Filtering Dishonest Trust Recommendations in Trust Management Systems in Mobile Ad Hoc Networks

Zakir Ullah[1], Muhammad Hasan Islam[2], Adnan Ahmed Khan[3] and Sohail Sarwar[4]

[1,2]Department of Electrical and Computer Engineering, CASE, Islamabad, Pakistan
[3]College of Signals, National University of Sciences and Technology (NUST), Rawalpindi, Pakistan
[4]Department of Computing and Technology, Iqra University, Pakistan,
zakirmohmand@yahoo.com, mhasanislam@gmail.com, adnankhan@mcs.edu.pk, sohail.sarwar@seecs.edu.pk

***Abstract*:** Trust recommendations, having a pivotal role in computation of trust and hence confidence in peer to peer (P2P) environment, if hampered, may entail in colossal attacks from dishonest recommenders such as bad mouthing, ballot stuffing, random opinion etc. Therefore, mitigation of dishonest trust recommendations is stipulated as a challenging research issue in P2P systems (esp in Mobile Ad Hoc Networks). In order to cater these challenges associated with dishonest trust recommendations, a technique named "*i*ntelligently Selection of Trust Recommendations based on Dissimilarity factor (*i*STRD)" has been devised for Mobile Ad Hoc Networks. *i*STRD exploits personal experience of an "evaluating node" in conjunction with majority vote of the recommenders. It successfully removes the recommendations of "low trustworthy recommenders" as well as dishonest recommendations of "highly trustworthy recommenders". Efficacy of the proposed approach is evident from enhanced accuracy of "recognition rate", "false rejection" and "false acceptance". Moreover, experiential results depict that *i*STRD has unprecedented performance compared to contemporary techniques in presence of attacks asserted.

***Keywords*:** Bad Mouthing Attack, Ballot Stuffing Attack, MANET, Random Opinion Attack, Trust Management.

1. Introduction

Mobile Ad Hoc Network (MANET) [1] is a multi-hop and infrastructure-less wireless network of self-organizable mobile devices, having the unique characteristics such as open and shared wireless medium, absence of centralized controller, assumption of node's cooperation, node's mobility and node's limited resources in the form of battery power, processing power and memory [2]. These unique characteristics make MANET vulnerable to different kinds of Outsider attacks [3] (e.g. Spoofing [4] etc.**)** and Insider attacks [3] (e.g. Blackhole [5], Modification [3] etc.). To deal with these attacks, various cryptography [6-9] and trust management schemes [3] have been proposed. The focus of cryptography based security schemes is to protect the network from Outsider attacks while trust management schemes take a further step to protect the network from Insider attacks. These trust management schemes themselves are prone to different attacks such as bad mouthing, ballot stuffing, random opinion [10-12] etc. Our research is an attempt to shield trust management scheme(s) in MANETs from attacks stated, the very focus of this paper.

In MANET's trust management system, "evaluating node" develops trust on "evaluated node" from its behavioral information extracted by direct/personal observations [13, 14]. The trust developed through personal observation is known direct trust. Moreover, trust recommendations of other nodes in the network [15, 16] regarding behavior of an "evaluated node" may also be exploited by "evaluating node". These trust recommendations assist a node to get aware of the node's behavior that (a) is not in direct contact (b) has no previous trust relationship (c) strengthens personal (i.e., direct) trust. Trust develops through recommendations is known as indirect trust.

Mainly two methods are used for acquiring trust recommendations as given in the following:

- **Trust Solicitation**: In this method trust "evaluating node" requests certain node (recommender node) to share its trust value regarding an "evaluated node". For example, node *i* ("evaluating node") requesting recommender nodes (R1, R2, R3) to share their trust values about node *j* ("evaluated node"), as illustrated in Figure 1(a).
- **Trust Advertisement:** In this method the "recommender nodes" broadcast their trust values regarding other nodes in the network. For example, recommender nodes (R1, R2, R3) broadcasting their trust values about node *j* ("evaluated node"), as illustrated in Figure 1(b). Node *i* ("evaluating node") uses these recommendations to compute indirect trust about node *j* ("evaluated node").

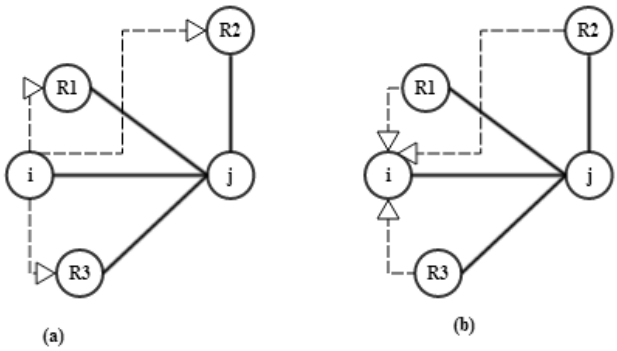

Figure 1. Acquiring trust recommendations in Trust Management System

Barring their assistance in trust management, these trust recommendations make trust management systems vulnerable to different attacks triggered by dishonest recommenders, i.e., recommenders give false trust recommendations deviating from their actual experiences [36] paving the way for attacks entailing in degraded performance of trust management system [17]. These attacks [10-12] are briefly described in following:

- **Bad mouthing attack:** also known as slandering attack. Here, dishonest recommender falsely shares decreased trust value of "evaluated node" to the "evaluating node".

Consequently, reputation of "evaluated node" is maligned.

- **Ballot stuffing attack:** also known as self-promoting attack. Here, dishonest recommender falsely shares increased trust value of "evaluated node" to the "evaluating node". Resultantly, reputation of "evaluated node" is boosted.
- **Random opinion attack:** is combination of bad mouthing and ballot stuffing attacks. Here, dishonest recommenders falsely share increased or decreased trust value of "evaluated node" to the "evaluating node". Subsequently, reputation of "evaluated node" appears to be highly ambiguous.

All of these stated attacks can either be launched by a dishonest recommender individually or in coalition with other dishonest recommenders. These attacks emphasize a great need for dealing with dishonest trust recommendations in trust management system. However, it is a challenging task since node's behavior is spread across the decentralized network and no single node ascertains information about behaviors of all nodes. Though diverse range of schemes have been proposed for the well-checked utilization of trust recommendations, which can be classified into weighted averaging based schemes e.g. [18-22], personal experience based schemes e.g. [23-28] and majority rule based schemes e.g. [29-31]. However, the weighted averaging based schemes do not try to eliminate the dishonest trust recommendations. The personal experience based schemes show inefficiency in situation when highly reputed smart attacker shared dishonest trust recommendations. Besides, the majority rule based schemes show inefficiency if majority of the recommenders are dishonest or the deviation in recommendation is minor.

Keeping in view issues of prevalent schemes, a dissimilarity factor based scheme, named *i*STRD (a quick glance is given in Figure 2), is proposed in this paper for filtering dishonest trust recommendations in trust management system in MANETs. A short version of this paper was presented in [32].

The proposed scheme combines personal experience (i.e., trust of "evaluating node" on recommender also known as trustworthiness of the recommender) and majority opinion (i.e., median of trust recommendations provided by recommenders) for filtering dishonest trust recommendations. In this scheme, "evaluating node" receives trust recommendations from all highly trustworthy and low trustworthy 1-hop neighbors and places these recommendations in trust recommendations set. These trust recommendations are then passed through a filter which detects and removes the dishonest trust recommendations. It removes the trust recommendations of low trustworthy nodes as well as falsely deviated trust recommendations of highly trustworthy nodes. After filtering dishonest trust recommendations, the scheme aggregates remaining recommendations using weighted averaging to get the overall aggregated trust. Effectiveness of the proposed scheme is evaluated in presence of bad mouthing, ballot stuffing and random opinion attacks.

Rest of the paper is organized as follows: Section 2 covers the literature review; Section 3 provides the detail implementation of proposed solution followed by results and evaluation in section 4. Section 5 concludes the work and provides future direction.

