

Herein, 'μ' is the trust component that ranges from $0 \leq \mu \leq 1$, '$T^{D}_{m,n}$' denotes the direct trust made by 'm' on 'n' from $0 \leq T^{D}_{m,n} \leq 1$, and '$T^{ID}_{m,n}$' is the indirect trust made by 'm' on 'n' from $0 \leq T^{ID}_{m,n} \leq 1$. The direct trust calculated from the direct observations of 'm' on 'n' at time 't' is given by the Eq. (2). With the change in the time 't_1', the trust may decay, represented by the decay component 'δ'.

$$T^{D}_{m,n} = \begin{cases} T^{D}_{m,n}(t); & if \quad hopcount == 1 \\ \delta T^{D}_{m,n}(t - t_1); & otherwise \end{cases} \tag{2}$$

The indirect trust evaluated by 'm' on 'n' with respect to the recommendation from one-hop neighbour of 'n' (node 'p'), at time 't' is given by the Eq. (3). The trust decays with 't_1' when 'm' receive incorrect recommendations from the recommender node 'p' located within an optimum trust length from 'm'.

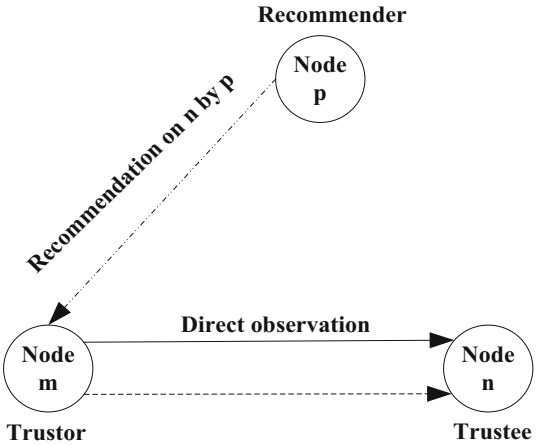


Fig. 1 Hybrid trust method

$$T_{m,n}^{ID} = \begin{cases} T_{p,n}; & |CR| > 0 \\ \delta T_{m,n}(t - t_1); & otherwise \end{cases} \tag{3}$$

Here, 'CR' is the set of correctly received recommendations from 'n's' one hop neighbour (i.e., 'p'). When 'CR > 0, m' employs those one-hop neighbo nodes to assess the trust indirectly. On the other hand, if 'CR = 0, m' uses its past trust value '$T_{m,n}(t - t_1)$', since it received no correct recommendations.

4.2 Distributed Trust System with Bayesian and Evidence Theorem

When a CH receives a misbehaviour alert message from any node, it verifies whether the message is attained from an acceptable node. The CH observes the suspected node and requests the 1-hop neighbours of the suspected node to share their independent observations about 'x'. The observations are considered as evidences that are in the form of number of observed misbehaviours to calculate the evidence trust factor '$\alpha^x(e)$'. The CH also computes the misbehaviour rate in terms of trust factor '$\alpha^x(d)$' by directly observing the node 'x'. The trust systems, usually combines the direct observations and the evidences obtained from the 1-hop neighbours to decide the trustworthiness of 'x'.

The existing trust systems let-down when the observing node itself is untrustworthy, which contributes no true evidences. Such systems might be impracticable especially to inform which observer node is untrustworthy. Hence, the proposed system employs Dempster–Shafer (DS) Evidence Theory (ET) developed by Dempster and revised later by Shafer, where the uncertainty is represented in the form of belief functions. The core idea of the DS theory is that an observer acquires a certain degree of belief on a proposition based on the subjective probability of a related proposition or hypothesis. DS theory aims to provide a convenient mathematical model to combine disparate information obtained from different sources.

4.2.1 Direct Trust Computation: Bayesian Theory

The proposed system assumes that the CH can lookout the key forwarded by the suspicious node and compare them with the original packets to identify the misbehaviour nature (deviation from expected trust behaviour) of the node 'x'. Therefore, the CH directly calculates the trust factor of its member nodes by Bayesian inference. In which, the unknown probabilities are inferred using observations. The measure of belief about a proposition or hypothesis is stated with well-known Bayes theorem as (4):

$$P(a|b) = \frac{P(b|a)P(a)}{P(b)} \tag{4}$$

where '$a|b$' is the measure of belief about the proposition 'a' with respect to the evidence 'b', '$P(a)$' denotes belief about 'a' in the absence of 'b'. The Baye's theorem can also be expressed in terms of probability distribution as in (5):

$$P(\Phi|data) = \frac{P(data|\Phi)P(\Phi)}{P(data)} \tag{5}$$

Herein, '$(\Phi|data)$' measures the posterior distribution for the parameter 'Φ', '$P(data|\Phi)$' is the sampling density function, '$P(\Phi)$' mentions prior distribution, and '$P(data)$' is the

marginal probability function of data. From the (5), the misbehaviour verification can be modified as:

$$P(\Phi, i|j) = \frac{f(j|\Phi, i)P(\Phi, i)}{\int_0^1 f(j|\Phi, i)P(\Phi, i)d\Phi} \tag{6}$$

In the Eq. (6), 'Φ' is the degree of belief that ranges from $0 \leq \Phi \leq 1$, 'j' denotes rate of correctly forwarded data by a node, 'i' is the rate of data received by the node, and 'f(j|Φ, i)' describes probability function that follows a binomial distribution given by (7):

$$f(j|\Phi, i) = \binom{i}{j}\Phi^j(1-\Phi)^{i-j} \tag{7}$$

Beta distribution is used to the Bayesian approach for describing the initial knowledge concerning probabilities of success. Therefore, the prior distribution 'P(Φ, i)' can be stated as in (8):

$$\text{ȝ}\,(\Phi;\ \alpha, \beta) = \frac{\Phi^{\alpha-1}(1-\Phi)^{\beta-1}}{\int_0^1 f(j|\Phi,i)P(\Phi,i)\,d\Phi} \tag{8}$$

where $\alpha, \beta > 0$, is the power function of 'i' and 'j'.

The mean and variance of the beta distribution function is given in (8) and (10):

$$E(\Phi|\alpha, \beta) = \frac{\alpha}{\alpha + \beta} \tag{9}$$

$$V(\Phi|\alpha, \beta) = \frac{\alpha\beta}{\alpha + \beta + 1} * \frac{1}{(\alpha + \beta)^2} \tag{10}$$

In the proposed scheme, the trust factor reflects the behaviour fading thereby giving more weights on the misbehaving rate in Bayesian network. The trust factor for misbehaviour verification is given as (11):

$$E(\Phi|\alpha, \beta) = \frac{\alpha}{\alpha + \alpha^x\beta} \tag{11}$$

On considering the transaction history in the Bayesian framework for misbehaviour calculation, the expectation of beta distribution can be written as (12):

$$E(\Phi|\alpha, \beta) = \frac{\alpha_t}{\alpha_t + \alpha_t^x\beta_t} \tag{12}$$

where $\alpha_t = \alpha_{t-1} + i_{t-1}$ and $\beta_t = \beta_{t-1} + j_{t-1}$, in which no observations are made at initial stage and so α_0, $\beta_0 = 0$. Based on the above deduction, the direct trust factor of the CH is computed on the node 'x' can be written as (13):

$$T_x^D(t) = (\alpha^x(d)) = E(\Phi|\alpha, \beta) \tag{13}$$

4.2.2 Indirect Trust Computation: Evidence Theory

This section describes the trust computation based on the indirect observations from the 1-hop neighbours of the suspicious node 'x'. As shown in the Fig. 2, the CH requests the

