Reasoning about user trustworthiness with non-binary advice from peers

J. Finnson[1], R. Cohen[1] J. Zhang[2], T. Tran[3], U.F. Minhas[1],

U. Waterloo, Canada[1]; Nanyang Tech. U., Singapore[2]; U. Ottawa, Canada[3]

Abstract. In this paper we outline an approach for reasoning about the trustworthiness of users when advice from peers is provided and a majority opinion is central to the overall calculation. Typically, trust modeling frameworks have reasoned with binary reports from peers (i.e. trustworthy or not). In this paper, we illustrate how to do trust modeling when numeric advice is provided, instead (i.e. degree of trustworthiness within a continuum). This is done in the context of a specific application: directing the travel decisions of users with peer-provided traffic reports. We demonstrate the effectiveness of our solution by mapping the average time for completing paths in simulated traffic environments when vehicles are modeling trust using our framework and road reports are shared. We conclude with a discussion of the value of our approach for other applications as well.

1 Introduction

Many current trust modeling researchers choose to represent the trustworthiness of a user (or agent) as a binary value. With each new experience, the user is determined to be trustworthy or untrustworthy, resulting in an increase or decrease in the overall reputation of that user (typically on a scale of 0 to 1). At times, it is the experiences of peers that help to determine the calculation that is done (rather than the direct experience of the party that is modeling the trust). The trustworthiness of these peers then also needs to be modeled.

In this research, we consider scenarios where users' trustworthiness is modeled on the basis of reports they have provided and where it would be valuable to determine whether there is a consensus among the views of various peers about another user's reputability. While it is relatively straightforward to determine consensus when the possible evaluations are simply binary values (yes or no, when asked whether another party is trustworthy), it becomes more challenging to do so when evaluations can assume a range of possible values on a continuum (a level of trustworthiness, in a range of 0 to 1). Since more specific evaluations are in fact more precise and valuable for the trust modeling, it is important to support these more complex calculations.

In this paper, we first present a trust model that includes a majority consensus calculation but that assumes only binary reports, one that is used in the context of vehicle travel planning (within mobile vehicular ad-hoc networks

(VANETS) communicating traffic reports). We then propose an extension of this trust modeling to support the exchange of numeric evaluations. For this, we sketch the algorithms required to employ the trust modeling as part of travel planning together with the formulae to introduce in order to achieve the proper calculation of majority consensus.

We then include a few results from a simulation of travel planning (using a metric of average path time to demonstrate effective solutions), to confirm that our proposed algorithms for reasoning about trustworthiness are able to cope well with reports presented from users with varying honesty: modeling trustworthiness, relying on more reputable sources and achieving effective travel decisions. This serves as a validation of our proposed model for reasoning about numeric trust values. We conclude with a brief discussion of the potential value of this framework for applications other than VANETs and a comparison to related work.

2 Multi-faceted Trust Model

We consider the driver of each vehicle in our VANET environment to be a user. In order for each vehicle on the road to make effective traffic decisions, information is sought from other vehicles (about the traffic congestion on a particular road). As a result, for each driver an intelligent agent constructs and maintains a user model for each of the other vehicles. Travel decisions are then made based on a multi-faceted model of user trustworthiness. In particular, we propose a core processing algorithm to be used by each user that seeks advice from other vehicles in the environment as summarized below.

Algorithm 1: Computation Steps

```
while on the road do
    Send requests and receive responses;
    if in need of advice then
        Choose n; //number of users to ask for advice
        //according to roles and experiences
        Prioritize n users;
        if response consensus > acceptable ratio then
            Follow advice in response;
        else
            Follow advice of user with highest role and highest trust value;
    Verify reliability of advice;
    Update users' trust values;
```