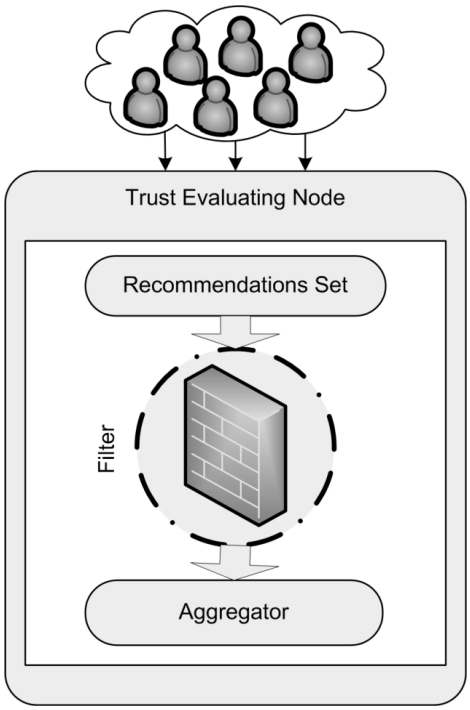

Figure 2. Overview of *i*STRD

2. Related Work

Utilization of trust recommendations and dealing with dishonest trust recommendations in trust management systems in MANETs is an open and challenging issue and it has attracted the efforts of many researchers in recent era. Different schemes have been proposed for treating dishonest trust recommendations and computing of indirect trust from trust recommendations in trust management system in MANETs [18-31]. These schemes can be classified into three categories: (a) weighted averaging based schemes [18-22], (b) personal experience based schemes [23-28] and (c) majority rule based schemes [29-31] [37].

2.1 Weighted Averaging Based Schemes

These schemes [18-22] work on the assumption that "everything is OK" i.e., these schemes do not grapple dishonest recommenders and just aggregate trust recommendations using trustworthiness of recommenders as weighting parameter in aggregating process. The authors in [18] [19] proposed four different methods for aggregating trust recommendations. The methods are: (a) optimistic or greedy approach (b) simple average of weighted product (c) weighted average (d) double weighted approach. Where, authors in [20-22] computed the indirect trust from trust recommendations using average of weighted trust. However, the availability of dishonest recommenders make these schemes [18-22] open to dishonest trust recommendations attacks such as ballot stuffing, bad mouthing and random opinion attack. Such attacks make the trust "evaluating node" to develop wrong indirect trust on "evaluated node".

2.2 Personal Experience Based Schemes

These schemes [23-28] accept trust recommendations from the highly trustworthy recommenders only. The trustworthy recommenders are such nodes whose trust level is greater than a certain threshold. After accepting the trust

recommendations from trustworthy recommenders, Qu et al. [23] aggregates the recommendations using simple average while authors [24-28] aggregate the trust recommendations using some weighting parameters in averaging process. Velloso et al. [24] employed maturity level and accuracy of recommendation as weighting parameters. Maturity level is the relationship maturity between recommender's and recommended node (i.e., "evaluated node"). This enables the trust "evaluating node" to give more significance to the trust recommendations provided by recommenders that are in mature relationship with the recommended node. The accuracy parameter handles variation in trust value that the recommenders has on the recommended node. Zakhary et al. [25] used degree of centrality and reputation of the recommenders as weighting parameters while Qureshi et al. [26] used trustworthiness of recommenders as weighting parameter. In [27], Xia et al. used recommender's and path credibility as weighting parameter for the aggregation of received trust recommendations. Recommender's credibility is the direct trust of "evaluating node" on recommender where path credibility is the credibility of all the recommenders in path through which trust recommendation is received. Similarly for computation of indirect trust, Chen et al. [28] used two schemes (a) threshold based filtering and (b) relevance based trust. In threshold based filtering, the recommendation that passes a threshold test from high trustworthy recommenders are considered where in relevance based filtering, trust from highly trustworthy recommenders in a particular context are considered. The recommendations that are passed through these filters are aggregated using weighted averaging.

These personal experience based schemes [23-28] are based on the assumption that highly trustworthy recommenders are always honest, hence these schemes only remove the trust recommendations of low trustworthy recommenders. However, this assumption is not always true. It is possible that a smart attacker can behave well for some time to get good reputation for itself. Once node develops good reputation for itself, it can start misbehaving and provide dishonest trust recommendations. Also, setting threshold for the selection of trustworthy recommenders is grave because of dynamic and decentralized characteristics of MANETs.

2.3 Majority Rule Based Schemes

These schemes try to eliminate or minimize the effect of dishonest trust recommendations. In these schemes [29-31], decision about the received trust recommendations is based on the opinion of majority. Such recommendations which are deviated from the majority opinion are treated as dishonest. For instance, in [29] the concept of court is introduced, i.e., performing a series of trials for mitigating the dishonest trust recommendations. In this scheme, the deviation detection module finds the deviation of received trust recommendation from the mean of received trust recommendations. The time verifying module is used to detect the correctness of deviation detection module by checking the impact of received trust recommendations on evaluated node's future behavior. Once these two modules decide about the behavior of recommenders, proof verifying module is used at the side of evaluated node to verify the correctness of trust recommendations. However, the deviation detection module performs 2^n comparisons where 'n' is the number of received trust recommendations. Similarly in proof verifying module, the "evaluating node" verifies the received trust recommendations from the "evaluated node". The dishonest "evaluated node" can agree with the trust recommendations provided by dishonest recommenders and disagree with the trust recommendations provided by honest ones.

In [30], the obtained trust recommendations are passed through an "evaluation difference" for minimizing the effect of dishonest trust recommendations. The evaluation difference finds the average of "absolute difference among the product of recommender's trustworthiness and trust recommendations of all the recommenders". Feng et al. [31] computed indirect trust by combining trust recommendations from others using D-S (Dempster and Shafer) theory. Intensity value, used in trust recommendations aggregation process, is calculated by finding distance between all trust recommendations and a recommendation that is far away from others will results in low intensity value. However, it does not remove deviated trust recommendations but gives less weightage to these in aggregation process. In [37], uncertainty in trust recommendations is handled with D-S theory. However, in this scheme recommender's credibility is not considered while obtaining trust recommendations.

Similarly, Iltaf et al. [11] proposed a scheme based on histogram and dissimilarity factor for removing the dishonest trust recommendations in pervasive computing. This scheme is based on assumption that frequency of dishonest trust recommendations is low as compared to honest trust recommendations. However, in this scheme width of bins is difficult to decide in histogram construction. In case of too wide bins, honest trust recommendations might be filtered out as dishonest ones. Similarly in case of too narrow bins, some dishonest trust recommendations might be treated as honest and vice versa.

These majority rule based schemes work under two conditions. First, number of honest recommenders is greater than the number of dishonest recommenders. Second, deviation in dishonest trust recommendations is sufficiently large as compared to majority opinions. However, dishonest recommenders may collude with one another, which results in increasing the number of dishonest recommenders. Also, the attackers can bypass the detection mechanism by introducing a relatively small deviation in dishonest trust recommendations.

Contrary to existing schemes, the proposed scheme neither merely rely on personal experience of "evaluating node" nor on the majority rule of the recommenders but combine both to filter out the dishonest trust recommendations. After filtering dishonest trust recommendations, the proposed scheme aggregates remaining trust recommendations using weighted averaging. This weighted averaging enables trust "evaluating node" to give different weight to different trust recommendations based on trustworthiness of recommenders.

3. Proposed Architecture of *i*STRD

The objective of the proposed scheme, "***i***ntelligently **S**election of **T**rust **R**ecommendations based on **D**issimilarity factor (*i*STRD)", is to assist trust management system in Mobile Ad Hoc Networks such as [16] [20] [28] in filtering-out dishonest trust recommendations provided by dishonest recommenders. The dishonest trust recommendations are the outliers that are inconsistent with other recommendations and

are recommended for launching dishonest trust recommendation attacks, i.e., bad mouthing, ballot stuffing and random opinion. These attacks can either be launched by dishonest recommenders individually or in coalition with other dishonest recommenders.

The role of proposed scheme comes in play once the "evaluating node" received trust recommendations from neighbors about the "evaluated node" in a trust management system. Detailed procedure of the proposed scheme, as given in Figure. 3, is demonstrated at finer level of granularity in following section. The main components of the proposed scheme are *Recommenders*, *RecommendationsFilter* and *RecommendationsAggregator*. *Recommenders* are the sources of trust recommendations; *RecommendationsFilter* segregates the dishonest recommendations from the honest ones while *RecommendationsAggregator* combines the honest trust recommendations to produce the indirect trust. Mathematical notations used by all these components are enlisted in Table 1.

