Thèse de Doctorat de l'université Paris VI
Pierre et Marie Curie

Spécialité

Systèmes Informatiques

présentée par

M. Pedro Braconnot Velloso

pour obtenir le grade de

Docteur de l'université Pierre et Marie Curie

Un modèle de confiance pour les réseaux ad hoc

Soutenance prévue le 18 juillet 2008 devant le jury composé de

Jury

Maryline Laurent-MAKNAVICIUS	**Rapporteur**	**Prof. à l'Institut National des Télécommunications**
Marcelo G. RUBINSTEIN	**Rapporteur**	**Prof. à l'Universidade do Estado do Rio de Janeiro**
Aline C. VIANA	**Examinateur**	**Chargé de Recherche INRIA**
Laurent REYNAUD	**Examinateur**	**Ingénieur à France Telecom R&D**
Guy PUJOLLE	**Directeur**	**Prof. à l'université Pierre et Marie Curie**
Otto Carlos M. B. DUARTE	**Directeur**	**Prof. à l'Universidade Federal do Rio de Janeiro**

Thèse de Doctorat de l'université Paris VI
Pierre et Marie Curie

Spécialité

Systèmes Informatiques

présentée par

M. Pedro Braconnot Velloso

pour obtenir le grade de

Docteur de l'université Pierre et Marie Curie

Un modèle de confiance pour les réseaux ad hoc

Soutenance prévue le 18 juillet 2008 devant le jury composé de

Jury

Maryline Laurent-MAKNAVICIUS	Rapporteur	Prof. à l'Institut National des Télécommunications
Marcelo G. RUBINSTEIN	Rapporteur	Prof. à l'Universidade do Estado do Rio de Janeiro
Aline C. VIANA	Examinateur	Chargé de Recherche INRIA
Laurent REYNAUD	Examinateur	Ingénieur à France Telecom R&D
Guy PUJOLLE	Directeur	Prof. à l'université Pierre et Marie Curie
Otto Carlos M. B. DUARTE	Directeur	Prof. à l'Universidade Federal do Rio de Janeiro

Doctor of Science Thesis
Pierre and Marie Curie University (Paris VI)

Specialization

Computer Science

presented by

Mr. Pedro Braconnot Velloso

Submitted in partial satisfaction of the requirements for the degree of

Doctor of Science of the Pierre and Marie Curie University

A human-based trust model for ad hoc networks

Commitee in charge:

Maryline Laurent-MAKNAVICIUS	**Reviewer**	**Prof. at Institut National des Télécommunication**
Marcelo G. RUBINSTEIN	**Reviewer**	**Prof. at Universidade do Estado do Rio de Janeiro**
Aline C. VIANA	**Examinator**	**INRIA Researcher**
Laurent REYNAUD	**Examinator**	**Engineer at France Telecom R&D**
Guy PUJOLLE	**Advisor**	**Prof. at Pierre et Marie Curie University**
Otto Carlos M. B. DUARTE	**Advisor**	**Prof. at Universidade Federal do Rio de Janeiro**

À ma mère.

Remerciements

Cette thèse, intitulée “Un modèle de confiance pour les réseaux ad hoc”, a été effectuée au Laboratoire d’Informatique de Paris 6, dirigée par Monsieur le Professeur Otto Carlos Muniz Bandeira Duarte, Prof. à l’Universidade Federal do Rio de Janeiro, et Monsieur le Professeur Guy Pujolle, Prof. à l’université Pierre et Marie Curie. Je les en remercie sincèrement.

Je remercie tous particulièrement Madame Maryline Maknavicius, Prof. à l’Institut National des Télécommunication, ansi que Monsieur Marcelo Gonçalves Rubinstein, Prof. à l’Universidade do Estado do Rio de Janeiro, qui ont accepté de juger ce travail et d’en être les rapporteurs.

Je tiens également à remercier Madame Aline Carneiro Viana, Chercheur à l’INRIA, et Monsieur Laurent Reynaud, ingénieur France Telecom R&D, d’avoir accepté de participer au jury de cette thèse.

Je remercie également mes parents, mes frères et Dei qui m’ont toujours soutenu dans toutes les entreprises de ma vie. Finalment, j’aimerais beaucoup remercier Clarissa Seixas qui m’a tendrement soutenue et incentivée.

J’adresse mes remerciements à tous les amis que j’ai pu faire à Paris.