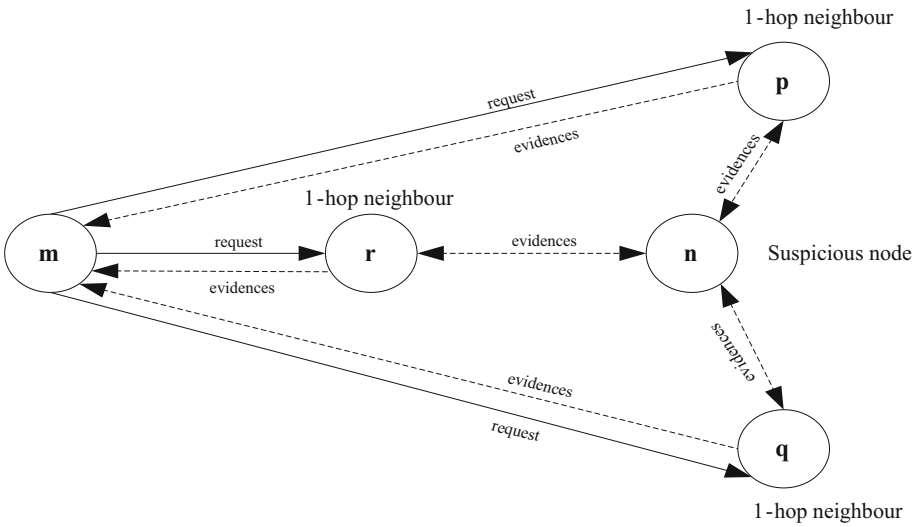


Fig. 2 Indirect trust computation

1-hop neighbours of 'x' to verify the misbehaviour rate from their independent observations about 'x'. The observations (also called evidences) obtained from the 1-hop neighbours help in judging the trustworthiness of 'x'. To perform this, the DS theory is used with uncertainty or ignorance. This theory is based on key element namely belief function which depends on the subjective probabilities that are combined to form indirect evidences.

In DS evidence system, the probabilities that are mutually exclusive and exhaustive are considered as a set and frame of discernment, denoted by 'Ω'. A power set represented by '2^{Ω}' includes all basic probabilities of the proposition called focal values 'A_k', which is a function of 'm' and satisfies the following two conditions:

1. Basic probability value of null set is zero (i.e., $m(\phi) = 0$).
2. Sum of elements in 2^{Ω} is 1 (i.e., $\sum_{A_k \subseteq \Omega} m(A_k) = 1$).

Moreover, the belief function can be defined as in (14):

$$B(x) = \sum_{A_k \subseteq x} m(A_k) \tag{14}$$

In the proposed scheme, two behaviour states are designed to the nodes namely, accept and revoke, demonstrated with DS theory. The frame of discernment consists of two possibilities concerning the behaviour level for any random node as Ω = (trust, distrust). This can further represented as: (a) an trust or allowable behaviour state, and (b) a distrust or mischievous state. The mischievous node behaviour is measured and isolated with the proposed verification mechanism that is integrated with the hybrid trust computation. On considering the Fig. 2, the 1-hop neighbours node A, B and C of 'x' shares their evidences as a subset of 'Ω'.

The power set '2^{Ω}' includes three possible forms of hypothesis and they are: hypothesis T = (trust), hypothesis D = (distrust), and hypothesis H = Ω. It represents node 'x' which is either in acceptable or mischievous state. The hypothesis H refers the Uncertainty

Degree (UD), which can be stated as the degree to which a node is unsuccessful to predict the behavior of its neighbouring node as either trustworthy or untrustworthy. The 1-hop neighbour provides evidences based on their direct observations by sharing its belief over 'Ω'. For example, if node 'A' believes 'x' behaves trustworthy, then $m_A(T)$is $\alpha^x(A)$ and therefore $m_A(D)$ is 0. The evidence from node A can be stated as in (15):

$$\begin{aligned} m_A(T) &= \alpha^x(A) \\ m_A(D) &= 0 \\ m_A(H) &= 1 - \alpha^x(A) \end{aligned} \tag{15}$$

Likewise, if node 'B' believes 'x' as misbehaved, its evidence favours revoke function as (16)

$$\begin{aligned} m_B(T) &= 0 \\ m_B(D) &= \alpha^x(B) \\ m_B(H) &= 1 - \alpha^x(B) \end{aligned} \tag{16}$$

4.2.3 DS Theory of Combining Evidences

The DS theory combines all the 1-hop neighbours evidences based on the condition that the evidences are independent. Suppose '$B_1(x)$' and '$B_1(x)$' are two independent observer's belief functions over same suspicious node, then the orthogonal sum of these functions is as in (17):

$$B(x) = B_1(x) + B_2(x) = \frac{\sum_{j,k,A_j \bigcap A_k = x} m_1\left(A_j\right) * m_2(A_k)}{\sum_{j,k,A_j \bigcap A_k \neq \Phi} m_1\left(A_j\right) * m_2(A_k)} \tag{17}$$

where '$A_j, A_k \subseteq \Omega$'. With reference to the Fig. 2, the belief of node 'A' and 'B' is calculated as in (18), (19) and (20):

$$m_A(T) \oplus m_B(T) = \frac{1}{I}[m_A(T)m_B(T) + m_A(T)m_B(H) + m_A(H)m_B(T)] \tag{18}$$

$$m_A(D) \oplus m_B(D) = \frac{1}{I}[m_A(D)m_B(D) + m_A(D)m_B(H) + m_A(H)m_B(D)] \tag{19}$$

$$m_A(H) \oplus m_B(H) = \frac{1}{I}[m_A(H)m_B(H)] \tag{20}$$

where,

$$\begin{aligned} I = {} & m_A(T)m_B(T) + m_A(T)m_B(H) + m_A(H)m_B(H) \\ & + m_A(H)m_B(T) + m_A(H)m_B(D) \\ & + m_A(D)m_B(D) + m_A(D)m_B(H) \end{aligned} \tag{21}$$

Assume, the acceptance rate of probability of node 'A' and 'B' is 0.8 and 0.7 respectively, and therefore,

$$B(T) = 0.94 \tag{22}$$

$$B(D) = 0 \tag{23}$$

$$B(H) = 0.6 \tag{24}$$

Thus, the acceptable behaviour value from the indirect observation with DS theory is 0.9. In general, the evidence trust factor obtained from the indirect observations can be computed as in the Eq. (25).

$$T_x^{ID}(t) = (\alpha^x(e)) = m_A(T) \oplus m_B(T) \oplus \ldots\ldots \oplus m_N(T) \tag{25}$$

where nodes A, $B \ldots N$ are one hop neighbours.

Eventually, the hybrid trust shall therefore be calculated as in (26):

$$T_{m,n} = (1 - \mu)(\alpha^x(d)) + \mu(\alpha^x(e)) \tag{26}$$

5 Proposed Clustering Model

This section provides a detailed description of the cluster construction and the header election process presented in the proposed HTMAC scheme.

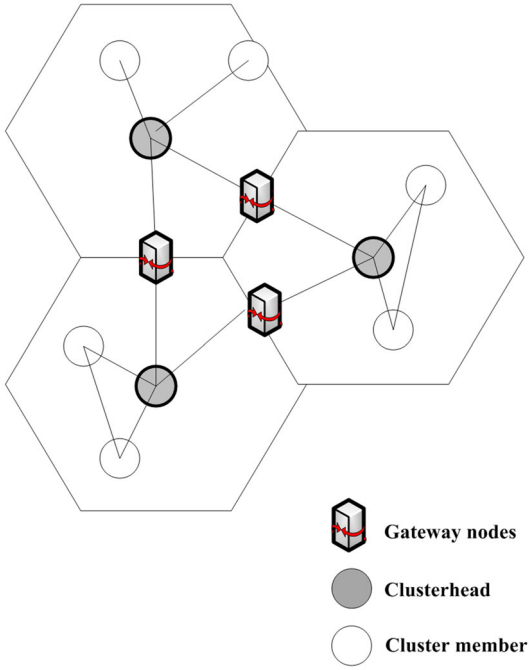
5.1 Cluster Construction

An efficient clustering scheme is designed with the ad hoc environment to form stable clusters for the underlying network operations. The Voronoi clusters are constructed with nodes $N_i = \{N_1, N_2 \ldots\ldots N_k\}$ with a distance function $d : D^m \times D^m \rightarrow D$ (m-dimensional space) giving the distance $d(i,j) \geq 0$ between any nodes $i, j \in D^m$. To adapt the dynamic mobility of MANET, the diameter of the cluster should be flexible and so herein, we use hexagonal shape non-overlapping clusters.