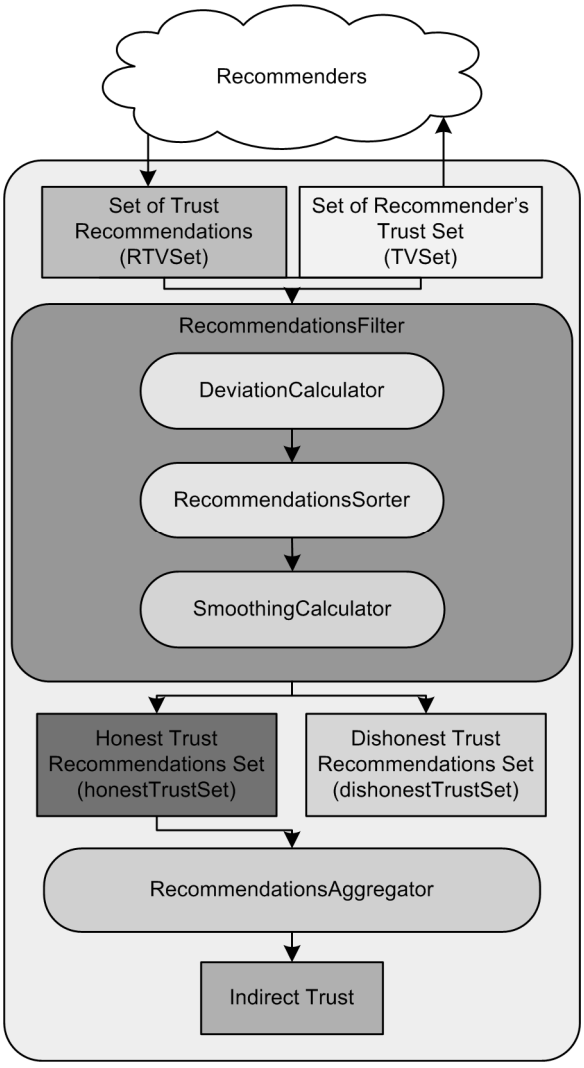

Figure 3. Detail architecture of *i*STRD

In the proposed scheme, a set of trust recommendations is received by "evaluating node" regarding "evaluated node" from all 1- hop neighboring recommenders, not only from trustworthy recommenders. If only trustworthy recommenders are considered for trust recommendations then a constant would be defined for deciding about trustworthiness of the recommenders that is not befitted in dynamic behavior of MANETs. The proposed scheme gives priority to highly trustworthy recommenders over low trustworthy recommenders and automatically detects the trust recommendations of low trustworthy recommenders.

Table 1. Mathematical notations used in *i*STRD

Notation	Meaning
C	Cardinality of set
DF	Dissimilarity factor
dishonestTrustSet	Dishonest trust recommendations set
honestTrustSet	Honest trust recommendations set
RTV	Trust recommendation
RTVSet	Set of trust recommendations
SF	Smoothing factor
SFmax	Maximum smoothing factor
SortRTVSet	Sorted set of trust recommendations
STSet	Suspected trust set
TVSet	Set of recommender's trust
TV	Recommender's trust

Set of trust recommendations provided by *recommenders* are represented by *RTVSet* and the coinciding trust values of "evaluating node" on *recommenders* are represented in *TVSet*. The trust values of "evaluating node" on *recommenders* represent trustworthiness of the *recommenders*.

$$RTVSet = \{RTV_i\}; i = 1, 2, ...n$$
$$TVSet = \{TV_i\}; i = 1, 2, ...n$$

Here, RTV_1 is the trust recommendation provided by $recommender_1$ and TV_1 is the trust value of the trust "evaluating node" on $recommender_1$. Similarly, RTV_2 is the trust recommendation provided by $recommender_2$ and TV_2 is the trust value of the trust "evaluating node" on $recommender_2$ and so on till '*n*' number of *recommenders*.

In case, *recommender* is new in the vicinity of "evaluating node" and "evaluating node" does not have trust (*TV*) on *recommender* then ignorance trust value is used. Ignorance trust value is 0.5 if TV $\in$ [0-1].

3.1 RecommendationsFilter

Trust sets, i.e., *RTVSet* and *TVSet* are passed to *RecommendationsFilter* which yields two sets of trust recommendation, honest as well as dishonest trust recommendation sets. The honest trust recommendations set contains the honest (i.e., non-malicious) recommendations while the dishonest recommendations set contains the dishonest (i.e., malicious) recommendations.

The *RecommendationsFilter* further comprises of *DeviationCalculator, RecommendationsSorter* and *SmoothingCalculator*. Working phenomenon of *RecommendationsFilter* is stipulated in **Pseudo Code 1**. Each of its functional components is described in succeeding sections.

Pseudo Code 1: RecommendationFilter

Input:
Set of Trust Recommendations (*RTVSet*)
Set of Recommender's Trust (*TVSet*)
Output:
Dishonest Trust Recommendations Set (*dishonestTrustSet*)
Honest Trust Recommendations Set (*honestTrustSet*)
Variables:
SortRTVSet //Sorted set of trust recommendations
DF //Dissimilarity factor
STSet //Set of suspected trust recommendations
Procedure:
1: [*DF*] = DeviationCalculator(*RTVSet*, *TVSet*)
2: [*SortRTVSet*] = RecommendationSorter(*RTVSet*, *DF*)
3: [*dishonestTrustSet*, *honestTrustSet*] = SmoothingCalculator(*SortRTVSet*, *DF*)
4: **Return** *dishonestTrustSet*, *honestTrustSet*

3.1.1 DeviationCalculator

DeviationCalculator finds Dissimilarity Factor (deviation) of each received trust recommendation (*RTV*) via Equation (1).
Definition: Dissimilarity Factor (*DF*) is the ratio of "squared difference of received trust recommendation from median of received trust recommendations set" and "trust of evaluating node on recommender".

$$DF = \frac{[\text{RTV-median}]^2}{\text{TV}} \quad (1)$$

It's worth mentioning that *DF* combines majority opinion (i.e., median of trust recommendations set) and personal experience (i.e., trust of "evaluating node" on recommenders). Employing personal experience with majority opinion induces trustworthiness of recommenders while deciding about the trust recommendations. *DF* based only on majority opinion may hamper performance of trust management system at some point if majority of recommenders are malicious or the deviation in recommendation is low.
Similarly, *DF* based only on personal experience, a smart attacker may deliberately feed malicious trust values to "evaluating node" after winning its confidence. So this combination of majority opinion and personal experience while calculating *DF* greatly reduces the potential negative impact of majority opinion/ personal experience when employed in segregation.
Here in Equation (1), median is used instead of mean as median is not affected by deviated trust recommendations [11]. The impact of deviation is further signified by taking square of deviation from the median (since the squared difference would be larger if a recommendation is farther from the median).
In order to detect the trust recommendation provided by low-trustworthy nodes and highly deviated trust recommendation provided by trustworthy nodes acting as malicious, trust of "evaluating node" on recommender (TV) is used which divides $[RTV - median]^2$ in Equation (1). As, trust *TV* of low-trustworthy node is low, so the *DeviationCalculator* produces high *DF*. Similarly, *TV* of trustworthy node, acting as malicious, is high but the dishonest trust recommendation of the said node results in higher value of $[RTV - median]^2$ compared to value of $[RTV - median]^2$ for trustworthy node provides honest trust recommendation. A trust recommendation that results in higher *DF* is considered to be more suspected.
Pseudo code of *DeviationCalculator* exploiting Equation (1) for Dissimilarity Factor is given in **Pseudo Code 2**.

Pseudo Code 2: DeviationCalculator

Input:
Set of Trust Recommendations (*RTVSet*)
Set of Recommender's Trust (*TVSet*)
Output:
Sorted set of Trust Recommendations (*SortRTVSet*)
Variables:
TV //Recommender's trust value
RTV //Recommended trust value
median //Median of the received recommendation set
DF //Dissimilarity factor
Procedure:
1: *median* = Median(*RTVSet*)
2: **for** i = 1 to Size(*RTVSet*) **do**
3: $DF = \frac{[\text{RTV-median}]^2}{\text{TV}}$
4: **end for** // end of loop (line 2)
5: **Return** *DF*

3.1.2 RecommendationsSorter

Once Dissimilarity Factor (*DF*) of each received trust recommendation (*RTV*) is computed, they are arranged in descending order with respect to *DF*. Highly deviated trust recommendations are enlisted at top of the list as a result of sorting. In this sorted list (*SortRTVSet*), trust recommendations at the top having the highest *DF* are considered to be suspicious. Now to segregate dishonest trust recommendations from the honest ones, this sorted list is subjected to *SmoothingCalculator*.

