Trust Dynamics: A Data-Driven Simulation Approach

Olufunmilola Onolaja, Rami Bahsoon, Georgios Theodoropoulos

Trust Dynamics: A Data-Driven Simulation Approach

Olufunmilola Onolaja, Rami Bahsoon, and Georgios Theodoropoulos

The School of Computer Science, University of Birmingham, UK
o.o.onolaja@cs.bham.ac.uk, r.bahsoon@cs.bham.ac.uk,
g.k.theodoropoulos@cs.bham.ac.uk

Abstract. Reputation and trust-based models have gained popularity recently because they have been shown to be promising in the area of trust management. Despite this fact, building reliable systems still remains a challenge. Proposed models focus on historical and online information to determine the reputation of domain members. However, the dynamic nature of reputation and trust requires an equally dynamic approach to computing and resolving trust related issues in any domain. This paper proposes a reliable and novel dynamic framework that utilises a data-driven approach for trust management. The framework uses past interactions, recent and anticipated future trust values of every identity in the domain. The proposed framework is critically evaluated and compared with existing work through experiments. The advantage of this proactive framework compared to other approaches is that informed decisions about the domain can be made before misbehaviour occurs.

Keywords: trust dynamics, trust management, reputation

1 Introduction

In a social context, when a person is *trusted*, it implicitly means that the probability that the person will perform an action that is beneficial or at least not detrimental in the society, is high enough to consider engaging in some form of cooperation with the individual [5]. *Reputation*, on the other hand, is the opinion of one person about another; it is a measure of the trustworthiness of a person. Both trust and reputation have been used synonymously in literature.

Behavioural expectation in any domain can be motivated from a social perspective, where individuals are expected to behave in certain ways within the society. The behaviour of an individual, whether good or bad, will determine how others will cooperate with the individual. The expected behaviour of a sensor for example, in a Wireless Sensor Network (WSN) set up for monitoring, is to be cooperative in collecting and processing observed data with neighbouring sensors. *Misbehaviour* is the deviation from the expected behaviour in the domain and entities that misbehave are said to be untrusted.

Reputation and Trust-based Models (RTMs) [1, 3, 4, 6, 7, 17] are described as systems that provide mechanisms to produce a metric encapsulating reputation

for a given domain for each identity in the domain . This is referred to as *Trust Value* (TV) or trust ratings in this paper. Generally, RTMs aim to provide information to distinguish between trustworthy and untrustworthy members. The models encourage members to cooperate by providing incentives and discourage maliciousness by punishment schemes such as isolation and service denial. RTMs have been used extensively in various e-commerce and online communities such as YouTube, Amazon and eBay. Some literatures also suggest their use in domains ranging from Peer-to-Peer (P2P) to mobile networks [3, 6, 8, 17].

Traditional RTMs rely on recommendations provided by entities in the domain to determine the reputation of others. Each of the models addresses some of the trust issues but not all of the problems, or in the process of solving one issue they introduce others. An example of the problem that arises from the reliance on these recommendations is *collusion*, where two or more entities team up to behave maliciously. Without countermeasures, the effects of this attack have shown to dramatically affect the network performance as evidenced in poor reliability and quality of service, higher overhead and throughput degradation [3]. Incentive policies that are used in P2P networks to ensure cooperation between peers are also generally susceptible to collusion attack [14].

Generally, RTMs make use of past events as a pointer for the future. However, for an RTM to be reliable and effective in trust management, trust has to be predictable. It is generally assumed that the predictive power of an RTM depends on the supposition that past behaviour is an indication of future behaviour [13]. This assumption might not be true with another malicious behaviour called *intoxication*. Intoxication occurs because the effect of past good behaviour outweighs the effect of current misbehaviour. Therefore, we argue that using historic (or past) interactions as the only basis for predicting the future TVs of identities in a domain is inadequate to provide a trusted system. Our framework extends the supposition further by not only considering past interactions but also anticipating possible future behaviour of members.

In previous papers [18, 19], we described how trust decisions can be corrupted through recommendations made by members. We proposed a framework that is capable of providing dynamic trust ratings of members at runtime and predicting the future trust ratings. The framework does not rely on collective opinion and recommendations to determine the reputation of members. Instead, the framework predicts a potential compromise before it occurs. In this paper we present an extension to our original design, which uses predictions of future behaviour to determine trust ratings. We also present experiments comparing the results obtained with and without the use of prediction capabilities, confirming that the framework can provide more reliable predictions.