Je tiens aussi à remercier le CNPq/Brésil pour le support financier de cette thèse.

Je remercie tous les membre de l’equipe PHARE et tout le personnel du LIP6.

Résumé

Cette thèse aborde le problème de l'évaluation et de la gestion de la confiance dans les réseaux ad hoc, oú les noeuds accumulent le rôle de routeur, de serveur et de client, les obligeant à coopérer pour un bon fonctionnement du réseau. Plusieurs protocoles et applications ont été proposés puisque les solutions conventionnelles ne sont pas adaptées aux réseaux ad hoc. Cependant, la plupart de ces travaux considèrent l'existence d'une parfaite coopération entre les noeuds supposant qu'ils se comportent tous selon les spécifications des applications et des protocoles précédemment déterminés pour le réseau. Néanmoins, cette condition peut être fausse, à cause de restrictions de ressources ou comportements malveillants. Par la suite, les noeuds peuvent ne pas se comporter comme prévu et entraîner un mauvais fonctionnement du réseau. Par conséquent, un mécanisme permettant à un noeud d'avoir confiance en d'autres noeuds est nécessaire.

Nous proposons un modèle de confiance oú les noeuds d'un réseau ad hoc établissent un rapport de confiance basé sur des expériences et des recommandations préalables. Nous présentons également le *Recommendation Exchange Protocol* qui permet aux noeuds d'échanger des recommandations avec ses voisins. Le but est de rendre les noeuds d'un réseau capables de recueillir des informations pour raisonner, apprendre et prendre leur propre décision. Nous nous concentrons sur fournir aux noeuds le niveau de confiance de chaque voisin direct, c'est-à-dire, un voisin a portée radio. Différemment de la majorité des travaux sur le sujet, notre modèle s'applique bien à d'autres échelles, en limitant les interactions aux voisins directs, ce qui diminue le nombre de messages et, par conséquent, la consommation d'énergie. En outre, elle aide à atténuer les effets des fausses recommandations. Nous présentons le concept de maturité de rapport qui permet aux noeuds d'améliorer l'efficacité du modèle dans les réseaux mobiles. Nous montrons l'exactitude de notre modèle dans un réseau ad hoc de communication directe par des simulations en utilisant un simulateur développé pour notre modèle. L'analyse a, alors, été étendue aux réseaux mobiles ad hoc multisaut, montrant les avantages d'employer le concept de rapport de maturité. Finalement, nous évaluons l'impact des noeuds malveillants qui envoient de fausses recommandations afin de dégrader l'efficacité du modèle de confiance. Les résultats montrent que notre modèle tolère jusqu'à 40% de noeuds malveillants.

Mots-clés :

Réseaux ad hoc mobiles , modèle de confiance, stimulation à la collaboration, sécurité.

Abstract

This thesis addresses the problem of trust evaluation and management in ad hoc networks, in which nodes accumulate the role of router, server, and client compelling them to cooperate for the correct operation of the network. Several new protocols and applications have been proposed and developed because traditional solutions are not adequate for ad hoc networks. Most of the proposed works, however, considers the perfect cooperation among all nodes assuming that they all behave according to the application and protocol specifications. Nevertheless, this assumption may be false, due to resource restrictions or malicious behavior. Eventually, this unexpected behavior can degrade network performance, increase resource consumption, and augment vulnerability to attacks. Therefore, a mechanism that allows a node to infer the trustworthiness of other nodes is necessary.

We propose a trust model based on the concept of human trust. The model builds a trust relationship among the nodes of an ad hoc network based on previous experience and recommendations. We present the Recommendation Exchange Protocol (REP), which allows nodes to send and receive recommendations of its neighbors. The goal is to make nodes capable of gathering information to reason, learn, and make their own decisions. We focus on providing nodes with a trust level for each direct neighbor, that is, a neighbor within the radio range. Different from most related works, our work scales well for large networks by restricting nodes to keep and exchange trust information solely with direct neighbors. This characteristic decreases the number of messages, and consequently, the energy consumption. In addition, it helps to mitigate the effect of colluding attacks of liars in the network. We also introduce the concept of relationship maturity which allows node to improve the efficiency of the proposed model in mobile scenarios. We show the correctness of our model in a single hop network through simulations in a simulator developed specifically for our model. Then, we extend the analysis to mobile multi-hop networks, showing the benefits from using the maturity relationship concept. At last, we evaluate the impact of malicious nodes that send false recommendations to degrade the efficiency of the trust model. The results show that our model tolerates up to 40% of malicious nodes.