3.1.3 SmoothingCalculator

In order to find the set of dishonest trust recommendations, sorted trust recommendations set is passed through *SmoothingCalculator*. The result of *SmoothingCalculator* is termed as Smoothing Factor (*SF*). *SF* indicates degree of dissimilarity that can be reduced by removing the suspicious trust recommendations from the whole trust recommendations set. Operational mechanism of *SmoothingCalculator* is based on the procedure in [33] to figure out *SF* of all subsets from sorted trust recommendations set. These subsets are termed as suspected trust set (*STSet*). The baseline expression used in *SmoothingCalculator* is given as Equation (2).

$$SF = | C(SortRTVSet - STSet) * \{DF(SortRTVSet) - DF(SortRTVSet - STSet)\} | \quad (2)$$

Where
k = 1 to n-1 where 'n' is the total number of trust recommendations in the sorted trust recommendations set (*SortRTVSet*).
C is the Cardinality that equals SizeOf{(*SortRTVSet* – *STSet*)}.
Once *SF* of each suspected trust set (*STSet*) is computed, *i*STRD declares the smallest suspected trust set having the

maximum *SF* as dishonest trust recommendations set (*dishonestTrustSet*). Mathematically:

If

$$\{STSet[k] \subseteq SortRTVSet\} \&$$
$$\{SF(STSet[k] \geq SF(STSet[j])\}$$

Then

$$STSet[k] \rightarrow dishonestTrustSet$$

Where $k, j \in n$.

Now to find honest trust recommendations set (*honestTrustSet)*, the dishonest trust recommendations set (*dishonestTrustSet*) is separated from the sorted trust recommendations set (*SortRTVSet*).

If *SmoothingCalculator* is not employed for pruning of honest trust recommendations from dishonest ones, a static (constant) threshold would be defined for deciding upon *DF* value that may not comprehend dynamic nature/behavior of MANETs.

Pseudo Code 3 finds Smoothing Factor (*SF*) of all suspected trust set, i.e., $SF(STSet[k] \subseteq SortRTVSset)$.

Pseudo Code 3: SmoothingCalculator

Input:
Sorted Recommendations Set (*SortRTVSet*)
Dissimilarity Factor (*DF*)
Output:
Dishonest Trust Recommendations Set (*dishonestTrustSet*)
Honest Trust Recommendations Set (*honestTrustSet*)
Procedure:
//initially the suspected trust set is empty
1: *STSet*[0] = {}
2: **for** k = 1 to Size(*SortRTVSet*) -1 **do**
3: *STSet[k] = STSet[k-1] U SortRTVSet[k]*
4: *SF[k] = |C(SortRTVSet – STSet[k]) * {DF(SortRTVSet) – DF(SortRTVSet – STSet[k])}|*
5: **end for** // end of loop (line 2)
// finds max smoothing factor
6: [*SFmax*] = Maximum (*SF*)
// finds the dishonest recommendation set
7: *dishonestTrustSet* = Smallest *STSet* with *SFmax*
// finds the honest recommendation set.
8: *honestTrustSet* = *SortRTVSet - dishonestTrustSet*
9: **Return** *dishonestTrustSet, honestTrustSet*

3.2 RecommendationsAggregator

Once *RecommendationsFilter* separates the honest trust recommendations from the dishonest ones, *RecommendationsAggregator* is used to compute the overall indirect trust of "evaluated node". *RecommendationsAggregator* uses weighted averaging for this step as given in Equation (3).

$$T_{i,k}^{ind} = \frac{\sum_{j=1}^{N} TV_{i,j} * RTV_{j,k}}{N} \quad (3)$$

Where, $T_{i,k}^{ind}$ is the aggregated indirect trust, $TV_{i,j}$ is the trust of "evaluating node" on the recommender, $RTV_{j,k}$ is the recommended trust and N is the total number of honest recommenders.

4. Performance Evaluation

In this section performance of the proposed scheme is presented, where model for proof of concept is implemented using NS-2.34 [34] as simulator. In simulation, [35] is used to develop and recommend the trust and then the proposed scheme is used to compute the indirect trust from the recommended trust. For performance analysis, the proposed scheme is compared with three indirect trust computation schemes, i.e., weighted averaging scheme [20], personal experience based scheme [28] and majority rule based scheme [30] referred as WAS, PES and MOS respectively. Varity of experimental evaluations is carried out for measuring the effectiveness of *i*STRD in the presence of bad mouthing, ballot stuffing and random opinion attacks.

Simulation scenarios are executed over network of 50 nodes. Each node belongs to one of three categories, i.e.,

(a) Evaluating nodes: compute the trust of a node
(b) Evaluated nodes: nodes under observation
(c) Recommenders: provide trust recommendations about evaluated nodes

Recommenders are further classified into two categories, i.e.,

(1) Honest recommenders: are trustworthy nodes and don't try to launch any attack.
(2) Dishonest recommenders: are low-trustworthy or well-trustworthy nodes and are capable of launching any of three attacks, i.e., bad mouthing, ballot stuffing or random opinion.

In simulations, 50% of the total nodes in the network are selected as recommenders. Out of these 50% recommenders, up to 48% of recommenders are acting as dishonest.

All scenarios are simulated over 100 rounds where the time span of each round is 1000 seconds. In each round, nodes computed the trust of 1-hop neighbor nodes by sending 100 packets to it and then observed its forwarding behavior. At the end of each round, these nodes recommended the computed trust about "evaluated node" to "evaluating node". The "evaluating node" then evaluated the trust recommendations for the detection of dishonest trust recommendations using the proposed scheme.

Three baseline factors are employed for performance evaluation of the proposed scheme, i.e., "Percentage of dishonest recommenders", "Recommendation Deviation ($\Delta\%$)" and "Mean Offset (MO)".

- **Percentage of dishonest recommenders:** is the percentage portion of dishonest recommenders out of total recommenders.
- **Recommendation Deviation** ($\Delta\%$)**:** is the deviation in recommendation value provided by recommender from actual trust value of a node (i.e., "evaluated node").
  If actual trust value of a node is *T* and recommendation deviation is $\Delta\%$ then the recommendation with bad mouthing, ballot stuffing and random opinion attack is represented using Equation (4), (5) and (6) respectively.

$$RTV = T - (T * \Delta\%) \quad (4)$$

$$RTV = T + (T * \Delta\%) \quad (5)$$

$$RTV = T \pm (T * \Delta\%) \quad (6)$$

- **Mean Offset (MO):** is the difference between mean of honest trust recommendations and mean of dishonest trust recommendations, as given in Equation (7).

$$MO = |Mean(HR) - Mean(DR)| \quad (7)$$

Where,
HR = Set of Honest Trust Recommendations
DR = Set of Dishonest Trust Recommendations
Here, its worth mentioning that "Recommendation Deviation" and "Mean Offset" are directly proportional to each other as represented by Equation (8).

$$\%\Delta \propto MO \quad (8)$$

It implies that if "Recommendation Deviation" is high then "Mean Offset" will also be high and vice versa. Using these three factors, different experiments are performed to observe behavior of the proposed scheme.

The default parameters setting for the simulations are given in Table 2.

Table 2. Default simulations parameters

Parameter	Default Value
Simulation Area	750m x 750m
Simulation Rounds	100
Simulation time	1000 seconds
Number of Nodes	50
% of Recommenders	50
% of Dishonest Recommenders	0
% of Recommendation Deviation	0
Attack	None

4.1 An Illustrative Example

In order to provide a step by step insight to working phenomenon of *i*STRD for filtering dishonest trust recommendations, following illustrative example is presented. This example considers scenario of random opinion attack. Here, "low trustworthy" as well as "highly trustworthy" recommenders act as random opinion attackers.

Let node 'A' has received trust recommendations $RTV_{i=1to10}$ about node 'B' from recommenders $R_{i=1to10}$ where $TV_{i=1to10}$ are the trust values of node 'A' on recommenders $R_{i=1to10}$.

These $RTV_{i=1to10}$ with corresponding $TV_{i=1to10}$ are placed in *RTV* and *TV* sets respectively, as given in following:

$$RTV = \{0.99, 0.2, 0.667, 0.7, 1, 0.2, 0.7, 0.767, 0.756, 0.789\}$$

$$TV = \{0.3, 0.6, 0.7867, 1, 0.3, 1, 0.8, 0.8675, 0.6754, 0.7554\}$$

Once *median* of the *RTV* set is computed, Equation. (1) (in **Pseudo Code 2**) is used to find dissimilarity factor (*DF*) of each trust recommendation. *DF* is calculated for all trust recommendations which are then sorted in descending order reference to *DF*. These sorted trust recommendations with corresponding *DF* are shown in Column 1 and 3 respectively (Refer Table 3).

In order to calculate smoothing factor (*SF*), sorted trust recommendations set along with respective *DF* is subjected to **Pseudo Code 3**. Every suspected trust set (*STSet*) with corresponding smoothing factor (*SF*) is shown in Column 4 and 9 respectively (Refer Table. 3).