The rest of this paper is organised as follows: Section 2 describes significant RTMs in literature. The motivation for the use of dynamic data-driven paradigm in this research is discussed in Sect. 3 while Sect. 4 details the components of the framework. Section 5 presents a set of experimental results and analysis that shows that the predictive capability of our framework. Finally, we discuss and conclude in Sect. 6 and Sect. 7 respectively.

2 Related Work

Researchers proposed RTMs to solve trust related issues and the models have shown positive results. Some models that have contributed significantly to trust management in literature are discussed in this section.

Michiardi and Molva [17] proposed a model where reputation is formed and updated over time by direct observations and information provided by other members of the network. In their model, nodes have to contribute continuously to remain trusted or their reputation will be degraded until they are excluded from the network. The model gives a higher weight to past behaviour. The authors argue that a more recent sporadic misbehaviour should have minimal influence on a node's reputation that has been built over a long period of time.

A file-sharing P2P reputation system's algorithm: EigenTrust [10], similar to the popular PageRank aims to identify sources of inauthentic file and to prevent peers downloading from them. The algorithm assigns each peer a unique global TV, based on the peer's history of uploads. EigenTrust's susceptibility to collusion has been demonstrated in [14], where certain colluding peers are able to obtain high TVs.

Buchegger *et al.* [3] proposed a protocol that aims to detect and isolate misbehaving nodes, making it unattractive for any node to deny cooperation with others. In the protocol, each node maintains a reputation and a trust rating about every other node of interest. Only fresh reputation is propagated in the network, with more weight given to the current behaviour of a node over its past behaviour. Nodes monitor and detect misbehaviour in their neighbourhood by means of an enhanced *packet acknowledgment* mechanism; where the confirmation of acknowledgment comes indirectly by overhearing the next node forward the packet [2, 20].

In the work of Ganeriwal *et al.* [6]; which is applicable to WSNs, each sensor node maintains reputation metrics. These metrics represent the past behaviour of other nodes and are used as an inherent aspect in predicting their future behaviour. The model relies on network members to maintain the reputation of others based on their experiences and uses this to evaluate their trustworthiness.

More recent studies on RTMs are discussed in [1, 4, 7]. A common problem seen in the models is the vulnerability to collusion attacks [9]. Models applicable in the mobile networks domain, make use of a component resident on each node called *watchdog* mechanism. This component monitors its neighbourhood and gathers data by *promiscuous observation*. By promiscuous observation we mean that each node overhears the transmission of neighbours to detect misbehaviour. Watchdog requires that every node report to the originator about the next node. Once misbehaviour is detected, a negative TV is stored. This detection mechanism also has a weakness of failing to detect a misbehaving device in case of collusions [16].

Let us consider a set of sensor nodes that are deployed along the roadside to monitor vehicular movement in order to obtain real traffic flow data and conditions. The sensors are equipped with wireless interfaces with which they form a network. Nodes collaborate to collect and process data that generate

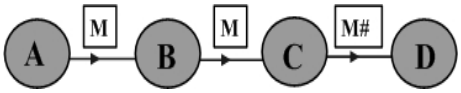

Fig. 1. Sensor node can misbehave by colluding to deceive the network

information about traffic conditions. When a sensor node receives information from another, this is combined and fused with local information before being sent to a server to control traffic. Figure 1 depicts collusion attack showing a downside of the watchdog mechanism. Knowing that WSNs are vulnerable to attacks due to their nature, an adversary compromises a sensor node, which in turn compromises other nodes. Consider a normal situation, where for example, sensor node A forwards a message to node B and B forwards the message to C. Node C then forwards the message to node D. However, node C may decide to alter the message before sending it to D. With the watchdog mechanism, it is possible that B colludes with C and does not report to A when C alters message M, before forwarding the message. Misbehaving nodes do not only have the chance to collude but can also propagate false information. Therefore, trust decisions can be corrupted through recommendations made by such sensor nodes.