Key Words:

Mobile ad hoc networks, trust models, collaboration stimulation, security.

Table of contents

Chapter 1

Introduction

The main difference between a conventional network and an ad hoc network is the lack of infrastructure. For this reason, nodes accumulate the role of router, server, and client, compelling them to cooperate for the correct operation of the network. This peculiar characteristic hinders applications and protocols conceived for conventional networks to perform efficiently in ad hoc networks. Therefore, new protocols specific for this type of network have been proposed and developed. Most protocols and applications for ad hoc networks considers the perfect cooperation among all nodes. It is assumed that all nodes behave according to the application and protocol specifications previously defined for the network. Nevertheless, this assumption may be false, due to resource restrictions or malicious behavior. Consequently, the nodes may not behave as expected causing the network to not work properly. The assumption that nodes behave correctly can lead to unforeseen pitfalls, such as low network efficiency, high resource consumption, and high vulnerability to attacks. Therefore, a mechanism that allows a node to infer the trustworthiness of other nodes is necessary.

The concept of trust has been widely studied in several domains in computer science [1], specially in computer networks [2]. Ad hoc networks can also profit from the benefits that trust models can offer. Providing nodes with a trust level is not only useful when nodes misbehave. In an ad hoc network there is no central entity responsible for configuring, managing, and repairing the stations. According to the paradigm of autonomic networks [3], a node should be capable of self-configuring, self-managing, and self-learning by means of collecting local information and exchanging information with its neighbors. Thus, it is important to communicate only with trustworthy neighbors, because the exchange of information with compromised nodes can deteriorate the autonomy of ad hoc networks. Therefore, a trust system must allow nodes to decide which neighbor is most likely to deliver a packet given a specific destination. Concerning the exchange of information, we should have a level of reliability based on which a node could estimate the veracity of the information, which will lead to an efficient and consistent learning system. Nevertheless, trust systems may suffer from slander and collusion attacks. A slander attack consists of sending false recommendations to injure the reputation of other nodes. Moreover, malicious nodes can work together, in collusion, to improve the effectiveness of the attack. For instance, nodes could lie about a misbehaving node to try to cover its real nature. These attacks can reduce or even ruin the performance of a distributed trust system.

We focus on providing nodes with a trust level for each direct neighbor, that is, neighbor within the radio range. The goal is to make nodes capable of gathering information to reason, learn, and make their own decisions. Different from most related work, our work improves scalability by restricting nodes to keep and exchange trust information solely with direct neighbors. We also introduce the concept of relationship maturity.

We present a trust model based on the concept of human trust. The model builds a trust relationship among the stations of an ad hoc network based on previous experiences and neighbor's recommendations. The ability of assessing the trustworthy of its neighbors brings several advantages. A node is capable of predicting its neighbor's behavior which allows him to decide on which neighbor it is willing to depend to serve as relay. Cooperation among nodes is stimulated by using the trust information to choose the neighbors with which they are ready to collaborate. Nodes learn based on information exchanged with trustworthy neighbors to build a knowledge plane. Moreover, nodes use trust information to detect and isolate malicious behaviors.

1.1 Contributions

We propose a trust model based on the human concept of trust [4]. The model builds a trust relationship among the nodes of an ad hoc network based on previous experience and recommendations. We also propose the Recommendation Exchange Protocol (REP), which enables nodes to send and receive recommendations of its neighbors.

In our model, the interactions among nodes are limited to direct neighbors only. It means that neighbors do not keep trust information about every single node in the network, but just about direct neighbor. This characteristic implies significant lower energy consumption, less processing for trust level calculation, and less memory space. This characteristic fits well to ad hoc networks, which are usually composed of portable devices with power, processing, and memory restrictions. Additionally, the topology changes constantly due to mobility or battery constraints. This approach also minimizes the effect of false recommendations because nodes exchange recommendations with direct neighbors exclusively. First, the number of received recommendations is significantly smaller. Second, recommendations are not forwarded, thus there is no intermediate node to increase the uncertainty of the information. Third, a node can always balance the recommendations with its own experiences to calculate the trust level because nodes do not calculate the trust level of neighbors that are not in direct contact. The decrease in the number of messages sent not only alleviates the network traffic, but also decreases the energy consumption.