Suspected trust set (*STSet* = {0.2, 0.2, 1, 0.99}) appears to have the highest smoothing factor (*SF* = 7.313104) in Table 3, so this set is filtered out as dishonest trust recommendations set. The set {0.789, 0.667, 0.767, 0.756, 0.7, 0.7} is declared as honest trust recommendations set.

4.2 Effect of Dishonest Trust Recommendations

An experiment is conducted to measure the effects of dishonest trust recommendations without employing any filtering mechanism prior to performance analysis of *i*STRD.

In this experiment, an average trust metric is defined using Equation (9). This equation is used to examine the average trust of nodes in presence of dishonest recommenders working in coalition for launching bad mouthing, ballot stuffing and random opinion attacks. Based on trust value of nodes, three of them are declared as good (node 7 with actual trust of 0.89), bad (node 12 with actual trust of 0.41) and average (node 21 with actual trust of 0.5) respectively. Node 7 endured bad mouthing, node 12 suffered ballot stuffing and node 21 faced random opinion attack.

$$\overline{T_i}(r) = \frac{\sum_{j=1}^{N} \mathrm{RTV}_{j,i}}{\mathrm{N}} \quad (9)$$

Where,

$RTV_{j,i}$ = Trust Recommendation of node *j* for node *i*
i = 7, 12, 21
N = Total Recommenders
r = 1 to 100 (round index)

With this metric, effects of bad mouthing, ballot stuffing and random opinion attacks are evaluated with varying percentage of dishonest recommenders and recommendation deviations. Dishonest recommenders are employed in low as well as in high percentages, i.e., 8% and 40%, whereas recommendation deviations have varying percentage values of 20% and 80%, i.e., low and high deviation.

The results of node 7, 12 and 21 in terms of average trust in presence of stated attacks are shown in Figure 4(a)-(c) respectively. Average trust is also computed in absence of dishonest recommenders (i.e., 0% dishonest recommenders) for the sake of comparison. In this situation it is observed that average trust values of node 7, 12 and 21 quickly converged to 0.89, 0.41 and 0.5 respectively.

In case of bad mouthing and ballot stuffing attacks, percentage increase in dishonest recommenders (i.e., 0% to 40%) and recommendation deviation (i.e., 0% to 80%) causes false manipulation in average trust values of node 7 and 12, as shown in Figure 4(a)-(b). With low percentage of dishonest recommenders (i.e., 8%) and recommendation deviation (i.e., 20%), the average trust values of node 7 and 12 congregated to 0.87 and 0.43 respectively. However, with high percentage of dishonest recommenders (i.e., 40%) and recommendation deviation (i.e., 80%), the situation is worse.

Table 3. An illustrative example of *i*STRD

RTV	TV	DF	STSet	SortRTVSet-STSet	DF(SortRTVSet)	DF(SortRTVSet-STSet)	C	SF
0.2	0.6	0.46464	{0.2}	{0.2, 1, 0.99, 0.789, 0.667, 0.767, 0.756, 0.7, 0.7}	1.233185	0.768545	9	4.18176
0.2	1	0.278784	{0.2, 0.2}	{1, 0.99, 0.789, 0.667, 0.767, 0.756, 0.7, 0.7}	1.233185	0.489761	8	5.947392
1	0.3	0.246613	{0.2, 0.2, 1}	{0.99, 0.789, 0.667, 0.767, 0.756, 0.7, 0.7}	1.233185	0.243147	7	6.930261
0.99	0.3	0.228813	{0.2, 0.2, 1, 0.99}	{0.789, 0.667, 0.767, 0.756, 0.7, 0.7}	1.233185	0.014334	6	7.313104
0.789	0.7554	0.004926	{0.2, 0.2, 1, 0.99, 0.789}	{0.667, 0.767, 0.756, 0.7, 0.7}	1.233185	0.009408	5	6.118883
0.667	0.7867	0.00473	{0.2, 0.2, 1, 0.99, 0.789, 0.667}	{0.767, 0.756, 0.7, 0.7}	1.233185	0.004678	4	4.914026
0.767	0.8675	0.001753	{0.2, 0.2, 1, 0.99, 0.789, 0.667, 0.767}	{0.756, 0.7, 0.7}	1.233185	0.002925	3	3.690779
0.756	0.6754	0.001161	{0.2, 0.2, 1, 0.99, 0.789, 0.667, 0.767, 0.756}	{0.7, 0.7}	1.233185	0.001764	2	2.462841
0.7	0.8	0.00098	{0.2, 0.2, 1, 0.99, 0.789, 0.667, 0.767, 0.756, 0.7}	{0.7}	1.233185	0.000784	1	1.232401
0.7	1	0.000784	-	-	-	-	-	-

The average trust values of node 7 and 12 are congregated to 0. 61 and 0.56 respectively, which are gravely digressed from the actual trust values.

This fact ascertains the expectation, more dishonest recommenders wreak more harm to trust management systems in the absence of a filtering scheme. Furthermore, increase in recommendation deviation hampers performance of trust management systems if dishonest trust recommendations are not filtered.

Another affirmation is that, low percentage of dishonest recommenders with high percentage of recommendation deviation produces approximately the same impact as that of high percentage of dishonest recommenders and low percentage of recommendation deviation. This is evident from 8% dishonest recommenders with 80% recommendation deviation and 40% dishonest recommenders with 20% recommendation deviation.

In case of random opinion attack, the average trust values of node 21 are shown in Figure 4(c). Here, after 10 rounds the percentage of bad mouthers and ballot stuffers changes from 10% to 30% and 30% to 10% respectively. The increase in percentage of bad mouthers decreases average trust value and increase in percentage of ballot stuffers increases the average trust value. Also, increase or decrease in average trust is less or more depends on percentage of recommendation deviation, as obvious from 20% and 80% recommendation deviation respectively. Random opinion attack as envisaged end up with fluctuation in average trust of node from high to low and vice versa, depending upon percentage of dishonest recommenders, distribution percentage of bad mouthers & ballot stuffers and percentage of recommendation deviation.

Results in Figure 4(a)-(c) also affirm that average trust of node converges to certain value after some iterations and does not change in subsequent iterations, provided that percentage of dishonest recommenders and recommendations deviation remain the same.

4.3 Detection Ratio

Detection Ratio is the function of three metrics, namely recognition percentage (RP), false negative percentage (FNP) and false positive percentage (FPP). It is used to evaluate performance of the *i*STRD. These metrics are represented by Equation. (10) - (12).

$$RP = \frac{\sum \text{Recgnized dishonest trust recommendations}}{\text{All dishonest trust recommendations}} * 100 \quad (10)$$

$$FNP = 100 - RP \quad (11)$$

$$FPP = \frac{\sum \text{Honest trust recommendations detected as dishonest}}{\text{All honest trust recommendations}} * 100 \quad (12)$$

Any successful defense scheme is expected to have high RP with low FNP and FPP.

Performance of *i*STRD in reference to Detection Ratio is furnished in Figure 5. Since proposed scheme exploits median for computing dissimilarity factor so maximum percentage of dishonest recommenders is restricted to 48%. In these scenarios, the scheme is evaluated under highly deviated as well as low deviated dishonest recommendations.

Initially, results are recorded with 80% recommendation deviation (high deviation) under different attacks, i.e., bad mouthing, ballot stuffing and random opinion, by continually incrementing dishonest recommenders up to 48%, as shown in Figure 5(a). Consequently, the lowest mean offset (MO) for bad mouthing attack is 0.6999, for ballot stuffing attack is 0.6181 and for random opinion attack is 0.6849. Figure 5(a) presents the effectiveness of proposed scheme with 100% RP, 0% FNP and FPP, despite of increase in dishonest recommenders up to 48%.