3 Why Dynamic Data-Driven Simulation?

A disreputable person could redeem himself through honest actions and a trusted person could become less reputable or untrustworthy if he misbehaves in a society. This analogy is applicable also in trust management and implies that trust can fluctuate over time, making it dynamic. This dynamic nature of trust therefore calls for an equally dynamic approach for identifying misbehaving members.

The missing element in traditional RTMs is the reliable prediction of future TVs of members to proactively prevent misbehaviour. The classification of members into different levels of risk is also an important missing element. This classification can potentially help the RTM to focus on members that are of high-risk in the domain. Hence, we propose an approach that

1. Predicts the future TVs using past events, recent events and possible future interactions
2. Provides information about members that are classified as high-risk
3. Prevents members' bias from influencing trust decisions
4. Provides dynamic TVs of domain members.

This fits within a more general emerging paradigm referred to as Dynamic Data-Driven Application Systems (DDDAS). The DDDAS approach is that of a symbiotic relationship between reality and simulations. The simulation is able to make predictions about how an entity would evolve and its future state. The predictions made can then influence how and where future data will be gathered from the system, in order to focus on areas of uncertainty [11].

DDDAS has been applied in the simulation of physical, artificial or social entities [12, 15]. The application of DDDAS for trust management provides dynamism in the detection of misbehaving members and prediction of future ratings. The data about behaviour of members is simulated to gain a better understanding and a more accurate prediction of the level of trust for each member.

4 Dynamic Data-Driven Framework

This section introduces the framework and gives a comparative analysis with pre-existing models using monitoring, simulation, dynamism, and prediction as criteria. Figure 2 illustrates the relationship between the framework components.

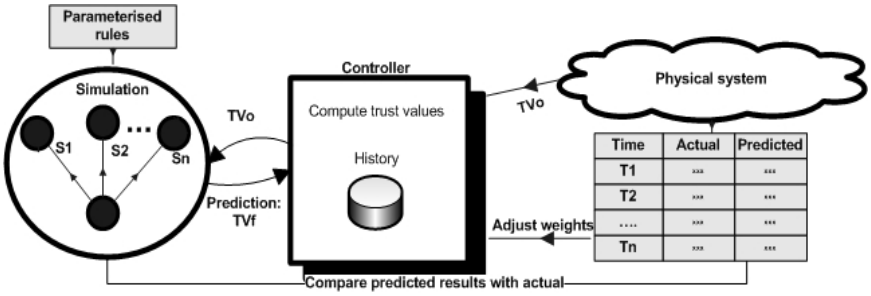

Fig. 2. Framework components showing how data is injected into the simulation and the scenarios $s_1, s_2, ..., s_n$

4.1 Trust Computation

Trust computation is very difficult, as trust has to be defined precisely. This is because the computation is crucial to the fulfilment of the functions in any trust-based framework. Computing trust in RTMs has been described as an abstract mathematical specification of how available information should be transformed into a usable metric [8]. In this framework, the specification is made through explicit equations.

A set of discrete TVs is assumed in the framework and each value represents a degree of trust [19]. These discrete degrees of trust introduce flexibility into applications of our framework, as different behaviours correspond to different levels of trust. Table 1 shows the trust table, the degrees of trust and corresponding level of risk in this framework.

Captured qualitative data is converted to a quantitative value. Data collected from the network (e.g. a P2P system, eBay, WSN etc) is transformed to a value ranging from 0 to 5, where a score of 0 means a node is completely untrusted, 5 means a node is absolutely trusted and if $0 < TV < 5$, then it implies that the node is trusted to a certain extent.

Using the notation tv^R, let the computed TV be

$$tv^R = \mu_h tv_h^R + \mu_o tv_o^R \tag{1}$$

where tv_h^R and tv_o^R are the average historical and recent online TVs respectively. Weights μ_h and μ_o are scaling factors of the TVs which can be varied and are introduced to allow for flexibility in the framework.

The simulation considers the possible scenarios a member may undertake in the future and the average of the ratings for the member determines the future tv_f^S. An predicted overall TV is computed as

$$TV = \mu_h tv_h^R + \mu_o tv_o^R + \mu_f tv_f^S \tag{2}$$

where μ_f is a scaling factor for the predicted value.