The hexagonal Voronoi technique divides a larger MANET into adjacent, non-overlapping clusters and can be subdivided into regular hexagonal regions. The nodes join to form Voronoi clusters and each cluster consists of Cluster Head (CH) and Cluster Members (CM), as shown in Fig. 3. In the proposed scheme, clusters have exactly one CH elected based on trust value, that is elected as given in following Sect. 5.2. The nodes in the boundary region and within the transmission range of any two CH are considered as gateway nodes, which handles cluster-to-cluster operations.

The CH monitors its neighbour nodes with their trustworthiness, within each cluster. We assume all the nodes communicate through bi-directional channels so that each node can forward as well as hear from its neighbouring nodes. The distance $D(i,j)$ between nodes in a cluster plays a significant role in determining the MANET performance. For computation, we assume the nodes of the network are independent and randomly distributed in the cluster region. The edges of the Voronoi hexagonal clusters are perpendicular to the line joining a cluster node with another in N_i. Considering the radius R_i, with respect to a node point $a \in D$ can be written as: $(a, c_x) - D(a, c_y) = R_x + R_y$; where$\{$ c_x, c_y $\}$: cluster representatives. If two nodes overlap, the distance $D(N_x, N_y) < R_x + R_y$ and distance $D(a, c_x) - D(a, c_y)$ become unreal, where the edges cannot be calculated and represents an empty cluster. The cluster construction is described in the Algorithm 1.

Fig. 3 Clustering model



Algorithm 1: Voronoi Hexagonal Clusters Input: Nodes $N_i = \{N_1, N_2 \ldots \ldots \ldots \ldots N_k\}$
Output: Clusters $C_i = \{C_1, C_2 \ldots \ldots \ldots \ldots C_k\}$

1. for each $N_n \in N_i$ do;
2. Consider an expected cluster region; $A_{R_i} \leftarrow D^m$.
3. for each $N_m \in N_i \bigwedge m \neq n$, do
4. Compute cluster edges; $E_n(m) \leftarrow$ edge ofN_n
5. Assign neighbours as $X_n(m) \leftarrow$ neighbouring regions of $E_n(m)$
6. To form non overlapping clusters, execute; $A_{R_i} \leftarrow A_{R_i} - X_n(m)$
7. end for
8. Verify the expected region lies within the region boundary (RB) of the network region. RB is the region with sides perpendicular to the principle axes of D^m of finite region. i.e., if $A_{R_i} \subseteq$ RB, dos
9. Assign the region as a Voronoi cluster
10. end if
11. end for.

5.2 Cluster Communication and Header Election

In the proposed scheme, each node calculates its own trust function (T_f) and trust rate (T_{rate}) for clustering and CH election, as given in (27) and (28).

$$T_f = w_1 * T^D_{m,n} + w_2 * T^{ID}_{m,n} + w_3 * M_b + w_3 * B_p \tag{27}$$

where $w_1 + w_2 + w_3 + w_4 = 1$, is the weight factor
M_b: mobility of a node
B_p: battery power of a node

T_{rate} is an important parameter that reflects the honesty of any node against network attacks (e.g. Sybil attack, false ID distribution). It is calculated as the ratio of positives experiences (p) received to the total experiences (η) send by a node, which is given as

$$T_{rate} = \frac{\mathrm{p}}{\eta} \tag{28}$$

Initially, each node sends a *HELLO* message to all the nodes within its transmission range as shown in Fig. 4.

The proposed scheme indicates two special fields, namely Node Location Identification (NLI) and Neighbour Trust Value (NTV) for mobility and security reinforcement. NLI represents the location, history of any node which moves frequently from one cluster to another, using the GPS information. Likewise, NTV indicates the trust a node keeps on its one-hop neighbours based on direct observations. The *HELLO* message, as shown in Fig. 4, includes: 4 byte field of sender's id (S_{id}) which represents the identity (e.g. IP address) of the sender node, 2 bit field of S that represent the current status of the sender as undefined, CH or Cluster member (CM) (S is set as 1 if sender node is header, 2 if the sender node is a CM and 0 if undefined), 2 byte field of T_f that represents the trust function of the sender node, 1 byte field $1 - HN$ which represents the number of one hop neighbours of the sender node, 1 byte *NLI* that represents the location details of the sender node 3 bytes for each one hop neighbouring node address along with the *NTV* of 1 byte, calculated by the sender node.

On receiving the *HELLO* message, each one-hop neighbor replies with a *REPLY* message. The *REPLY* message includes the receiver node's ID along with its trust function and neighbor trust value, as shown in Fig. 5.

In addition, all the nodes maintain a $h()$, as shown in Fig. 6, to record the frequently changing neighboring nodes and their trust values. The neighbor table will be utilized in clustering and routing processes. It includes the ID of the neighbor, status of the neighbor node (which is set to 1 if it is a CH or as 0 if it is a member), its trust component (x), the

Fig. 4 *HELLO* message

0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
Sender ID (S_{id}) | S
T_f | $1 - HN$ | NLI
Neighbor 1 address | NTV_1
Neighbor 2 address | NTV_2
• | •
• | •
• | •
Neighbor n address | NTV_n

Fig. 5 REPLY message

Receiver ID	T_f	NTV

ID	Status	x	$T_{m,n}{}^{D}$	$T_{m,n}{}^{ID}$	T_f	Timer

Fig. 6 Table of Neighbour (ToN)

direct and indirect trust of the node, its trust function and finally a timer to refresh the table at regular time intervals.

An efficient cluster header election algorithm that can adapt the random mobility changes is shown in Algorithm 2.

Algorithm 2: Header Election Initialise S: as a sender node that wants to become cluster head

CH: cluster head

T_{f_s}: trust function of S

1. S broadcasts *ELECT* message to all its one hop neighboring nodes with its location, mobility and battery power.
2. Each node that receives the *ELECT* calculates the trust function T_{f_s} and check whether T_{f_s} is greater than the trust threshold (T_t) set by the node, below which the node assumes a certain risk in communication.
3. If $T_{f_s} > T_t$, that node will sign a header certificate '*HEADER_CERTI*' and sends to S.
4. S waits for a voting time of T_{vote} to hear from all its neighbors.
5. After T_{vote}, S counts the votes it received by computing the T_{rate}.
6. If the T_{rate} is high (i.e., $0.5 \leq T_{rate} \leq 1$), S advertises itself as header and broadcasts a *HEADER* message to nodes which elected S.
7. An *ERROR* message is send to S by a node whoses ID is wrongly included.
8. After a time T_{CH}, neighbor nodes sends a signed trust certificate(TC) for S, in response to the *HEADER* message.
9. Thus S becomes cluster header (CH)and the elector nodes who all signed the certificate becomes the cluster member (CM).

6 Proposed Misbehaviour Verification

The proposed HTMAC scheme evaluates node behaviour in order to detect the misbehaviour due to mischievous and selfish nodes. To attain this goal, the proposed hybrid trust management is accompanied with misbehaviour detection mechanism. The mischievous and selfish behaviour of nodes are distinguished based on its intention over the ad hoc networks. The mischievous state can be categorized as any action intended to harm the MANET system. Such node behaviour shall be characterized by packet dropping, altering control packets, IP spoofing, MAC spoofing or denial of service. Some of the attacks that affect the network adversely on account of the mischievous node are generally classified as: Black hole attack, Wormhole attack, Denial of Service (DoS) attacks, Jamming attack or Rushing attack. Whereas, a selfish node utilizes its battery power, energy, memory space and other resources selfishly for its own benefits. These selfish nodes use all the network services and decline to co-operate with other members in a cluster. Such node behaviour is featured by denial to share and forward the HELLO message, routing and control packets, and delaying the routing packets, for example: Sleep deprivation, DoS attack etc. Any such

misbehaviour either with selfish or mischievous node weakens the trust level by providing false recommendations about a trustworthy neighbouring node.