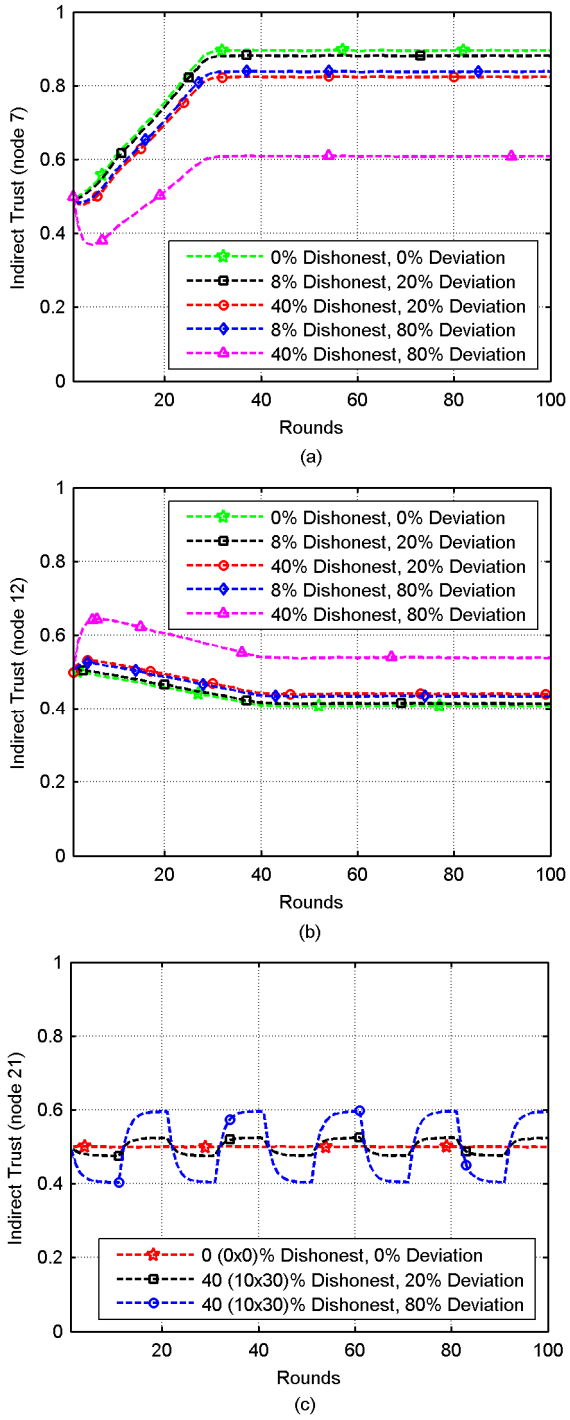

Figure 4. (a). Effect of bad mouthers on trust of node 7, (d). Effect of ballot stuffers on trust of node 12, (c). Effect of random opinions on trust of node 21

After evaluating the performance for high recommendation deviation, recommendation deviation is remarkably decreased. The recommendation deviation is set to 20% (low deviation) with mean offset of 0.42, 0.376 and 0.394 for bad mouthing, ballot stuffing and random opinion attacks respectively.

Proposed scheme gets 100% RP and 0% FNP and FPP in presence of up to 44% dishonest recommenders. However, RP decreases to 91% and FNP increases to 9% when percentage of dishonest recommenders is increased beyond 44%, as shown in Figure 5(b). Moreover, FPP is still 0% in presence of up to 48% dishonest recommenders, though low recommendations deviation is employed.

Results in Figure 5 persistently exhibit similar behavior in all the three types of attacks, it clearly implies that *i*STRD is not sensitive to type of dishonest recommenders. But its performance relies on percentage of dishonest recommenders, recommendation deviation and mean offset (MO).

4.4 Comparative Analysis

Here comparative analysis of *i*STRD is presented with three indirect trust computation schemes, i.e., weighted averaging scheme [20], personal experience based scheme [28] and majority rule based scheme [30] referred as WAS, PES and MOS respectively in simulation.

Aggregated indirect trust of the evaluated node is presented for the sake of better comparison in absence of dishonest recommendations using weighted aggregating metric as given in Equation. (12).

$$T_{i,k}^{agg} = \frac{\sum_{j=1}^{N} TV_{i,j} * RTV_{j,k}}{} \quad (12)$$

Where,

$T_{i,k}^{agg}$ = aggregated indirect trust

$TV_{i,j}$ = trust of trust evaluating node on recommender

$RTV_{j,k}$ = recommended trust

N = Total recommenders whose trust recommendations are used in aggregation

Firstly, bad mouthing attack is launched by attackers and evaluated node's (node 7) aggregated indirect trust is examined with various percentages of bad mouthers and recommendation deviations. Here in this scenario, bad mouthers are in combination of low trustworthy ("evaluating node" trust on the recommender $\leq$ 0.4) and trustworthy ("evaluating node" trust on the recommender > 0.4) recommenders. The results of the simulations are given in Figure 6(a)-(d).

We can observe that *i*STRD accurately computes the evaluated node's aggregated indirect trust as evident from these four figures, i.e., aggregated indirect trust is same as that of aggregated indirect trust in absence of bad mouthers.

Performance of *i*STRD in comparison with WAS, PES and MOS proves better in all scenarios of bad mouthers. *i*STRD removed all bad mouthers and computed same aggregated indirect trust as in absence of bad mouthers. Contrary to *i*STRD, WAS does not remove bad mouthers and just aggregates received recommendations using trust of "evaluating node" on the recommender as weighting parameter. As expected, the indirect trust of "evaluated node" decreases due to presence of bad mouthers.

In PES, evaluating node removes low trustworthy recommenders and aggregates recommendations from the highly trustworthy recommenders. This scheme shows inefficiency due to bad mouthing behavior of highly trustworthy recommenders, as shown in Figure 6(a)-(d). The scheme aggregated indirect trust is greater than the WAS but less than the actual indirect trust of the "evaluated node". This facet is due to removal of low trustworthy recommenders only and not removing highly trustworthy recommenders acting as bad mouthers.

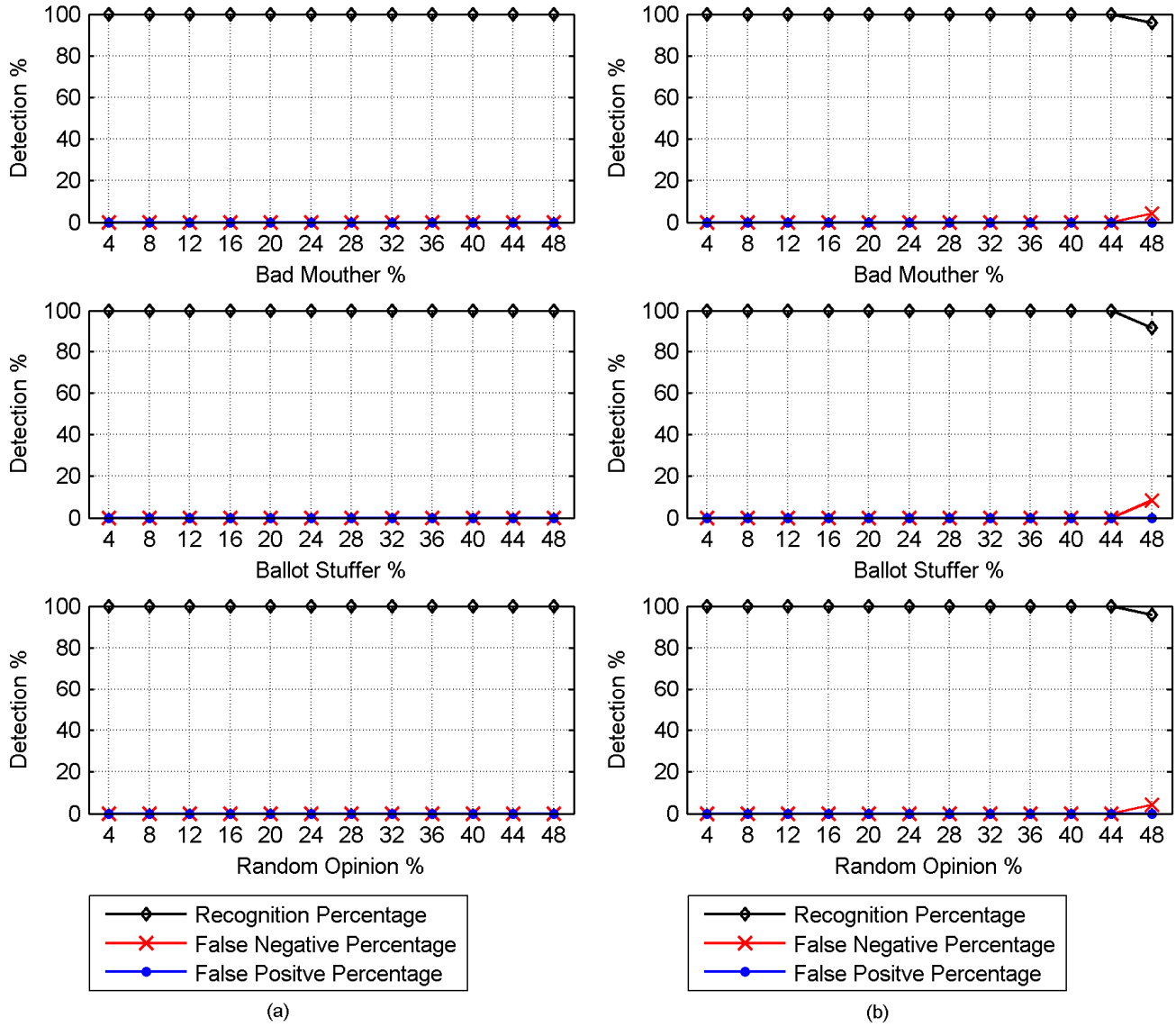

Figure 5. Recognition percentage, false negative and false positive of bad mouthers, ballot stuffers and random opinions respectively, (a). 80% Recommendation deviation, (b). 20% Recommendation deviation

MOS gives less weightage to deviated recommendations and does not remove the bad mouthers. As shown in Figure 6(a)-(d), the scheme shows inefficiency because of the availability of low trustworthy and highly trustworthy bad mouthers.