In the framework, recent behaviour has more weight than past interactions. This is to prevent nodes from attaining a good reputation and subsequently misbehaving (intoxication attack described in Sect. 1). The weights are used to control the effect of historical behaviour of nodes on their recent activities. For example, if (μ_o, μ_h) > 0 and $\mu_o > \mu_h$, this places more emphasis on recent behaviour as opposed to historical.

Table 1. Trust table showing the degrees of trust, meanings, descriptions and corresponding risk levels

TV	Meaning	Description	Risk Level
5	Complete trust	Trusted node with an excellent reputation	Low risk
4	Good trust level	Very reliable node	Low risk
3	Average trust level	Average value and somewhat reliable node	Medium risk
2	Average trust level	Average value but questionable node	Medium risk
1	Poor trust level	A questionable node	High risk
0	Complete distrust	Malicious node with a bad reputation	High risk

4.2 Simulation

In order for any RTM to fulfil its functions; observations, experiences and recommendations need to be captured and represented numerically. The simulation of the network runs concurrently with the real system itself. The aim of the simulation is to predict TV of members by using past interactions, current events and possible future scenarios. However, this component of the framework works ahead in time of the system. At specific time slots, the current state of the system is obtained and adapted to the simulation.

Data collected from the system are the online TV (tv_o^R) that represents the current rating of a member and the computed TV (tv^R) using the online ratings and past events. These values from reality are injected into the simulation at the

start. The simulation runs for more time steps and considers different what-if scenarios in which a member may be in the future.

Possible outcomes in the what-if scenarios are simulated to anticipate possible fluctuations in member behaviour. This is because the behaviour of members generally in any network, domain or context is dynamic and changes with time. Examples of possible scenarios that can be considered by the simulation are collusion attacks such as altering a message, intoxication and normal expected behaviour. The resulting TV for a member in each scenario is considered and with this information, it is possible to compute and anticipate the future TV of the member. In the controller (a trusted framework component depicted in Fig. 2), the data from the simulation is combined with online and historical TVs in order to obtain an overall TV.

After some specified time intervals $T_1, T_2, ..., T_n$, the simulation state is observed and compared with the actual state; this comparison is done automatically in the controller. The framework is adaptive such that if there are any differences in the predicted values and the reality, the weights for the trust computation can be continually adjusted to reflect reality. Each instance of the adjustment always ensures that the condition $\mu_o > \mu_h$ holds. This means that an entity's most recent action has more impact on its TV than past actions; consequently preventing intoxication. The exact way the adjustment may be achieved is beyond the scope of this paper.

Table 2 compares the extended framework with the RTMs described earlier based on the criteria of monitoring, simulation, dynamism and prediction.

Table 2. Summary table comparing existing RTMs with framework

Models	[17]	[10]	[3]	[6]	**Framework**
Monitoring	Watchdog mechanism	Peer recommendation	Watchdog mechanism	Watchdog mechanism	Controller monitoring
Simulation	n/a	n/a	n/a	n/a	Simulation of possible future states
Dynamism	Ratings are not constant	Periodic iterations to compute global TVs	Periodically updated	Provides real time feedback	Online ratings and control at intervals
Prediction	n/a	Past interactions serve as an indication of TVs	n/a	Trust metric that is representative of a nodes' future behaviour	Prediction of TVs using data from history, online and possible future behaviours

The framework performs better by predicting the future TVs of members. The prediction gives the network enough time for preventive measures, making the framework proactive compared to other models that are reactive. We refer to being proactive in terms of providing control such as downgrading of TV of suspect members that are predicted to be malicious before they can carry out

an attack. This is contrary to how other approaches work, that only downgrade the TV as a reaction to misbehaviour. The assumption is that a member that has been compromised by an adversary exhibits a sequence of behaviour in order to misbehave. A hypothetical example is depicted in Fig. 3a and Fig. 3b which show the time difference in response time between the framework and other approaches. Figure 3a shows that the TV is only downgraded at time t_5 after the member exhibits maliciousness. The simulation in the framework predicts the maliciousness between time interval t_1 and t_2 and the TV is downgraded at time t_3 in Fig. 3b.