Experience-based trustworthiness is represented and maintained following the model of [6] where $T_A(B) \in (-1, 1)$ represents A's trust in B (with -1 for total distrust and 1 for total trust) which is incremented by α if B's advice is found to be reliable or decremented by β if unreliable, with $|\beta| > |\alpha|$ to reflect

that trust is harder to build up but easier to tear down. Distinct from the original model of [6], the values of α and β can be set to be event-specific. The equations which adjust experience-based trust are as below:

$$T_A(B) \leftarrow \begin{cases} T_A(B) + \alpha(1 - T_A(B)) \text{ if } T_A(B) \geq 0, \\ T_A(B) + \alpha(1 + T_A(B)) \text{ if } T_A(B) < 0, \end{cases} \tag{1}$$

$$T_A(B) \leftarrow \begin{cases} T_A(B) + \beta(1 - T_A(B)) \text{ if } T_A(B) \geq 0, \\ T_A(B) + \beta(1 + T_A(B)) \text{ if } T_A(B) < 0, \end{cases} \tag{2}$$

For each user B_i $(1 \leq i \leq n)$ belonging to a subset of users $\mathcal{B}(R_j) \subseteq \mathcal{B}$ who report the same report $R_j \in \mathcal{R}$ $(1 \leq j \leq m)$, we aggregate the effect of its report according to the above factors. The aggregated effect $E(R_j)$ from reports sent by users in $\mathcal{B}(R_j)$ can be formulated as follows (per [2]):

$$E(R_j) = \sum_{B_i \in \mathcal{B}(R_j)} \frac{T_e(B_i) T_r(B_i)}{C_t(R_j) C_l(B_i) W(B_i)} \tag{3}$$

We define C_t (time closeness), C_l (location closeness), T_e (experience-based trust) and T_r (role-based trust). Note that all these parameters belong to the interval $(0, 1)$ except that T_e needs to be scaled to fit within this interval. $W(B_i)$ is a weight factor set to 1 if user B_i who sent report R_j is an indirect witness, and $W(B_i)$ is set to a value in $(0, 1)$ if user B_i is a direct witness[1].

A majority consensus can be reached if

$$\frac{M(R_j)}{\sum_{R_j \in \mathcal{R}} E(R_j)} \geq 1 - \varepsilon \tag{4}$$

where $\varepsilon \in (0, 1)$ is set by user A to represent the maximum error rate that A can accept and $M(R_j) = \max_{R_j \in \mathcal{R}} E(R_j)$. A majority consensus can be reached if the percentage of the opinion (the effect among different reports) over all possible opinions is above the threshold set by user A.

3 Our proposed numeric trust modeling

Our new proposed confidence metric and use of numeric congestion and trust values serve to allow a more accurate description of traffic and agent information, which will be explained below.

The original theory in Section 2 assumed that congestion would be communicated as a simple *true* (Heavy) or $false$ (Light), stating either that the road was congested or not. However, direct application may result in an unfair and biased calculation of the majority opinion. This is because determining whether a road is congested or not is a subjective opinion and is prone to inaccuracies. Also, by representing the congestion as a Boolean, this severely limits the system's ability

[1] For example, setting $W(B_i) = 1/2$ for the case of direct witnesses indicates that the requesting user values direct evidence two times more than indirect evidence.

to compare roads, evaluate agents, and make the best decisions. Our proposed model seeks to alleviate this problem by representing congestion as a number , which will bring a more suitable level of accuracy to the system [2].

Formula (3) shows the calculation for the aggregated effect of a majority opinion. The new way of representing congestion as a numeric value requires a careful recasting of formula (3). (3) aggregates the effect of all agents that sent the same report (i.e. cong = true). This simple aggregation of similar reports is impossible with the new congestion representation because there are no longer only two types of reports (Cong=true or Cong=false). In the new framework, each report must be evaluated for addition into the majority opinion system. This is done by giving the report a confidence and then evaluating it for inclusion into the majority opinion (similar to the aggregated effect calculation). The following sections will detail how the factors of experience and role based trust, time and location closeness, and whether the advice is direct or indirect are incorporated into our proposed confidence metric and utilized in calculating a majority opinion.

Confidence Calculation Confidence functions as a metric similar to trust, and is calculated by combining many different report and agent factors, which were introduced in Formula 3 and will be described in detail later in this section. These factors include experience and role based trust, time and location closeness, and whether the advice is direct or indirect.