We introduce the concept of relationship maturity, which improves the efficiency of the trust evaluation process in the presence of mobility. The basic idea consists of using different levels of recommendations based on the relationship duration. Hence, we use the period of time the recommender node knows the target node as a metric to calculate the weight of its recommendation. Therefore, nodes increase the weight of recommendations based on long term relationship whereas decreases the short term based ones.

We have developed a simulator, which is specifically designed for our model. A set of sanity tests were performed to validate the correctness of the proposed model. The most relevant parameters are extensively analyzed through simulations.

We prove the correctness of our model in a single hop network through simulations. An analysis of the impact of the most relevant parameters on the trust level evaluation process is performed [5, 6]. We also present the benefits from using the concept of relationship maturity in mobile ad hoc networks [7, 8]. Finally, the effect of liars on the trust evaluation process is analyzed [9]. The results show that the relationship maturity parameter decreases the trust level error up to 50%. Moreover, the proposed model is robust, tolerating up to 40% of liars.

1.2 Document Outline

This thesis is organized as follows.

Chapter 2 gives a brief overview of trust in ad hoc networks. The main characteristic of an ad hoc network is the absence of any kind of infrastructure. We present its main advantages and applications. Afterwards, the most important routing protocols are briefly explained. The main challenges in ad hoc networks are identified and we discuss the problem of depending on the collaboration among nodes. We present several approaches to stimulate cooperation in ad hoc networks. The concept of trust used in this work and why trust is important to secure and improve ad hoc networks efficiency is presented. Then we expose how to formalize and represent trust in computer science. We list several domains in which trust can be applied, such as multi-agent systems, e-commerce, and recommender systems. At last, we expose the most significant work on trust for ad hoc networks. We highlight the main differences between our work and the related work.

In Chapter 3 we propose a new trust model based on the human concept of trust. The trust model architecture illustrates the main characteristics of our model and present how the Trust and Learning layers interact with all the other layers on ad hoc network. The Trust layer is extensively detailed showing how a node evaluates the trust level of its neighbors based on its own experiences and the recommendation of other nodes. We introduce the concept of relationship maturity to improve trust evaluation in mobile ad hoc networks. We also describe the Recommendation Exchange Protocol (REP), which permits nodes to send and receive recommendations of its neighbors. Then, a brief discussion on authentication issues in our trust model is presented. At last, we describe the details of our simulator. A new simulator was developed in this work. We explain the main characteristics and present all the relevant parameters of our simulator.

Chapter 4 presents the results of the experiments. First we expose the results related to single-hop ad hoc networks, which demonstrate the correctness of our model and the impact of the main parameters on the trust evaluation process. The main characteristics of trust dynamics in an ad hoc network can also be noticed. Then, we evaluate our model in mobile multi-hop ad hoc networks. We show the effectiveness of the relationship maturity parameter and how the other parameters can be tuned to improve the trust evaluation in the presence of mobility. The last results assess the robustness of our model to slander attacks. We take into account the presence of malicious nodes lying about their recommendations. In the first scenario we consider that malicious nodes collude to hide from other nodes the misbehavior of another malicious node. The other scenario considers nodes that collude to slander one of its neighbors, namely, nodes sending false recommendations to depreciate the reputation of other neighbors.

Chapter 5 presents our conclusions and future work.

1.3 Résumé du Chapitre

La principale différence entre un réseau conventionnel et un réseau ad hoc est le manque d'infrastructure. Pour cette raison, dans les réseaux ad hoc, les noeuds acumulent le rôle de routeur, de serveur et de client, les obligeant à coopérer pour un bon fonctionnement du réseau. Cette caractéristique particulière rend difficile l'exécution des applications et des protocoles, conçus pour les réseaux conventionnels, sur des réseaux ad hoc. Par conséquent, de nouveaux protocoles spécifiques pour ce type de réseau sont proposés et développés.

La plupart des protocoles et des applications pour réseaux ad hoc considèrent l'existence d'une parfaite coopération entre tous les noeuds du réseau. Il est supposé que tous les noeuds se comportent selon les spécifications des applications et des protocoles précédemment déterminés pour le réseau. Néanmoins, cette condition peut être fausse, à cause de contraintes de ressources ou de comportements malveillants. Par la suite, les noeuds peuvent ne pas se comporter comme prévu entraînant un mauvais fonctionnement du réseau. Prétendre que ces noeuds se comportent correctement peut entraîner des problèmes, tels qu'une faible efficacité du réseau, une consommation élevée de ressources et une vulnérabilité importante aux attaques. Par conséquent, un mécanisme permettant à un noeud d'avoir confiance en d'autres noeuds est nécessaire.