We consider the CH in each cluster as the judging node for misbehaviour verification. The judging node maintains a Rating Table (R_T) which records the direct rating of *CH* about the cluster members within a cluster. This direct rating is the summation of all the hybrid trust obtained from the Table of Neighbour(ToN) of the cluster members. Once a CH collects sufficient entries in the 'ToN', it can then proceed with the misbehaviour detection test as given in below to specify the trust values for all the MANET nodes. To adapt the detection scheme, the rating table entries are set as binary value; '0' or '1' for computation simplicity. That is, a node with a trust rating '1' would be indicated as selfish or mischievous node. The proposed HTMAC scheme detects and revokes the misbehaving nodes from the network to minimize their challenges.

For the verification purpose we classified the misbehaviour action into two cases: In the first case; the misbehaving nodes deny packets they received and generate a false trust to deliver to other misbehaving nodes via trustworthy nodes. Such illegitimate performance is considered as selfish node misbehaviour, which can degrade the MANET performance in terms of packet delivery ratio and data availability. In the second case; we assume the mischievous nodes simultaneously execute black hole attack and wormhole attack on legitimate packet, Denial of Service (DoS) attacks, Jamming attack on reliable communication, and Rushing attack on trustable users, in order to corrupt the underlying proposed trust management scheme.

The trust rating of node x on neighbouring node y; $T_{x,y}$, is defined by Beta distribution functions (α', β') which is initially set as (1, 1). The distribution function is updated whenever the ToN is revised by the corresponding nodes. Let the observed misbehaviour be m'; with $m' \in \{0, 1\}$, and the distribution function as (29) and (30):

$$\alpha' = \omega\alpha' + m' \tag{29}$$

$$\beta' = \omega\beta' + (1 - m') \tag{30}$$

where 'ω': fading factor obtained from the past experiences. The value of ω decays exponentially, so that the suspected and revoked node cannot restore by itself. This makes the proposed HTMAC scheme more advantageous with updated trust values.

To identify and avoid false trust entries due to the selfish or mischevious attacks, we assume node x receives recommendation from node i (i.e., $R_{i,y}$). Therefore, $T_{x,y}$ shall be modified as in (31):

$$T_{x,y} = T_{x,y} + \rho R_{i,y} \tag{31}$$

where ρ is a positive constant. The update is performed for all y node members. On considering the recommendation, $R_{i,y} = (\alpha_R, \beta_R)$ in the Bayesian framework for misbehaviour calculation, the expectation of beta distribution can be written as in (32). We consider the expectation computation as the verification test for false trust entries as:

$$E(Beta|\alpha_R, \beta_R) - E(Beta|\alpha, \beta) \geq \tau \tag{32}$$

where τ is the rating threshold. If the rating obtained by solving (32) is positive, $R_{i,y}$ is considered mutually exclusive and so eliminated. Else, $R_{i,y}$ is considered for rating calculations.

7 Mobility Adaptive Clusters

The proposed clustering scheme is designed to achieve a stable cluster organization with minimum communication overhead and complexities, in the presence of dynamic node mobility. This is established by two processes namely *NodeRegistration* and *NodeResign*, as described below.

7.1 Node Registration

When a node attempts to join a cluster, it should be registered with the other nodes, specially in an unstable topological ad hoc network. The registration procedure is described in Algorithm 3.

Algorithm 3: Node Registration

1. As shown in Fig. 7, *CH* broadcasts an *CLUSTER_ACTIVE* beacon to update the status of the existing cluster members as well as to make feasible the cluster for new nodes '*n*' to join.
2. At regular intervals, the nodes that sense the *CLUSTER_ACTIVE* beacon sends a *ACTIVE* message which includes S_{id}, T_f and *NLI*.
3. The *CH* verifies each *ACTIVE* message to validate the trust and location of the *CM*.
4. When a new node attempts to join the cluster, it sends *REG_CLUSTER* message to the *CH*, which includes S_{id}, T_f and *NLI* of the node *n*.
5. After verifying the T_f and location history, if the *CH* finds the node *n* as trustable, it sends a *TEMP_JOIN* message to temporarily join the cluster.
6. *CH* broadcasts a *VOTE* message along with the status of newly joined node to all its members, for calculating the *NTV* for the node *n*.
7. After a review period, *CH* calculates the T_{rate} of the node *n*, with the experiences got from the voters.
8. If the T_{rate} is higher than the T_t set by the *CH*, it sends a *TC* to the node *n* to confirm the temporary registration. The status of is node *n* is broadcast to all the members by the *CH*.

7.2 Node Resignation

The nodes get deactivated from a cluster due to certain reasons; connection failure, cluster disconnection or self departure. If a node voluntarily departs from the cluster, it announces its resignation by broadcasting a *RESIGN* message to all the nodes before leaving the cluster. At periodic interval, each node in a cluster has to send a *ACTIVE* message in response to the *CLUSTER_ACTIVE* message broadcast by the *CH*. When the *CH* doesn't receive *ACTIVE* message from any node for certain waiting time (say t_w), the *CH* broadcasts a *SENSE* message to all its members. For example, if the silent node is node *m*, *CH* sends $SENSE_m$ to the members. The cluster members who senses the presence of the deactivated node either due to connection failure or cluster disconnection, sends a *DETECT* message to the *CH*. The *CH* verifies the *DETECT* message and tries to establish a connection with the silent node ans asks for its *ACTIVE* message. The *CH* ratify the node *m* if the node replies or instead, *CH* assumes node *m* as damaged and broadcast a $RESIGN_m$ message to all the members.

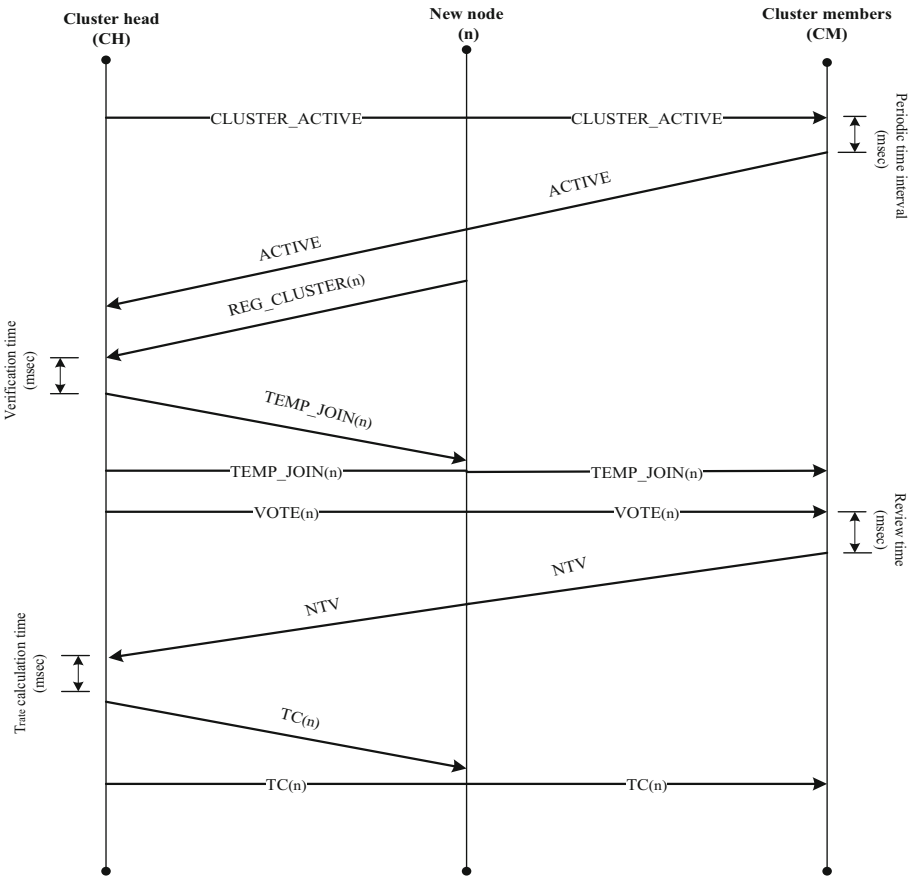


Fig. 7 Node registration in a cluster

8 System Model

This section describes the system model which includes the cluster model, security model and the attack model presented in the proposed HTMAC scheme in order to deploy PKI based framework in MANET.