Our proposed Equation for calculating confidence must effectively replace Formula 3, while representing a trust-like metric. Modifications to confidence should then be reflected in a manner similar to how trust is increased and decreased in Equations 1 and 2. α and β function in these Equations as a standard for increasing and decreasing trust, respectively. For our proposed confidence calculation it did not make sense to atomically increase or decrease the value according to the influencing factor (role, time closeness, etc.). The increase or decrease should reflect the significance of the factor. As a result, our proposed confidence metric replaces Formula 3 with Equation 5, where Equations 1 and 2 are used as the basis for calculating the confidence of report R_j, through a modified summation of a geometric series [3].

The factors of role based trust, time and location closeness, and whether the advice is direct or indirect in formula (3), are reflected through Variable (G). Experience based trust of an agent automatically forms the default value

[2] Note that a reported congestion value for instance of 23 would ideally be representing the actual number of cars on the road; in reality a car would be more likely to provide a value on a fixed scale to represent whether the traffic it observed were very heavy, moderate or light, for example. It may also be reasonable for cars to report their speed and for this to be used as a reflection of the road's congestion. In so doing, this deflects issues of number of cars relative to length of the road, for example.

[3] A Geometric series is necessary because the calculations are capturing atomic increases in trust values but we are reasoning about non-Boolean factors that are therefore not atomic.

of the confidence metric ($CurrConf(R_j)$). Variable (G) represents the number of times[4] to increase or decrease confidence. G's calculation is specific to each factor. If G is calculated to a negative value, this indicates that β should be used instead of α. An example is shown in Section 3.2. The following sections briefly detail how each factor influences G; however the exact calculations are dependent on how parameter values are chosen, within an implementation.

$$Conf(R_j) = (CurrConf(R_j) - 1)(1 - (\alpha \ or \ \beta))^{|G|} + 1 \tag{5}$$

3.1 Majority Calculation

Algorithm 2 is a modified algorithm from Algorithm 1, which shows the calculation of a majority opinion in the framework. The algorithm uses suspicious agent detection in helping to avoid the inclusion of congestion advice which is outside a standard deviation from the current majority congestion. The majority opinion is used if there are at least n agents to use advice from and the majority confidence is above the majority threshold.

Algorithm 2: New Majority Computation Steps, with Numerical Congestion Metric

```
while on the road do
    Send requests and receive responses;
    if in need of advice then
        Choose n reports R; //number of reports to use for advice
        Check Priority Road(Current Road);//to help update the Priority list
        Prioritize n reports; //according to Confidence (roles, experiences, time,
        location, and if report is indirect or direct)
        foreach n reports do
            if R_j suspicious then
                Report suspicious agent R_j;
            else
                Include report R_j in Majority;
            if Majority suspicious then
                Decrease Majority confidence;
        if Majority confidence > acceptable threshold && Number of reports >
        n threshold then
            Follow advice in response;
        else
            Follow advice of report with highest confidence;
    Verify reliability of reports;
    Update users' trust values;
```

[4] Note that we use the absolute value of G as the exponent in order to ensure that the number of times is a positive number.

Algorithm 2 clarifies whether an agent will choose to take a certain road or not based on consensus about the congestion on the road. If the agent wants to reason about which road to choose (from a set of possible roads), it can run Algorithm 2 for each road [5].

Suspicion detection is important to include to help avoid congestion advice that greatly deviated from the current majority. Only using advice that has similar congestion reports forms our majority opinion, rather than conceiving of majority opinion as just an average congestion of the highest trusted agents (n). If an agent is deemed suspicious, then they are reported and the agent's advice is not used in the majority opinion calculation. However, the reverse is possible where if an agent's advice has higher confidence than the majority and confidence greatly deviates from the majority. If this happens then the majority confidence is decreased proportionally and the agent's advice is potentially used as the *report with highest confidence*.

Experience based trust is the most basic type of trust and is applied to every agent in our model framework. This is the initial confidence value we begin with.