In second scenario, only ballot stuffing attack is launched by attackers. Evaluated node's (node 12) aggregated indirect trust is examined under varying percentage of ballot stuffers and recommendation deviation. Here, ballot stuffers are in combination of low trustworthy (evaluating node trust on the recommender ≤ 0.4) and trustworthy (evaluating node trust on the recommender > 0.4) recommenders. Figure 6(e)-(h) shows results of the simulations. *i*STRD filters out all the ballot stuffers and accurately evaluate the evaluated node's aggregated indirect trust as envisioned, i.e., the aggregated indirect trust is same as that of aggregated indirect trust in absence of ballot stuffers. However, performance of MOS is better in this scenario than that of WAS and PES because of giving less weightage to deviated recommendations.

In third scenario, evaluated node's (node 21) aggregated indirect trust is examined in the presence of varying percentage of bad mouthers and ballot stuffers in order to launch random opinion attack.

Evaluated node's aggregated indirect trust using *i*STRD is the same as that of node's aggregated indirect trust in the absence of dishonest recommenders as attested by Figure 6(i)-(j). The accurate computation of evaluated node's aggregated indirect trust makes us clear that *i*STRD filters out all dishonest recommendations in the form of random opinions. Moreover, proposed scheme gives better performance than all the three comparative schemes for removing random opinion attackers.

Conclusively stating *i*STRD is a good choice under various parameters of dishonest recommenders and recommendations deviation for removing all three types of attacks. Furthermore, it gives similar results in presence of all three types of attacks as evident from results. Results ensure that the proposed scheme is not sensitive to type of dishonest recommendation attacks.

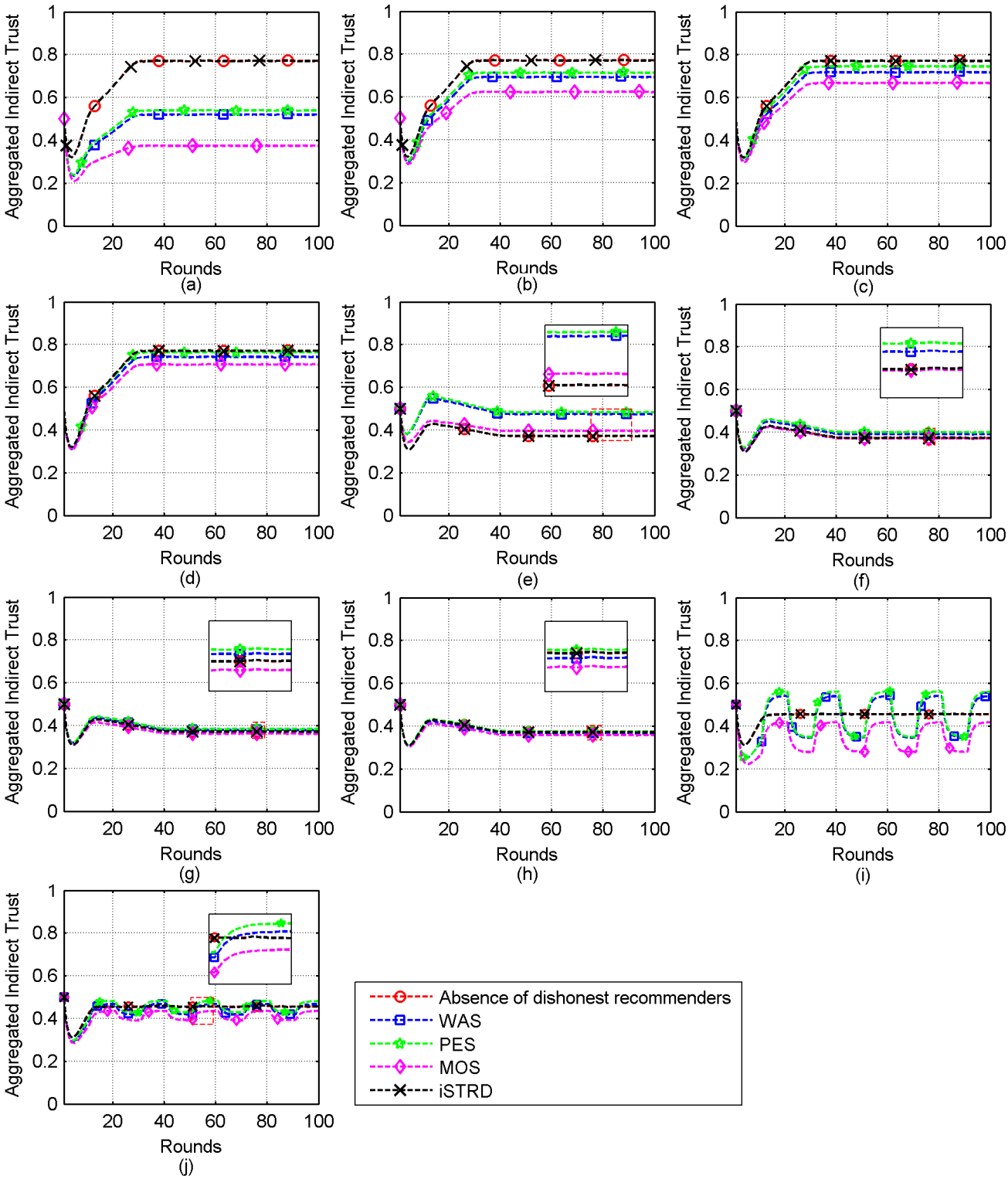

Figure 6. Comparisons of *i*STRD with WAS, PES and MOS. (a)-(b). 40% bad mouthers with 80% and 20% recommendation deviation respectively, (c)-(d). 8% bad mouthers with 80% and 20% recommendation deviation respectively, (e)-(f). 40% ballot stuffers with 80% and 20% recommendation deviation respectively, (g)-(h). 8% ballot stuffers with 80% and 20% recommendation deviation respectively, (i)-(j). 40% random opinion attackers with 80% and 20% recommendation deviation respectively. In random opinion, bad mouthers and ballot stuffers are in ratio of (8%, 32%) and vice versa.

5. Conclusion

In this paper, a scheme named "*i*ntelligently Selection of Trust Recommendations based on Dissimilarity factor (*i*STRD)" is presented in order to shield trust management system in MANETs against dishonest recommendations.

This scheme, utilizing personal experience of "evaluating node" and majority opinion of the recommenders, is based on median based deviation of trust recommendations and trustworthiness of recommenders for filtering dishonest recommendations. Smoothing factor is used for removal of deviated recommendations having high impact on the overall recommendations. The proposed scheme is featured to mitigate dishonest recommendations of low trustworthy as well as highly trustworthy recommenders, which has not been catered so far to the best of our knowledge. Moreover, our approach does not require any constant or threshold while deciding about dishonest recommendations.

Effectiveness of the proposed scheme is evident from efficient detection of dishonest recommendations shared by bad mouthers, ballot stuffers and random opinion attackers. Simulation results ascertain that our scheme differentiates 100% the dishonest recommendations from the honest ones in the presence of up to 44% dishonest recommenders with as low as 20% recommendation deviation. Similarly, in situation of highly deviated dishonest recommendations, our scheme differentiates 100% the dishonest recommendations from the honest ones in the presence of up to 48% dishonest recommenders. Furthermore, *i*STRD outperforms existing schemes by computing the accurate indirect trust in presence of bad mouthers, ballot stuffers and random opinion attackers with varying percentage of dishonest recommenders and recommendations deviation. Lastly, proposed scheme is flexible enough to work efficiently regardless of the type of attacks by exhibiting uniform results for the asserted attacks and can be used with any trust management system in MANETs.