8.1 Cluster Model

In the proposed clustering method, we divide the entire operational region into different equi-sized hexagonal clusters of area $A = \pi r^2$, where r is the radius of the cluster region. The number of hexagonal clusters in a region is given by (33)

$$N_c = 3d^2 + 3d + 1 \tag{33}$$

where d is the degree of rings. The N_c value is obtained from the mathematical induction in the network coverage model used. For example, in Fig. 8 the total number of clusters 'N_c' is 7 with degree of rings surrounded 'd' = 1.

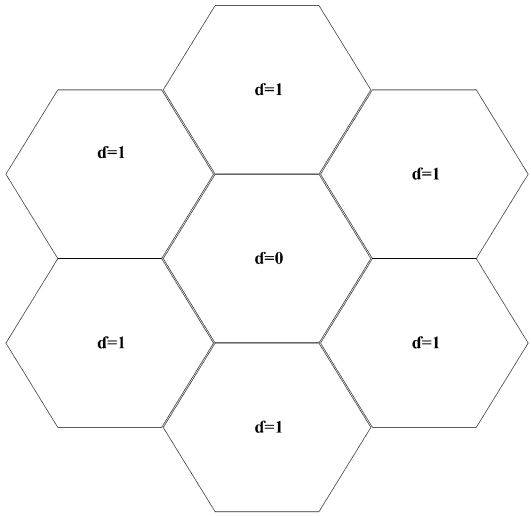


Fig. 8 Network region with N_c is 7 and $d = 1$

We assume the random nodes are distributed in a homogeneous passion fashion of density d_p, over the hexagonally clustered network region of area A. The average number of nodes (N) in any cluster depends on the area of the cluster and the density of distribution, which is given as

$$N = \pi r^2 d_p \tag{34}$$

Assume the rate of node joining the cluster be λ_j and the rate of node resigning from the cluster be λ_r. Therefore, the probability that a node is registered in a cluster can be $\lambda_j/(\lambda_j + \lambda_r)$ and the probability that it is resigned from a cluster is $\lambda_r/(\lambda_j + \lambda_r)$. In a hexagonal cluster region, the average number of active nodes is given as $N\lambda_j/(\lambda_j + \lambda_r)$. Moreover, the rate of all nodes registered in any cluster be R_j and the rate of all nodes resigned from the cluster be R_r, given in (35) and (36).

$$R_j = \lambda_j * N * \lambda_r / \left(\lambda_j + \lambda_r\right) \tag{35}$$

$$R_r = \lambda_r * N * \lambda_j / \left(\lambda_j + \lambda_r\right) \tag{36}$$

In the mobility adaptive cluster, the nodes can move dynamically within a cluster and across the boundary region. Let μ be the rate of mobility of a node when there is one cluster and μ_m be the mobility rate of the node for N_c clusters. Then, the mobility rate across the boundary is given by

$$\mu_m = (2d + 1)\mu * K \tag{37}$$

where factor K represents the intra-cluster mobility of the nodes for whom the mobility across the boundary is not applicable.

8.2 Security Model

The proposed clustering model provides assurance for security using hard security and soft security approaches using trust, as mentioned below.

- Authentication: When a node joins a cluster, the node's identity is authenticated based on the trust function calculated by hybrid trust mechanism. *Source authentication* is ensured during the verification process and *TC* is signed to authenticate the source node. *Location authentication* is performed by authenticating the *NLI*, especially in a mobile network like MANET.
- Integrity: To preserve the integrity, a node calculates its T_{rate} with the positive responses it obtained during cluster construction.
- Access control: The unauthorized use of resources is prevented using trust within each cluster. The services and resources allocated to the network are accessed by trustable node alone in the proposed cluster model.
- Communication risk: The proposed system indicates the presence of untrustworthy nodes that dissimate false communication. The *CH* validates each of its member's trust with T_t, below which it presumes certain communication risks and revoke those dishonest nodes from further cluster applications.
- Cluster availability: An *CLUSTER_ACTIVE* beacon at regular intervals make the cluster to work promptly so that no service denial for trustworthy nodes is assured.

8.3 Attack Model

We consider certain attacks in MANET as follows.

- *Dropping attack* interrupts the service availability of the nodes. The attackers deactivate nodes from their cluster by making a connection failure or cluster disconnection. The *SENSE* beacon send by the *CH* during node missing, re-establishes the connection with the deactivated node, after verification processes.
- *Fake recommendation attack* falsely sends recommendations to include an untrustworthy node in the cluster functionalities. The hybrid trust calculation we used, measures the direct trust from direct observations, in addition to the indirect trust obtained in the form of recommendations. This direct trust value gives higher importance for analyzing the trustworthiness of any node, which degrades fake recommendations.
- *Sybil attack* can break down the security, when a node in the network claims multiple identities. The integrity check of the node gets rid of such attackers, where the honesty of that node is proved. Also the *NLI* records the location history of each node, which aids the *CH* to detect the attacker node with multiple identities and same location particulars.
- *Impersonation attack* can be an identity spoofing, node cloning, reply or an unauthorized access. However, the attackers fail to pass the source and location authentication as well as integrity check.

With the misbehaviour verification mechanism in the proposed HTMAC scheme, the attacks that generates due to the selfish node and the mischievous node are detected and isolated from the MANET. This includes: Black hole attack, Wormhole attack, Denial of Service (DoS) attacks, Jamming attack, Rushing attack and Sleep deprivation

9 Simulation Results

The performance of the HTMAC scheme was tested through a series of simulation experiments on the QualNet simulator with IDE: Visual studio 2013, programming language: VC ++ and SDK: NSC_XE-NETSIMCAP (Network Simulation and Capture). For comparison, we also simulated three other clustering algorithms, namely the CBRP [11], 2ACK [12] and CBTRP [28]. A MANET environment is configured with many mobile devices (mobile phones, laptops, etc.) which move randomly to communicate among their neighbours in the network of transmission range 250. The probability of selecting new node as CH is set to 0.3. The nodes follow a Random Way Point (RWP) approach presented by the authors [38, 39], where the speed and the direction of each nodes are chosen randomly and independently.

When simulation starts, each node chooses one location randomly as the destination within terrain of 1000 m the simulation field. The nodes then moves with constant velocity chosen uniformly and randomly in a range $[0, V_m]$; where 'V_m' is the maximum range of velocity that a node travels. When the node reaches its destination, it halts for a time period, referred as halt time 'T_{halt}'. If $T_{halt} = 0$, a continuous mobility is experienced. However, when the 'T_{halt}' expires, the nodes again move randomly in the simulation field. The performance of the proposed THCM is evaluated by varying the two parameters 'V_m' and 'T_{halt}' for topology alterations (i.e., if 'V_m' is less and 'T_{halt}' is high, a relatively stable topology is achieved, while a highly dynamic topology is obtained if 'V_m' is high and 'T_{halt}' is less). Each data point in the simulation was run 10 times to compute the average value.

9.1 Mobility Adaptiveness

The two main properties of cluster that reflect the efficient adaptiveness of any clustering algorithm in MANET namely size of a cluster and the probability of node in a cluster subjected to node mobility are considered in this section. Figures 9 and 10 depict the mean cluster size and the node probability in a cluster with respect to the node velocity respectively.

9.1.1 Mean Cluster Size with Mobility

In MANET, with the increase in the node density, the size of cluster increases. The performance of any clustering scheme may not be worthy if the cluster size is larger. This is because the load to manage the traffic by the cluster head within each cluster gets accumulated. Therefore, the size of the clusters should be optimized in order to avoid high cluster maintenance overhead and to achieve desirable clustering scalability.