Role based trust is incorporated into a proposition's confidence calculation by increasing it by a magnitude proportional to the particular role's rank. Equation 6 shows how G is calculated for Equation 5. $RPenal$ is a standard value for weighting roles, and $RoleRank$ is the rank of the roles. G is inversely proportional to $RoleRank$ so that higher roles ($Authority$ has $RoleRank$ of 2) warrant greater increases in confidence.

$$G = RPenal/RoleRank \tag{6}$$

Time and location closeness helps alleviate the issue of old and inaccurate reports. Equations 7 and 8 show how G is calculated for Equation 5. $TPenal$ and $LPenal$ are standard values for weighting time and location respectfully. *TimeDifference* and *LocDifference* are time difference and location difference respectively. *MultiplicativeFactor* is a standard multiplicative factor for the calculation (max confidence increase will be *MultiplicativeFactor*, and not 1, if *TimeDifference* or *LocDifference* is 0.). The calculation finds the difference between *TimeDifference/LocDifference* and *TPenal/LPenal* and then divides the difference by *TPenal/LPenal*. This achieves the purpose of scaling the values to be within their unit metrics.[6]

$$G = (TPenal - TimeDifference)/TPenal * MultiplicativeFactor \tag{7}$$
$$G = (LPenal - LocDifference)/LPenal * MultiplicativeFactor \tag{8}$$

Direct reports are reports which have been directly observed and reported by an agent. Indirect reports are direct reports of a third agent which are stored in the knowledge base of the agent the resident agent is communicating with.

[5] Note that this is in fact what we do in our implementation in Section 4.

[6] This required scaling was not considered in sufficient detail in the model of Minhas et al. and Equation 3.

Equation 9 shows how G is calculated for Equation 5. $InPenal$ is a standard value for penalizing indirect reports, and *IfIndirect* is 1 if the report is indirect and 0 otherwise.

$$G = InPenal * IfIndirect \tag{9}$$

3.2 Confidence Calculation Example

The following calculation demonstrates how a confidence value can be calculated.

Example 1: (illustrating α)

Confidence = *Agent_39:trust_degree* (0.6)

G_{time} = (TPenal(90)-TimeDiff(18))/TPenal(90) *MultiplicativeFactor(1.5)
G_{time} = 1.2

Confidence(0.6) = (Confidence(0.6)-1)$(1-\alpha)^{|G_{time}|}$+1
Confidence = 0.6475

G_{loc} = (LPenal(200)-LocDiff(100))/LPenal(200) *MultiplicativeFactor(1.5)
G_{loc} = 0.75

Confidence(0.6475) = (Confidence(0.6475)-1)$(1-\alpha)^{|G_{loc}|}$+1
Confidence = 0.674

Example 2: (illustrating β)

Confidence = *Agent_41:trust_degree* (0.7)

G_{role} = RPenal(8)/RoleRank(2)
G_{role} = 4

Confidence(0.7) = (Confidence(0.7)-1)$(1-\alpha)^{|G_{role}|}$+1
Confidence = 0.8032

G_{time} = (TPenal(90)-TimeDiff(180))/TPenal(90) *MultiplicativeFactor(1.5)
G_{time} = -1.5

Confidence(0.7813) = (Confidence(0.7813)-1)$(1-\beta)^{|G_{time}|}$+1
Confidence = 0.7413

G_{loc} = (LPenal(200)-LocDiff(500))/LPenal(200) *MultiplicativeFactor(1.5)

G_{loc} = -2.25

Confidence(0.7604) = (Confidence(0.7604)-1)$(1-\beta)^{|G_{loc}|+1}$
Confidence = 0.6100

$G_{indirect}$ = InPenal(-2)*IfIndirect(1)
$G_{indirect}$ = -2

Confidence(0.6991) = (Confidence(0.6991)-1)$(1-\beta)^{|G_{indirect}|+1}$
Confidence = 0.4385

4 Simulation

This section describes the simulation tests performed to compare and contrast the effectiveness of our model's implementation against a system that does not use traffic information in routing; a best case scenario; the inclusion of time, location, and indirect advice.