In future, we look forward to incorporate the proposed scheme in our in-progress multi-factors trust management system for defending routing in MANETs from insider attackers. This step will be followed by carrying out stipulated simulations in real time environment to affirm usefulness of proposed approach after integration in our in-progress multi-factors trust management system.

References

[1] S. Corson, J. Macker, "Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations," RFC 2501, 1999.

[2] I. Chlamtac, M. Conti, J.-N. L. Jennifer, "Mobile ad hoc networking: imperatives and challenges," Ad Hoc Networks, Vol. 1, Issue. 1, pp. 13-64, 2003.

[3] J. H. Cho, A. Swami, I. R. Chen, "A Survey on Trust Management for Mobile Ad Hoc Networks," IEEE Communications Surveys & Tutorials, Vol. 13, No. 4, pp. 562-583, 2011.

[4] D. Djenouri, L. Khelladi, A.N. Badache, "A survey of security issues in mobile ad hoc and sensor networks," IEEE Communications Surveys &Tutorials, Vol. 7, Issue. 4, pp. 2–28, 2005.

[5] Y. Khamayseh, B. Abdulraheem, M. Wail, B. Y. Muneer, "A New Protocol for Detecting Black Hole Nodes in Ad Hoc Networks," International Journal of Communication Networks and Information Security, Vol. 3, No. 1, pp. 36-41, 2011.

[6] M. Zapata, N. Asokan, "Securing ad-hoc routing protocols," ACM Workshop on Wireless Security (WiSe), Atlanta, GA, pp. 1–10, 2002.

[7] Y. C. Hu, A. Perrig, D. B. Johnson, "Ariadne: a secure ondemand routing protocol for ad hoc networks," Wireless Networks, Vol. 11, Issue. 1-2, pp. 21–38, 2005.

[8] Y. C. Hu, D. B. Johnson, A. Perrig, "Sead: Secure efficient distance vector routing for mobile wireless adhoc networks," 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '02), pp. 3–13, 2002.

[9] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, E. Roye, "A secure routing protocol for adhoc networks," 10th IEEE International Conference on Network Protocols (ICNP-02), pp. 78–89, 2002.

[10] N. Iltaf, A. Ghafoor, U. Zia, "A mechanism for detecting dishonest recommendation in indirect trust computation," EURASIP Journal on Wireless Communications and Networking, Vol. 2013, No. 1, pp. 1-13, 2013.

[11] C. Dellarocas, "Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior," 2nd ACM Conference on Electronic Commerce, pp. 150–157, 2002.

[12] Z. Zhenjing, M. Maode, J. Zhigang, "TR-SDTN: Trust Based Efficient and Scalable Routing in Hostile Social DTNs," International Journal of Distributed Sensor Networks, Vol. 2015, Article ID 690482, 11 pages, 2015.

[13] S. Marti, T. Giuli, K. Lai, M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," 6th Annual International Conference on Mobile computing and networking, pp. 255-265, 2000.

[14] K. Paul, D. Westhoff, "Context-Aware Detection of Selfish Nodes in DSR based Ad Hoc Networks," IEEE Global Telecommunications Conference (GLOBECOM'02), Vol. 1, pp. 178-182, 2002.

[15] J. H. Cho, A. Swami, I. R. Chen, "Modeling and analysis of trust management with trust chain optimization in mobile ad hoc networks," Journal of Networks and Computer Applications, Vol. 35, Issue. 3, pp. 1001-1012, 2012.

[16] S. Buchegger, J. Y. L. Boudec, "Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes- Fairness In DynamicAd-hoc NeTworks," 3rd ACM Int. Symposium on Mobile Ad Hoc Networking and Computing, pp. 226-236, 2002.

[17] K. Hoffman, D. Zage, C. Nita-Rotaru, "A survey of attack and defense techniques for reputation systems," ACM Computing Surveys, Vol. 42, Issue. 1, pp. 1–31, 2009.

[18] M. Virendra, M. Jadliwala, M. Chandrasekaran, S. Upadhyaya, "Quantifying trust in mobile ad-hoc networks," IEEE International Conference on Integration of Knowledge Intensive Multi-Agent Systems (KIMAS'05), 2005.

[19] R. Ferdous, M. Vallipuram, A. Sattar, "Trust formalization in mobile ad-hoc networks," IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, pp. 351 – 356, 2010.

[20] A .M. Abd El-Haleem, I. A. Ali, "TRIUMF: Trust-Based Routing Protocol with controlled degree of Selfishness for Securing MANET against Packet Dropping Attack," International Journal of Computer Science Issues (IJCSI), Vol. 8, Issue. 4, 2011.

[21] Z. Li, X. Li, V. Narasimhan, A. Nayak, I. Stojmenovic, "Autoregression Models for Trust Management in Wireless Ad Hoc Networks," IEEE Global Telecommunications Conference (GLOBECOM 2011), pp. 1-5, 2011.

[22] M. Sardar, K. Majumder, "A new trust based secure routing scheme in manet," International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA), Springer International Publishing, pp. 321–328, 2014.

[23] C. Qu, L. Ju, Z. Jia, H. Xu, L. Zheng, "Light-Weight Trust-Based On-Demand Multipath Routing Protocol for Mobile Ad Hoc Networks," 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (IEEE TrustCom-13), Melbourne, Australia, pp. 42 – 49, 2013.
[24] P. B. Velloso, R. P. Laufer, D. de O Cunha, O. C. M. B. Duarte, G. Pujolle, "Trust management in mobile ad hoc networks using a scalable maturity-based model," IEEE Transactions on Network and Service Management, Vol. 7, No.3, pp. 172-185, 2010.
[25] S. R. Zakhary, M. Radenkovic, "Reputation-based security protocol for manets in highly mobile disconnection-prone environments," Seventh International Conference on Wireless On-demand Network Systems and Services, pp. 161-167, 2010.
[26] B. Qureshi, M. Geyong, D. Kouvatsos, "A distributed reputation and trust management scheme for mobile peer-to-peer networks," Computer Communications, Vol. 35, Issue. 5, pp. 608-618, 2012.
[27] H. Xia, Z. Jia, L. Ju, X. Li, E. H.M. Sha, "Impact of trust model on on-demand multi-path routing in mobile ad hoc networks," Computer Communications, Vol. 36, Issue 9, pp. 1078-1093, 2013.
[28] I. R. Chen, J. Guo, F. Bao, J. H. Cho, "Trust management in mobile ad hoc networks for bias minimization and application performance maximization," Ad Hoc Networks, Vol. 19, pp. 59-74, 2014.
[29] S. Chen, Y. Zhang, Q. Liu, J. Feng, "Dealing with dishonest recommendation: The trials in reputation management court," Ad Hoc Networks, Vol. 10, Issue 8, pp. 1603-1618, 2012.
[30] B. Yang, R. Yamamoto, Y. Tanaka, "Historical evidence based trust management strategy against black hole attacks in MANET," 14th International Conference on Advanced Communication Technology pp. 394-399, 2012.
[31] R. Feng, C. Shenyun, W. Xiao, N. Yu, "A credible routing based on a novel trust mechanism in ad hoc networks," International Journal of Distributed Sensor Networks, 2013.
[32] Zakirullah, M. H. Islam, A. A. Khan, "Detection of dishonest trust recommendations in mobile ad hoc networks," 5th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1-7, 2014.
[33] A. Arning, R. Agrawal, P. Raghavan, "A linear method for deviation detection in large databases," Data Mining and Knowledge Discovery, Portland, Oregon, pp. 164-169, 1996.
[34] http://www.isi.edu/nsnam/ns/, accessed 12 July, 2011.
[35] C. Zouridaki, B. Mark, M. Hejmo, R. Thomas, "Hermes: a quantitative trust establishment framework for reliable data packet delivery in MANETs," Journal of Computer Security, Vol. 15, Issue. 1, pp. 3–38.
[36] X. Wei, J. Fan, M. Chen, T. Ahmed, A. S. K. Pathan, "SMART: A subspace based malicious peers detection algorithm for P2P systems," International Journal of Communication Networks and Information Security, Vol. 5, Issue. 1, pp. 1-9, 2013.
[37] W. Zhexiong, H. Tang, F. R. Yu, W. Maoyu, P. Mason, "Security Enhancements for Mobile Ad Hoc Networks With Trust Management Using Uncertain Reasoning," IEEE Transaction on Vehicular Technology, Vol. 63, Issue. 9, pp. 4647-4658, 2014.