The Fig. 9 shows the influence of mobility on mean cluster size for the existing and the proposed clustering schemes. The results show how each clustering methodologies adapt the dynamic mobility of mobile nodes. When the node velocity is increased form 5 to 25 m/s, the size of the cluster diminishes from 25 to 7 numbers of nodes to aid from to benefit from more suitable routing. Whereas, the existing schemes shows a higher cluster size of 41–20 for 2ACK, 36–14 for CBRP and 35–14 for CBTRP, in the presence of different node speed of MANET nodes. It is precise that all the schemes maintain larger cluster size for lower mobility and reduced cluster size over greater mobility. The proposed

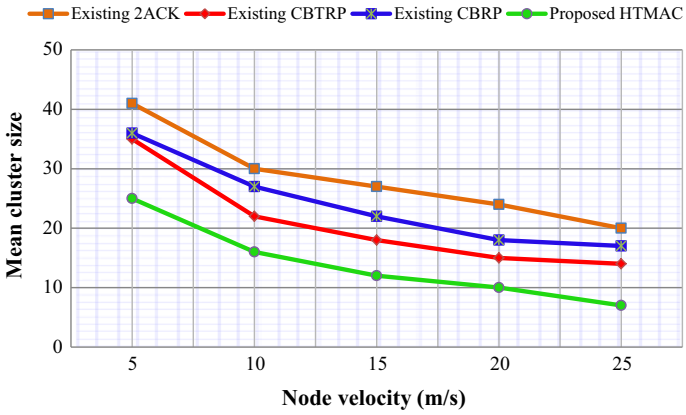


Fig. 9 Mean cluster size with node velocity

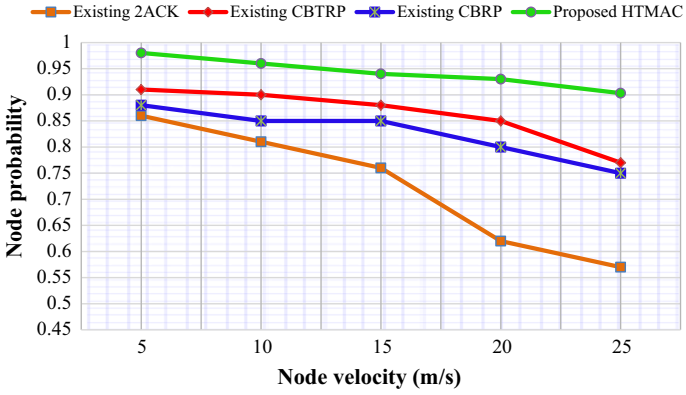


Fig. 10 Node probability with mobility

HTMAC clustering scheme, therefore, forms an appropriate number of clusters of optimal size with efficient mobility adaptive property.

9.1.2 Probability of Node with Mobility

The Fig. 10 illustrates the probability of node in a cluster with respect to the effect of mobility. It is clearly shown from the simulation result that the nodes remain higher probability in any cluster at larger mobility. The probability that nodes belong to any cluster greatly depends on the system parameters. Specifically, in the proposed HTMAC scheme the nodes remain clustered with probability greater than or equal to 0.9 at higher node mobility of 25 m/s. On the other hand, the existing methods such as 2ACK, CBRP and CBTRP show a lower probability range of 0.57, 0.75 and 0.77 respectively, on comparing with the HTMAC scheme. This higher probability of HTMAC scheme is achieved because of the partitioning technique that is carried out with Voronoi geometric structure in the proposed method. Thus result clearly shows the desirable property of

HTMAC that the mobile nodes remains clustered with high probability even at high node velocity.

9.2 Effectiveness in Stability

The stability of the proposed HTMAC scheme is measured in terms of the node residence time and the cluster age.

9.2.1 Residence Time with Mobility

Figure 11 represents the time each node survive in a cluster, referred as the residence time (R_T), in regard to the node velocity. The node residence time varies with the probability of node in the cluster. The clustering effort shall be creditable with longer residence time and higher probability of node being clustered. The residence time for the existing schemes is lesser than the proposed HTMAC scheme. The R_T value of the 2ACK methodology falls to 2 s from 17 s at greater node speed of 25 m/s. While considering the other two existing schemes; CBTRP and CBRP, the R_T value varies from 20 to 8 s and 25 to 10 s respectively, when the node mobility is increased from 5 to 25 m/s. Compared to these existing schemes, the proposed HTMAC shows an improved value of R_T from 27 to 15 s for the node velocity 5–25 m/s. This betterment in the residence time of HTMAC is due to the least probability that a node is clustered at that rate. By increasing the pause time further, the R_T value of the proposed scheme shall be increased significantly.

9.2.2 Cluster Duration with Mobility

Figure 12 demonstrates the cluster age of the proposed HTMAC scheme against different existing schemes. The cluster age is measured as the amount of time a cluster is active at each instant of time. Thus it represents the lifetime of each clusters in MANET. For a stable cluster topology, the cluster duration should be relatively longer. As shown in the Fig. 12, the cluster age in general, decreases with the increase in the node velocity. Compared to the other three existing schemes, the proposed HTMAC scheme has longer cluster age even at higher rate of node mobility. The cluster continues for time duration of

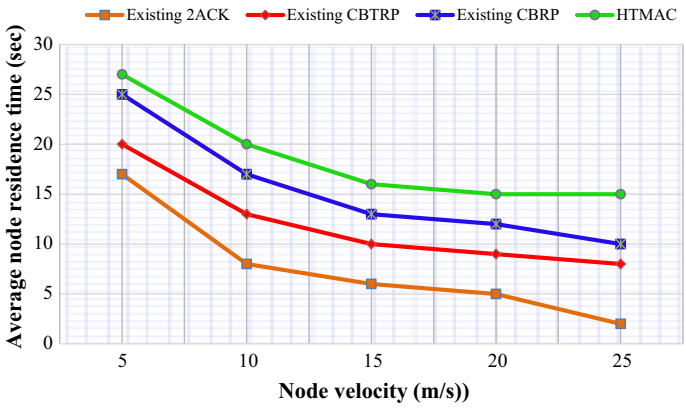


Fig. 11 Average node residence time with node velocity

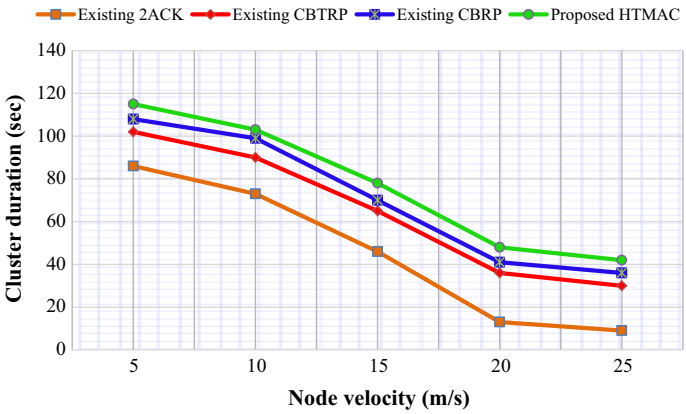


Fig. 12 Cluster duration with node velocity

maximum 42 s at higher node mobility to form stable clusters in the proposed HTMAC scheme. Whereas, for the existing schemes the cluster remains only for the maximum duration of 9 s in 2ACK, 30 s in CBTRP and 36 s in CBRP. This advantage in the cluster duration for the proposed scheme is due to the lower rate of link failures and link activations, which is around a maximum of 3 per sec at higher node velocity. This is because of the mobility adaptiveness of the nodes in the cluster construction which makes the members in the cluster stay connected to the CH for a longer time. Whereas, the clusters in the existing clustering schemes are less stable against the node speed. This shows the inability of those methodologies to adapt the mobility characteristics of the nodes in cluster formation.