Nous proposons un modèle de confiance basé sur le concept humain de confiance. Le modèle établit un rapport de confiance, parmi les noeuds d'un réseau ad hoc, basé sur des expériences préalables et des recommandations. Le but est de rendre les noeuds d'un réseau ad hoc capables de recueillir des informations pour raisonner, apprendre et prendre leur propre décision. Le modèle de confiance se compose d'une couche de confiance et d'une couche d'apprentissage. La première établit comment évaluer le niveau de confiance des voisins. La couche d'apprentissage est responsable de la surveillance et de l'envoi d'informations sur le comportement des voisins à la couche de confiance.

Nous proposons également le Protocole d'Echange de Recommandation (Recommendation Exchange Protocol - REP) qui permet aux noeuds d'envoyer et de recevoir des recommandations de ses voisins. Nous nous concentrons sur fournir aux noeuds le niveau de confiance pour chaque voisin direct, c'est-à-dire, un voisin à portée radio. Par conséquence, les interactions parmi les noeuds sont limitées aux voisins directs. Cela indique que les voisins ne maintiennent pas d'information de confiance sur chaque noeud du réseau, mais juste sur ses voisins directs. Cette caractéristique mène à une consommation d'énergie et de traitement significativement moindre et à une économie de mémoire considérable. Cette caractéristique est adaptée aux réseaux ad hoc qui se composent habituellement de dispositifs portables avec des restrictions de puissance, de capacité de traitement et de mémoire. De plus, la topologie est dynamique. Cette approche réduit également l'effet des fausses recommandations puisque les noeuds échangent des recommandations exclusivement avec leurs voisins directs. D'abord, le nombre de recommandations reçues est sensiblement moindre. En second lieu, les recommandations ne sont pas expédiées, ainsi il n'y a aucun noeud intermédiaire pour augmenter l'incertitude de l'information. Troisièmement, un noeud peut toujours contrebalancer les recommandations avec ses propres expériences pour calculer le niveau de confiance puisque les noeuds ne calculent pas le niveau de confiance des voisins qui ne sont pas en contact direct. La diminution du nombre de messages envoyés peut non seulement alléger le trafic du réseau, mais aussi diminuer la consommation d'énergie.

Nous présentons le concept de maturité de relation qui peut améliorer l'efficacité du processus d'évaluation de confiance en présence de mobilité. L'idée fondamentale comprend l'emploi de la durée de la relation entre le noeud qui recommande et le noeud cible comme métrique pour calculer le poids de chaque recommandation. Ansi, les noeuds peuvent augmenter le poids des recommandations basées sur des relations à long terme tout en diminuant celui des relations à court terme.

Nous montrons l'exactitude de notre modèle dans un réseau ad hoc de communication directe par des simulations en utilisant un simulateur spécialement développé pour notre modèle. L'analyse a, alors, été étendue aux réseaux mobiles ad hoc multisaut, montrant les avantages d'employer le concept de maturité de relation. Finalement, nous évaluons l'impact des noeuds malveillants qui envoient de fausses recommandations afin de dégrader l'efficacité du modèle de confiance.