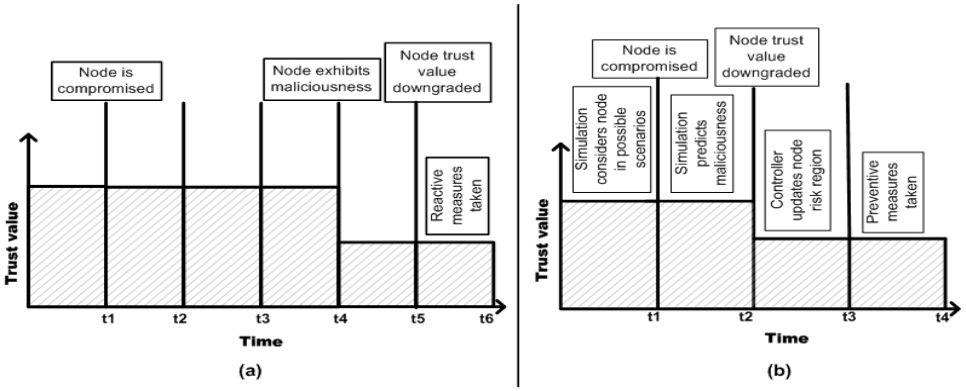

Fig. 3. Other approaches in (a) compared with the dynamic approach in (b)

5 Experiments

This section describes the simulation environment setup using Repast Simphony[1], an agent based simulation toolkit. The experimental analysis is to confirm the hypothesis described in Fig(s). 3a and 3b showing the reliability of the framework in providing timely predictions. Experiments were carried out using a P2P network scenario where the framework anticipates the behaviour of members in different network scenarios and predicts the TV of network peers.

The network is modelled with certain properties. Peers interact with others using the communication mechanism found in a P2P network, causing peer states to change. The peers are self-contained as they are uniquely identifiable with a set of characteristics, behaviours and attributes. Also, the peers function independently and interact with other peers by message transfer.

In each experiment, the network consists of dormant peers that do not participate in network activities, misbehaving peers and reputable peers that are active in file upload and download. The network parameters used are in Table 3.

[1] http://repast.sourceforge.net/

The simulation which runs concurrently with the network contains a snapshot of the network and is 20 *ticks* (a compression of time) ahead.

5.1 Implementation Environment

The experiments were carried out with and without the predictive capability of the framework. In the first experiment, trust computation was based on only the online data and past interactions with no predictions from the simulation.

Table 3. Simulation parameters

Parameter	**Value**
Total simulation time (in ticks)	100
Total number of nodes	100
Percentage of malicious nodes	4
Total number of messages transferred	27
Default trust values tv_o^R, tv_h^R	2.5
Online weight μ_o	0.5
Historical weight μ_h	0.3
Prediction weight μ_f	0.17

The TV derived from nodes recent activities tv_o^R is updated every 5 ticks. The tv_o^R from the last update replaces the value of tv_h^R every 5 ticks. The set of past tv_h^Rs is stored in a database for records of historical TVs. With every observation k in the experiment, we compute tv_o^R with the formula $(tv_o^R)k^{th} = ((tv_o^R)k - 1^{th}) - \alpha$ and $(tv_o^R)k^{th} = ((tv_o^R)k - 1^{th}) + (\alpha + 0.5)$ for observed bad and good behaviour respectively, where α is set to 0.5.

From mathematical proofing, the weightings of the TVs serve as a scaling factor and must be such that $0 \leq 1/\mu_o + 1/\mu_h < 1$ for the overall TV to be within the range of 0 and 5. Also, in order for more emphasis to be placed on recent observations, $\mu_o > \mu_h$. For these experiments, the weights were kept at constant values of 0.5, 0.3 and 0.17 for μ_o, μ_h and μ_f respectively.

The simulation component in the second experiment considered 3 possible scenarios and the corresponding TVs for each scenario was obtained. The what-if scenarios considered are collusion, intoxication and failure to cooperate in forwarding of files. A scenario where the peer is active and behaves as expected is also considered. The average of the TVs (tv_f^S) from the scenarios was used and combined with tv_o^R and tv_h^R (of each peer) to compute the overall TV in the second experiment.