This also shows the advantages of the HTMAC scheme in the reaffiliation rate. The reaffiliation rate represents the rate of change of CH and its affiliation per unit time with respect to the mobility speed of the node. The simulation result in Fig. 12 also reveals that the existing schemes have relatively higher reaffiliation rate than the HTMAC scheme. This is due to the reduced CH duration at increasing node speed in the existing schemes. On the other hand, in HTMAC scheme each node selects the node which remains its CH for a longer time, and so it reduces the probability of reaffiliation.

9.3 Efficiency of HTMAC Algorithm

The efficiency of the proposed HTMAC algorithm is measured in terms of performance parameters such as cluster overhead, message rate, and security as depicted in the Figs. 13, 14 and 15.

9.3.1 Cluster Overhead with Misbehaving Nodes

The cluster overhead includes the overhead in cluster organization and maintenance for the ad hoc network. The efficiency of any clustering algorithm depends on its ability to provide stable clusters. Figure 13 shows that the overhead caused by the proposed HTMAC scheme is minimum compared to the existing schemes.

The cluster overhead rises gradually from 0.056% with 5% misbehaving nodes to 0.072%, when 25% of the nodes are misbehaving. Whereas, the existing schemes CBRP

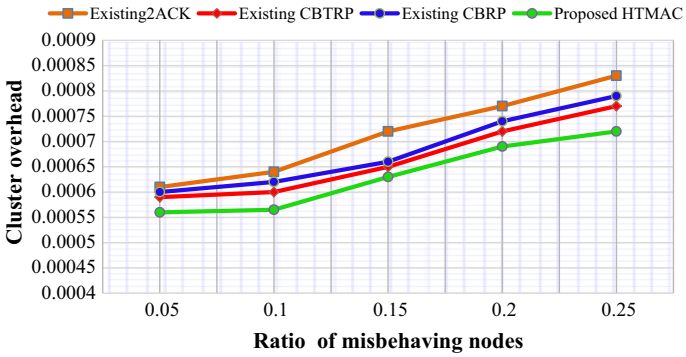


Fig. 13 Cluster overhead with misbehaving nodes

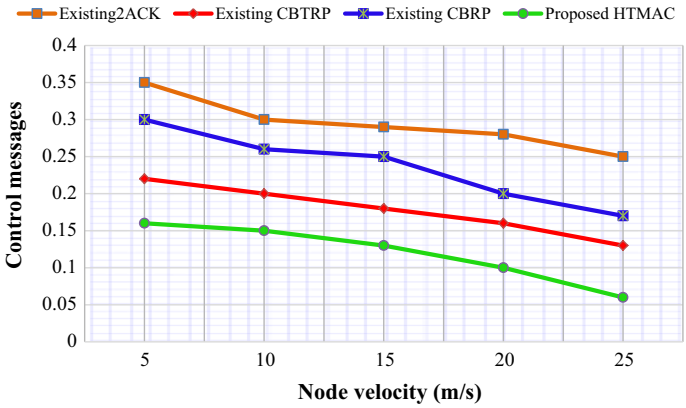


Fig. 14 Control message overhead

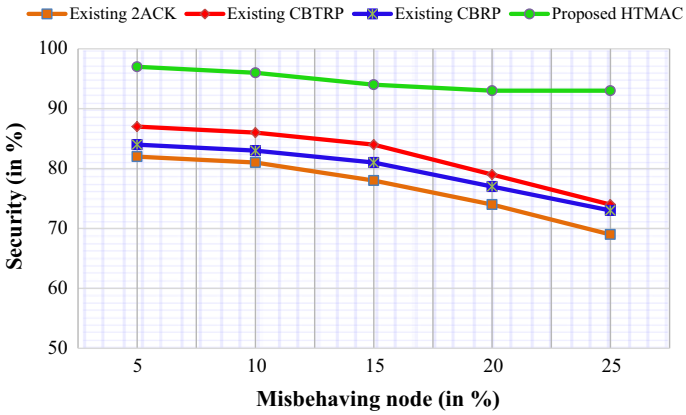


Fig. 15 Security rate

and CBTRP shows more or less an equal overhead percentage, which is around 0.07% for higher percentage of misbehaving nodes. While considering the 2ACK scheme, the 0.08% when the ratio of misbehaving nodes is 25%. This rise in the overhead is due to the flat architecture of the existing schemes which floods the cluster registration and header election packets throughout the network.

9.3.2 Rate of Control Message with Mobility

In the Fig. 14, the efficiency of the clustering techniques measured in terms of the rate of control messages per node are plotted. This measurement includes the number of cluster updates that are processed whenever cluster membership changes. It is noted that the proposed HTMAC scheme protects the mobile nodes from the frequent topology changes with respect to the increased mobility. Initially, the message rate increases with the change in topology in the proposed scheme. It is clearly shown in the Fig. 14 that the mobility adaptive clustering property of the HTMAC scheme diminishes the message rate by decreasing the size of clusters as node velocity increases beyond 10 m/s. For higher mobility, the rate of message attains an optimal value of 0.06, which is very much lower than the existing schemes.

9.3.3 Security Level with Mobility

The Fig. 15 shows the rate of security, which is considered as another significant parameter for measuring the algorithm's efficiency. The Hackman tool along with the Qualnet simulator verifies different attackers at regular time interval. The BCCC (Block Cipher Cryptography Class) interface is used to link with Hackman tool using Visual Studio IDE enabled with Hackman SDK. The Hackman tool the simulator continuously tries to break the data packets and counts the number of packets that are hacked successfully for total number of packets and then calculates the security level in %. The proposed HTMAC scheme demonstrates a higher rate of security to different attacks (as mentioned in Sect. 8.3) compared to other three existing schemes. A maximum of 97% security is achieved for minimum node mobility. The security level drops slightly to 93% when the velocity of node reaches the maximum of 25 m/s. On the other hand, the existing schemes such as 2ACK, CBRP and CBTRP possess a lower security level of 69, 73 and 74% respectively, for higher percentage of misbehaving nodes.

10 Conclusion

In this paper we have addressed hybrid trust based mobility aware clustering scheme(HTMAC) for mobile ad-hoc networks. In contrast to the existing techniques, we have proposed HTMAC to efficiently partition the network into non-overlapping clusters of trustable nodes. Our approach enables each node to establish trustability with other interacting nodes, in each hexagonal cluster, with minimal complexities in header selection and maintenance. In addition, the HTMAC scheme takes the advantage of trust mechanism to detect and revoke the selfish and mischievous nodes from the network in a short period of time. Simulation results shows that our scheme achieves *beneficial over (a) mobility adaptiveness (b) cluster stability with reduced overhead of clustering and cluster*

maintenance (c) control message rate and security. Therefore, our scheme, HTMAC, can be adequately adopted for infrastructural less and dynamic wireless ad-hoc networks.

Acknowledgement This research is supported by All India Council for Technical Education (AICTE), Government of India.