Le Chapitre 2 donne une brève vue d'ensemble de la confiance dans les réseaux ad hoc présentant les caractéristiques, les avantages, les applications et les défis principaux. Nous discutons le problème de devoir être sous la dépendance de la collaboration entre les noeuds du réseau. Enfin, nous présentons plusieurs approches pour stimuler la coopération dans les réseaux ad hoc. Nous présentons le concept de confiance utilisé dans ce travail et expliquons pourquoi elle est importante pour sécuriser et améliorer l'efficacité des réseaux ad hoc. Nous énumérons plusieurs domaines dans lesquels le concept de confiance peut être appliqué. Enfin, nous exposons les travaux les plus significatifs sur la confiance pour réseaux ad hoc. Nous accentuons les principales différences entre notre travail et les travaux relatifs. Dans le Chapitre 3 nous proposons un nouveau modèle de confiance basé sur le concept humain de confiance. L'architecture du modèle de confiance illustre les principales caractéristiques de notre modèle et montre comment les couches de confiance et d'apprentissage interagissent avec les autres couches du réseau ad hoc. La Couche de confiance est minutieusement détaillée et montre comment un noeud évalue le niveau de confiance de ses voisins. Nous présentons le concept de la maturité de la relation pour améliorer l'évaluation de confiance dans les réseaux ad hoc mobiles. Nous décrivons également le Protocole d'Echange de Recommandation (Recommendation Exchange Protocol - REP) qui permet à des noeuds d'envoyer et de recevoir des recommandations de ses voisins. Les détails de notre simulateur sont exposés. Un nouveau simulateur a été développé dans ce travail. Nous expliquons les caractéristiques principales et présentons tous les paramètres importants de notre simulateur. Le Chapitre 4 présente les résultats des expérimentations. D'abord nous exposons les résultats liés aux réseaux ad hoc de communication directe, qui démontrent l'exactitude de notre modèle et l'impact des paramètres principaux sur processus d'évaluation de confiance. Les caractéristiques principales de la dynamique de confiance dans un réseau ad hoc peuvent être également notées. Puis, nous évaluons notre modèle dans les réseaux mobiles ad hoc multisaut. Nous montrons l'efficacité du paramètre de maturité de rapport et comment les autres paramètres peuvent être syntonisés pour améliorer l'évaluation de confiance en présence de mobilité. Les derniers résultats évaluent la robustesse de notre modèle contre les attaques de diffamation. Nous prenons en considération la présence de noeuds malveillants qui envoient des fausses recommandations. Le Chapitre 5 présente nos conclusions et travaux futurs.

Chapter 2

Trust in Ad Hoc Networks

Wireless communications in 802.11 use one of the two available operation modes. In the infrastructure mode, all nodes communicate through an access point. Therefore, when a node wants to send a message to another node in the same network, the message is sent to the access point (AP) and the AP relays the message to the destination node. In the ad hoc mode, however, there is no infrastructure and nodes directly communicate with each other. The main advantages of ad hoc networks are flexibility, low cost, and robustness. Ad hoc networks can be easily set up, even in deserts, and can endure to natural catastrophes and war. Therefore, ad hoc networks fit well where there is no infrastructure [10] and it is too expensive to build it, or when local infrastructure is not reliable, as for military operations in the enemy territory.

The simplest ad hoc network is composed of nodes that communicate strictly with direct neighbors, namely, nodes within the radio range [11]. In this kind of network, so-called single-hop networks, there is no need for routing, as show in Figure 2.1(a). As the network gets larger, however, nodes further apart also need to communicate and this simplistic design is not enough. Another possibility of ad hoc networks consists of nodes that communicate over a multi-hop radio network in a utterly decentralized manner. Therefore, each node must act both as a host and a router. Figure 2.1(b) illustrates this example and available multi-hop paths are shown as continuous lines. In Mobile Ad Hoc Networks (MANETs), the network topology is dynamic due to mobility of nodes. Moreover, each node must implement distributed medium access control mechanisms and deal with exposed and hidden terminal problems.

The nature of the wireless and mobile environment makes ad hoc networks vulnerable

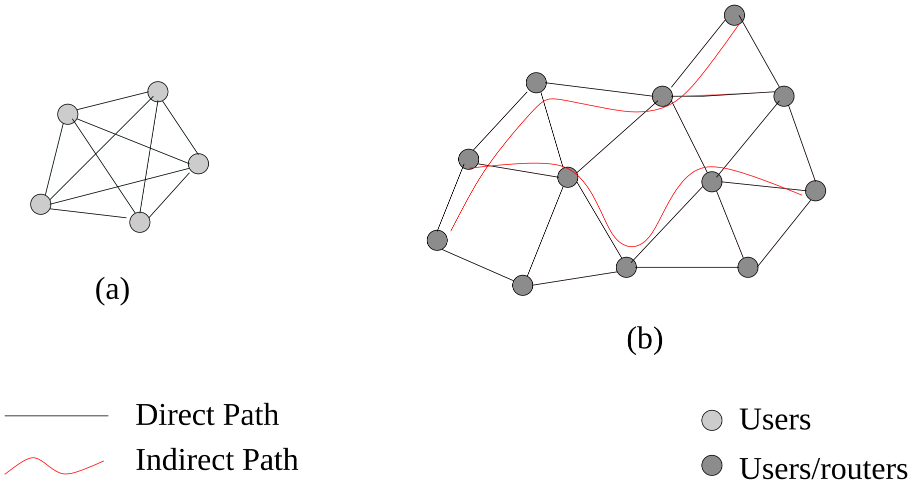


Figure 2.1: Single-hop and multi-hop ad hoc networks.