5.2 Preliminary Results

In the absence of prediction, the misbehaving nodes colluded and sent inauthentic files through the network at 60 ticks. With prediction, the framework detected and flagged the peer as malicious at 40 ticks and with a downgrade of

its TV immediately. Figure 4 shows the TVs of one of the misbehaving peers, with and without the use of prediction. The figure shows the time gained with the use of prediction with a downgrade of the peer's TV immediately to below the allowed threshold of 2.

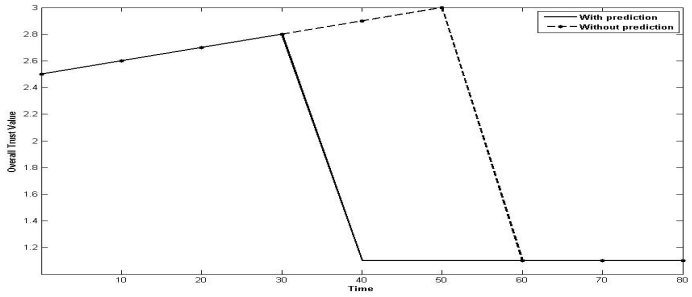

Fig. 4. P2P file-sharing network result (with and without prediction of TVs)

Ultimately in the experiment with prediction, the peer is isolated because its overall TV is below the threshold for other peers to want to cooperate with the peer. This averts the misbehaviour, unlike in the experiment without the prediction (similar to the models that do not anticipate future behaviour by simulation), where the TV was downgraded as a response to the attack. Figure 5 compares the predicted trust with actual TV for some peers. The graph shows the changes in the value of a peer exhibiting intoxication, an untrusted peer whose TV continues to drop and a trusted peer that is active with a high value.

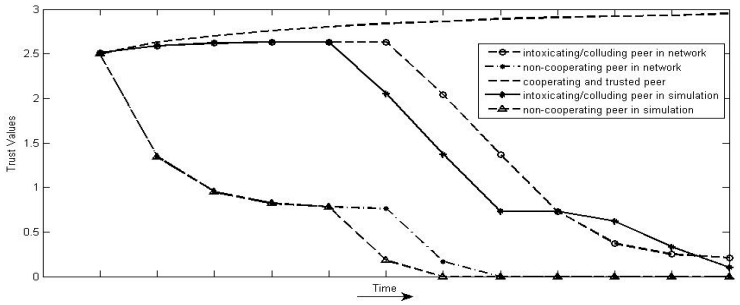

Fig. 5. TVs of a peer and the comparison of the values in the network and simulation

6 Discussion

Ad hoc networks are traditionally known to lack a central entity; therefore, this framework will be most applicable in semi-distributed contexts such as sensor networks, which lend themselves to centralised control.

By comparing the results from the simulation with those from the network, we observed some degree of variance and this might account for possible false-positives or false-negatives generated from the simulation. Hence, we shall explore approaches to improve the correlation of these trust values (i.e. simulated and actual) in the future. Approaches to parameterise the simulation rules for more dynamism in the framework will also be considered in the near future.

In this paper, our experimental study considers only the case of fixed number of identities in the network without random entry and exit of peers. In the future, we shall analyse the implication of dynamic admission and departure of nodes on accuracy of the predictive capability of our framework. Even though we have assumed a constant value for the TV weights, the simulation has a potential to be adaptive in a way that the feedback gathered from the system can help in the adjustments of the weights for future rounds.

7 Conclusion

This paper proposes a dynamic reputation and trust-based framework that is able to predict the behaviour of network members in the future. The framework anticipates future events and considers available information for prediction. Compared to other existing work on trust management, this framework has shown to have the potential to be useful in terms of providing timely information about the domain. This approach is not only useful at the network level but also at a higher level, providing adequate and timely information that allows for countermeasures and making security aware decisions in the network by stakeholders. It can therefore be concluded that the use of monitoring, simulation, and feedback in terms of prediction and control mechanisms, can potentially improve the reliability of systems that rely on trust management to function.