The implementation makes use of the following third party software, JiST/SWANS, vans, DUCKS, and Protege. JiST stands for Java in Simulation Time; it is a high-performance discrete event simulation engine that runs over a standard Java Virtual Machine (JVM). SWANS stands for Scalable Ad-hoc Network Simulator; it is built on top of the JiST platform and serves as a host of network simulation tools. Vans is a project comprising the geographic routing and the integrated Street Random Waypoint model (STRAW). STRAW utilizes an A* search algorithm to calculate shortest path to a destination. DUCKS is a simulation execution framework, which allows for a Simulation Parameters file to be provided to define the simulation. Protege is a free, open source ontology editor and knowledge base framework.

The simulation was set to poll cars every 6-15 seconds; with 100 cars in total, experience with every other car would be gained quickly. In order to simulate environments with low experience-based trust, we introduce a variable called sparsity. For example, 80% sparsity resembles having a lack of previous experience with 80% of the agents. In the simulation, this variable effectively ignores updates of trust values, thus hindering experience-based trust.

These graphs chart the performance of simulations that either use trust modeling (i.e. profiling) (Hon #) or not (no P, Hon #)#)[7]. Agent honesty represents the percent of honest agents in the simulation (i.e. 0.5 is 50% honesty). By default, trust modeling uses at least experience and majority based trust. These are the central elements of the original model of Minhas et al. [2, 3]. In the simulations displayed below, we map an initial version of the implementation with only these two trust elements included and refer to this as Basic.

Role-based trust represents the percent of agents in the simulation that have been assigned a role and this is the default value that we used in the simulations

[7] With no profiling, agents do not model the trustworthiness of the traffic reports received and assume that they are truthful.

displayed below (i.e. 0.2 will have 20% of agents assigned a role). Sparsity represents the percent sparsity in the simulation (i.e. 0.8 will have 80% sparsity and this is the default value that we used in the simulations displayed below). The other trust model components individually indicated are time closeness (Time), location closeness (Loc), and indirect advice (Indir). (Full) indicates when all multidimensional trust components are being used.

The VANET trust modeling results are also compared against two additional simulations: the first is a worst case scenario where traffic is ignored (no traffic)[8], and the other is a best case omnipresent version (omni) which simulates the ability for any car to look up the exact congestion of any road at anytime. All simulation tests results are averaged over 5 runs.

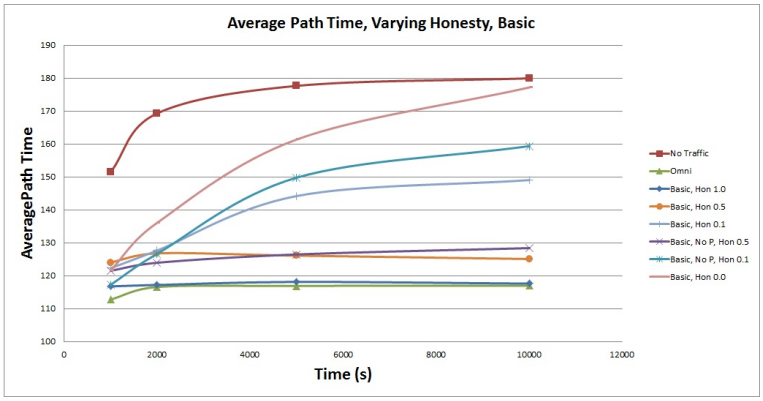

Fig. 1. Avg Path Time comparison of our model vs. best and worst case scenarios

Figure 1 examines a metric referred to as average path time (appropriate due to the ultimate goal of reducing the travel time of users). This figure compares the worst case scenario against the best case scenario and various simulations which use our VANET system, at different degrees of honesty. Greater average path time in the figure indicates lower performance. As seen in the figure, the simulations that used our trust modeling framework (except *Basic, Hon 0.1*) and the *Omni* setup averaged close to the same path time at the end of the 10000 second simulation. The other simulations produced a predictably declining performance as the honesty percentage approached the worst case scenario. The *Basic, Hon 0.1* simulation did much worse than the other *Basic* simulations most likely due to the extreme lack of trustworthy agents, but it still performed significantly better than the *Basic, No P, Hon 0.1* simulation. The VANET trust modeling simulations show approximately a 35% decrease in average path time over the worst case scenario. The curves in the scenarios are representative of the simulations approaching a steady state. Another observed trend is the tendency

[8] Routing without traffic just uses a shortest path calculation.

for the profiling-enabled simulations to reach a steady state faster than the other simulations.