References

1. Yu, J. Y., & Chong, P. H. J. (2005). A survey of clustering schemes for mobile ad hoc networks. *IEEE Communications Surveys and Tutorials, 7,* 32–48.
2. Agarwal, R., & Mahesh Motwani, D. (2009) Survey of clustering algorithms for MANET. *International Journal on Computer Science and Engineering*, *1*(2), 98–104.
3. Bentaleb, A., Boubetra, A., & Harous, S. (2013). Survey of clustering schemes in mobile ad hoc networks. *Communications and Network*, *5*, 8–14.
4. Cho, J. H., Chan, K. S., Chen, I. R. (2013). Composite trust-based public key management in mobile ad hoc networks. In *ACM 28th symposium on applied computing, Coimbra, Portugal*.
5. Cho, J. H., & Kevin Chan, I.-R. C. (2013). A composite trust-based public key management in mobile ad-hoc networks. In *ACM 28th symposium on applied computing, trust, reputation, evidence and other collaboration know-how (TRECK)*.
6. Ferdous, R., Muthukkumarasamy, V., & Sithirasenan, E. (2011). Trust-based cluster head selection algorithm for mobile ad hoc networks. In *Proceedings of international joint conference on IEEE TrustCom*.
7. Wei, Z., Tang, H., Richard Yu, F., Wang, M., & Mason, P. (2014). Security enhancements for mobile ad hoc networks with trust management using uncertain reasoning. *IEEE Transaction on Vehicular Technology*, *63*(9), 4647–4658.
8. Chau, M., Cheng, R. B., Kao, B., & Ng, J. (2006). Uncertain data mining: An example in clustering location data. In *Proceedings of PAKDD* (pp. 199–204).
9. Cheng, R., Xie, X., Yiu. M. L., Chen, J., & Sun, L. (2010). Uv-diagram: A Voronoi diagram for uncertain data. In *Proceedings of the 26th IEEE International Conference on Data Engineering* (pp. 796–807).
10. Zhuang, Y., Gulliver, T. A., & Coady, Y. (2013). On planar tessellations and interference estimation in wireless ad-hoc networks. *IEEE Wireless Communication Letters*, *2*(3), 331–334.
11. Jiang, M., Li, J., & Tay, Y. C. (1999). Cluster based routing protocol(cbrp). In *Internet draft, MANET working group*.
12. Liu, K., Deng, J., Varshney, P. K., & Balakrishnan, K. (2007). An acknowledgment-based approach for the detection of routing misbehavior in MANETs. *IEEE Transactions on Mobile Computing*, *6*(5), 536–550.
13. Chinara, S., & Rath, S. (2009). A survey on one-hop clustering algorithms in mobile ad hoc networks. *Journal of Network and Systems Management*, *17*(1–2), 183–207.
14. Agarwal, R., & Motwani, M. (2009). Survey of clustering algorithms for MANET. *International Journal on Computer Science and Engineering*, *1*(2), 98–104.
15. Anupama, M., & Sathyanarayana, B. (2011). Survey of cluster based routing protocols in mobile ad hoc networks. *International Journal of Computer Theory and Engineering*, *3*, 806–815.
16. Ni, M., Zhong, Z., & Zhao, D. (2011). MPBC: A mobility prediction-based clustering scheme for ad hoc networks. *IEEE TVT*, *60*(9), 4549–4559.
17. Afsar, M., Tayarani-N, M.-H., & Aziz, M. (2015). An adaptive competition-based clustering approach for wireless sensor networks. *Telecommunication Systems*, *61*(1), 1–24.
18. Khakpour, S., Pazzi, R. W., & El-Khatib, K. (2017). Using clustering for target tracking in vehicular ad hoc networks. *Vehicular Communications, 9,* 83–96.
19. Cooper, C., Franklin, D., Ros, M., Safaei, F., & Abolhasan, M. (2017). A comparative survey of VANET clustering techniques. *IEEE Communications Surveys & Tutorials, 19*(1), 657–681.
20. Sucasas, V., Radwana, A., Marques, H., Rodriguez, J., Vahid, S., & Tafazolli, R. (2016). A survey on clustering techniques for cooperative wireless networks Victor. *Ad Hoc Networks*, *47*, 53–81.
21. Zouridaki, C., Mark, B. L., Hejmo, M., & Thomas, R. K. (2005). A quantitative trust establishment framework for reliable data packet delivery in MANETs. In *Proceedings of the 3rd ACM workshop SASN* (pp. 1–10).
22. Chen, T. M., & Venkataramanan, V. (2005). Dempster–Shafer theory for intrusion detection in ad hoc networks. *IEEE Internet Computing, 9*(6), 35–41.

23. Sun, Y., Yu, W., Han, Z., & Liu, K. J. R. (2006). Information theoretic framework of trust modeling and evaluation for ad hoc networks. *IEEE Journal on Selected Areas in Communications*, *24*(2), 305–317.
24. Jhaveri, R. H., & Patel, N. M. (2016). "Attack-pattern discovery based enhanced trust model for secure routing in mobile ad-hoc networks. *International Journal of Communication Systems*. https://doi.org/10.1002/dac.3148.
25. Movahedi, Z., Hosseini, Z., Bayan, F., & Pujolle, G. (2016). Trust-distortion resistant trust management frameworks on mobile ad hoc networks: A survey. *IEEE Communications Surveys & Tutorials, 18*(2), 1287–1309.
26. Li, F., & Wu, J. (2007). Mobility reduces uncertainty in MANETs. In *IEEE international conference on computer communications, INFOCOM'07* (pp. 1946–1954).
27. Raya, M., Papadimitratos, P., Gligor, V. D., Hubaux, J.-P. (2008). On datacentric trust establishment in ephemeral ad hoc networks. In *Proceedings of the IEEE INFOCOM* (pp. 1–11).
28. Safa, H., Artail, H., & Tabet, D. (2010). A cluster-based trust-aware routing protocol for mobile ad hoc networks. *Wireless Networks, 16*(4), 969–984.
29. Wang, Y., Chen, I. R., Cho, J. H., Swami, A., Lu, Y. C., Lu, C. T., & Tsai, J. J. P. (2016). CATrust: Context-aware trust management for service-oriented ad hoc networks. *IEEE Transactions on Service Computing*. https://doi.org/10.1109/TSC.2016.2587259.
30. Lee, S. D., Kao, B., & Cheng, R. (2007) Reducing UK-means to K-means. In *The 1st workshop on data mining of uncertain data (DUNE), in conjunction with the 7th IEEE international conference on data mining (ICDM)*.
31. Nichols, J. M., & Michalowicz, J. V. (2017). Distance distribution between nodes in a 3D wireless network. *Journal of Parallel and Distributed Computing, 102*(2017), 71–79.
32. Kao, B., Lee, S. D., Lee, F., Cheung, D., & Ho, W. S. (2010). Clustering uncertain data using voronoi diagrams and R-tree index. *IEEE Transactions on Knowledge and Data Engineering*, *22*(9), 1219–1233.
33. Xie, X., Cheng, R., Yiu, M., Sun, L., & Chen, J. (2013). Uv-diagram: A voronoi diagram for uncertain spatial databases. *The VLDB Journal*, *22*(3), 319–344.
34. Elwin, M. L., Freeman, R. A., & Lynch, K. M. (2017). Distributed voronoi neighbor identification from inter-robot distances. *IEEE Robotics and Automation Letters, 2*(3), 1320–1327.
35. Zhou, D., Wang, Z., Bandyopadhyay, S., & Schwager, M. (2017). Fast, on-line collision avoidance for dynamic vehicles using buffered voronoi cells. *IEEE Robotics and Automation Letters*, *2*(2), 1047–1054.
36. Fan, P., Li, G., Cai, K., & Letaief, K. B. (2007). On the geometrical characteristic of wireless ad-hoc networks and its application in network performance analysis. *IEEE Transaction on Wireless Communications, 6*(4), 1256–1265.
37. Tong, F., Pan, J., & Zhang, R. (2016). Distance distributions in finite ad hoc networks: Approaches, applications, and directions. *Ad Hoc Networks*, *184*, 167–179.
38. Bettstetter, C., & Wagner, C. (2002). The spatial node distribution of the random waypoint mobility model. In *Proceedings of the German workshop on mobile ad hoc networks (WMAN)*.
39. Aschenbruck, N., Ernst, R., Gerhards-Padilla, E., & Schwamborn, M. (2010). BonnMotion—A mobility scenario generation and analysis tool. In *Proceedings of international conference on simulation tools andtechniques*.

V. S. Janani received her B.E. degree in Electronics and Communications Engineering from Anna University in 2009 and M.E. degree in Embedded System Technologies from Anna University in 2011. Since 2012 she has been pursuing her Ph.D. degree in the department of Electronics and Communication Engineering at Thiagarajar College of Engineering.

Dr. M. S. K. Manikandan received his B.E. degree in Electronics and Communications Engineering from National Institute of Technology. Trichy, Bharathidasan University in 1998 and M.E. degree in Communication System from Madurai Kamaraj University in 2000. He received his Ph.D. in Wireless Communication from Anna University; Chennai in 2010. He has been working as associate professor in Thiagarajar college of Engineering for years. He has been serving as reviewer for several IEEE conferences, Springer, IET and other international journals.