References

1. Balakrishnan, V., Varadharajan, V., Lucs, P., Tupakula, U.: Trust enhanced secure mobile ad-hoc network routing. In: Advanced Information Networking and Applications Workshops, AINAW. vol. 1, pp. 27–33 (2007)
2. Buchegger, S., Le Boudec, J.: Self-policing mobile ad hoc networks by reputation systems. IEEE Communications Magazine 43(7), 101–107 (2005)
3. Buchegger, S., Le Boudec, J.: Performance analysis of the CONFIDANT protocol (Cooperation of nodes: Fairness In Dynamic Ad-hoc Networks). In: Proceedings of the International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc. pp. 226–236 (2002)

4. Chen, H., Wu, H., Hu, J., Gao, C.: Event-based trust framework model in wireless sensor networks. In: Proceedings of the International Conference on Networking, Architecture, and Storage, NAS. pp. 359–364. IEEE Computer Society (2008)
5. Gambetta, D.: Can we trust? Basil Blackwell, New York, trust: making and breaking cooperative relations edn. (1988)
6. Ganeriwal, S., Balzano, L.K., Srivastava, M.B.: Reputation-based framework for high integrity sensor networks. ACM Transactions on Sensor Networks 4(3), 15:1–37 (2008)
7. He, Q., Wu, D., Khosla, P.: SORI: A secure and objective reputation-based incentive scheme for ad-hoc networks. In: Proceedings of WCNC Wireless Communications and Networking Conference. IEEE, vol. 2, pp. 825–830 (2004)
8. Hoffman, K., Zage, D., Nita-Rotaru, C.: A survey of attack and defense techniques for reputation systems. ACM Computing Surveys 42(1), 1–31 (2009)
9. Hu, J., Burmester, M.: Lars - a locally aware reputation system for mobile ad hoc networks. In: Proceedings of the ACM SE Regional Conference. vol. 2006, pp. 119 – 123 (2006)
10. Kamvar, S., Schlosser, M., Garcia-Molina, H.: The eigentrust algorithm for reputation management in p2p networks. In: Proceedings of the 12th international conference on World Wide Web. pp. 640–651. WWW, ACM (2003)
11. Kennedy, C., Theodoropoulos, G.: Intelligent management of data driven simulations to support model building in the social sciences. Computational Science-ICCS 2006. 6th International Conference. Proceedings, Part III LNCS 3993, 562 – 569 (2006)
12. Kennedy, C., Theodoropoulos, G., Sorge, V., Ferrari, E., Lee, P., Skelcher, C.: Aimss: An architecture for data driven simulations in the social sciences. In: Computational Science-ICCS. 7th International Conference. Proceedings, Part I LNCS. vol. 4487, pp. 1098 – 1105 (2007)
13. Kollock, P.: The production of trust in online markets. In: Advances in Group Processes. vol. 16 (1999)
14. Lian, Q., Zhang, Z., Yang, M., Zhao, B.Y., Dai, Y., Li, X.: An empirical study of collusion behavior in the maze p2p file-sharing system. Proceedings of the 27th IEEE International Conference on Distributed Computing Systems 0, 56 (2007)
15. Madey, G., Szabo, G., Barabasi, A.: WIPER: the integrated wireless phone based emergency response system. In: Computational Science-ICCS 2006. 6th International Conference. Proceedings, Part III LNCS. vol. 3993, pp. 417 – 424 (2006)
16. Marti, S., Giuli, T., Lai, K., Baker, M.: Mitigating routing misbehavior in mobile ad hoc networks. In: Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM. pp. 255–265 (2000)
17. Michiardi, P., Molva, R.: CORE: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In: Proceedings of the IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security. vol. 100, pp. 107–121 (2002)
18. Onolaja, O., Bahsoon, R., Theodoropoulos, G.: An architecture for dynamic trust monitoring in mobile networks. In: On the Move to Meaningful Internet Systems: OTM Workshops, LNCS. vol. 5872, pp. 494 – 503 (2009)
19. Onolaja, O., Bahsoon, R., Theodoropoulos, G.: Conceptual framework for dynamic trust monitoring and prediction. Procedia Computer Science 1(1), 1241 – 1250 (2010), iCCS
20. Srinivasan, A., Teitelbaum, J., Liang, H., Wu, J., Cardei, M.: Reputation and Trust-based Systems for Ad Hoc and Sensor Networks. In Algorithms and Protocols for Wireless Ad Hoc Networks, Wiley & Sons (2008)