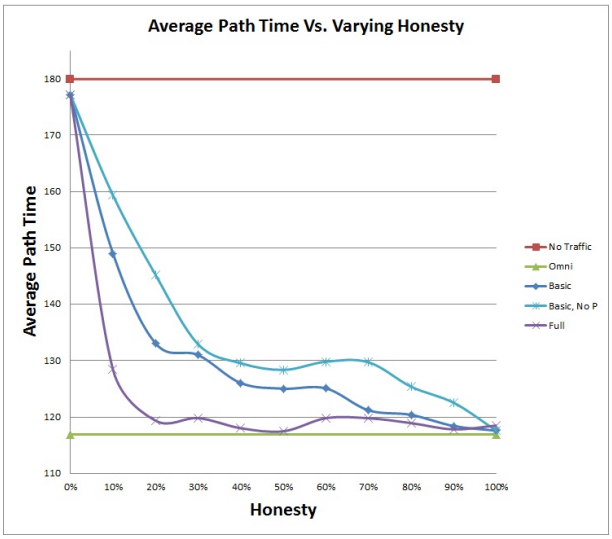

Fig. 2. Avg Path Time comparison of simulation types over varying degrees of honesty at 10,000 seconds

Figure 2 compares the average path time, at 10,000 seconds, of the *No Traffic*, *Omni*, *Basic*, *Basic, No P*, and *Full* scenarios, across a range of honesty values. *No Traffic* and *Omni* are shown as straight lines because they do not use honesty values, but are useful as comparisons. The figure clearly shows the effectiveness of our framework across the range of honesty values. The *Basic* scenario consistently performs better than the *Basic, No P* scenario. The *Full* scenario also consistently performs better than the *Basic* scenario. All of the framework enabled simulations have a similar average path time at 0% honesty because they have no useful traffic data (and at 100% honesty because there are no untrustworthy agents to deflect through profiling). Figure 2 clearly demonstrates the impact dishonest agents can have on simulations (*Basic, No P*) and the effectiveness our proposed model framework scenarios (*Basic* and *Full*) can have on countering the influence of dishonest agents.

Figure 3 compares the average path time, at 10,000 seconds, of the *No Traffic*, *Omni*, *Basic*, *Basic, No P*, and *Full* scenarios, across a range of values for the number of agents in the environment. The figure clearly shows the robustness of our framework across the span of agent values. The simulations around 50 agents have approximately the same path time because with such a small number of cars there is no real need for using traffic information in path planning. When increasing the number of agents, the *Basic* scenario consistently performs better than the *Basic, No P* scenario. The *Full* scenario also consistently performs

better than the *Basic* scenario, when there are more than 50 agents. Figure 3 clearly demonstrates the robustness and scalability of our proposed model framework and implementation across a range of values for the number of agents in the environment.

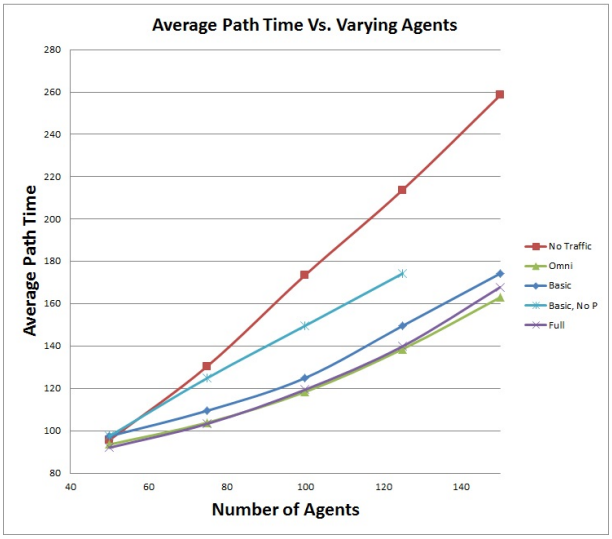

Fig. 3. Avg Path Time comparison, varying number of agents

5 Discussion

The framework presented in this paper required a calculation of majority consensus in order to guide the decision making of a user. Other researchers have integrated majority opinion into their trust modeling but have instead used this calculation to reflect the general reputation of an agent (e.g. just how trustworthy a user is may be represented as a numeric value calculated as the average of all the scores provided by peers (say 1 for trustworthy and 0 for untrustworthy). For instance, Zhang [9] has calculations that integrate a public reputation into the trustworthiness calculation and that also weight the contributions provided by peers according to the estimated trustworthiness of each of the advisors. Work with Singh, with various co-authors [8, 7] and that of Gorner [1] outlines the use of a social network of advice as well, for trust calculations. While reputation becomes a non-numeric value, the input to this calculation are always binary reports. This is true as well of the models TRAVOS [5] and BLADE [4]. Our approach, in contrast, illustrates how to support advice that is numeric, instead.

While we have sketched our proposed formulae and their validation in the context of a specific VANET application, the approach is applicable to any scenario where experience-based trust and majority consensus are to be integrated

into the overall determination of user trustworthiness. The formulae in use would simply omit the undesired elements of Equation 5: for instance, time and location may be irrelevant. The remaining calculations would remain the same.

In conclusion, we offer an approach for supporting reasoning about agent trust with advice from peers, whose trustworthiness is then also modeled, when non-numeric reports are provided and have shown the merit of our framework in the context of the VANET application (resulting in effective travel decisions due to the modeling of trustworthiness). As such, we offer a method that supports the exchange of more detailed trustworthiness information, leading to more precise and valuable calculations. Future work includes the exploration of a variety of additional applications and their trust modeling needs, towards refinement and expansion of the approach. Additional research will explore in greater depth the related elements of reputation and majority consensus, to determine to what extent both may be integrated into the trust modeling.

References

1. Gorner, J., Zhang, J., Cohen, R.: Improving the use of advisor networks for multi-agent trust modelling. In: PST. pp. 71–78 (2011)
2. Minhas, U.F., Zhang, J., Tran, T.T., Cohen, R.: Intelligent agents in mobile vehicular ad-hoc networks: Leveraging trust modeling based on direct experience with incentives for honesty. In: Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT) (2010)
3. Minhas, U.F., Zhang, J., Tran, T.T., Cohen, R.: A multifaceted approach to modeling agent trust for effective communication in the application of mobile ad hoc vehicular networks. IEEE Transactions on Systems, Man, and Cybernetics, Part C 41(3), 407–420 (2011)
4. Regan, K., Poupart, P., Cohen, R.: Bayesian reputation modeling in e-marketplaces sensitive to subjectivity, deception and change. In: AAAI. pp. 1206–1212 (2006)
5. Teacy, W.T.L., Patel, J., Jennings, N.R., Luck, M.: Travos: Trust and reputation in the context of inaccurate information sources. Journal of Autonomous Agents and Multi-Agent Systems 12, 2006 (2006)
6. Tran, T., Cohen, R.: Modelling reputation in agent-based marketplaces to improve the performance of buying agents. In: Proceedings of the Ninth International Conference on User Modelling (UM) (2003)
7. Yolum, P., Singh, M.P.: Engineering self-organizing referral networks for trustworthy service selection. IEEE Transactions on Systems, Man, and Cybernetics, Part A 35(3), 396–407 (2005)
8. Yu, B., Singh, M.P.: Detecting deception in reputation management. In: Proceedings of the second international joint conference on Autonomous agents and multiagent systems (2003)
9. Zhang, J., Cohen, R.: Evaluating the trustworthiness of advice about seller agents in e-marketplaces: A personalized approach. Electronic Commerce Research and Applications 7(3), 330–340 (2008)