to different sorts of attacks. Such networks are susceptible to attacks ranging from passive eavesdropping to active interfering. In particular, attacks in ad hoc networks can cause congestion, incorrect propagation of routing information, prevent services from working properly or even shut them down completely [12, 13]. In mobile ad hoc networks, the scenario is even more challenging. Portable devices have very limited resources in terms of power, processing, and memory. These constraints impose several restrictions to well-known solutions only available for wired networks. Providing the same level of security in ad hoc networks is hard due to its distributed nature and also crucial to keep the network functional and safe from malicious nodes.

Another important aspect to be considered in ad hoc networks is the absence of infrastructure and centralization. There is no dedicated entity responsible for providing the basic functionalities. Although the decentralization brings the advantage of robustness to the network, since the single point of failure does not exist, conventional solutions to traditional network problems are not suitable for ad hoc networks.

2.1 Routing protocols

Routing in ad hoc networks is an important and challenging issue. Due to the lack of infrastructure and mobility, routing protocols for wired networks do not fit to ad hoc networks. Thus, a new set of routing protocols dedicated to this kind of network has been proposed [14]. There are three main categories for ad hoc routing protocols. The classification is based on how routing information is acquired and maintained by mobile nodes. According to this classification, ad hoc routing protocols are denoted as pro-active, reactive or hybrid.

Pro-active algorithms are inspired by traditional algorithms for wired networks. In proactive routing protocols, also called “table driven”, each node keeps routing information to every node in the network. The routing tables are periodically updated regardless of whether data traffic exists or not. Therefore, for mobile nodes, using proactive routing algorithms, the overhead to maintain up-to-date network topology information is high [15]. The main advantage from this approach is that there is no delay of route discovery, since a node has all routes updated when needed. Examples of this kind of protocols are the Destination-Sequenced Distance Vector (DSDV) [16] and Wireless Routing Protocol (WRP) [17].

In a reactive routing protocol, also called ”on-demand”, routes are searched only when needed. Every time a node sends a packet, before sending, it must invoke a route discovery procedure and wait for an answer. If there is no answer after a certain period of time, then it is assumed that the node is unreachable . Since nodes do not need to keep and update routing information for every node, reactive routing protocols have better scalability than proactive routing protocols in mobile ad hoc networks. However, nodes using reactive routing protocols may suffer from long delays for route discovery before sending or forwarding data packets. Dynamic Source Routing (DSR) [18] and Ad hoc On-demand Distance Vector routing (AODV) [19] are well known examples of reactive routing protocols.

Hybrid routing protocols combine the advantages of both proactive and reactive routing protocols and overcome their shortcomings. Hybrid routing protocols, usually exploit hierarchical network architectures. They use proactive routing and reactive routing for different hierarchical levels. Examples of hybrid routing protocols for mobile ad hoc networks are the Zone Routing Protocol (ZRP) [20] and the Zone-based Hierarchical Link State routing (ZHLS) [21]. These two protocols are also known as zone-based hierarchical protocols.

There are several other classifications for ad hoc networks, such as, cluster-based protocols [22], core-based protocols [23, 24], location-based protocol [25], link-stability-based protocols [26].

2.1.1 Routing attacks

Ad hoc networks suffer from a significant vulnerability to attacks caused by its intrinsic characteristics [12]. As a consequence, the routing process presents several weaknesses which are considered as security problems [27]. Malicious nodes can perform several types of attacks in ad hoc networks, such as:

- Spoofing: attackers create false identities
- Sybil: a malicious node assumes multiple identities of other nodes. Consequently, the attacker can answer to routing requests
- DoS: the Denial of Service (DoS) in ad hoc routing protocols is an adaptation of traditional DoS attacks [28, 29]. The attack consists of one node, or colluding nodes, continually sending requests to the target node until it runs out resources, such as battery, memory, or processing power
- Sinkhole: in this attack, the attacker announces to its neighbors as being the most attractive relay in the available multi-hop routes. All packets are then forwarded to the attacker and dropped
- Warmhole: in this attack, two malicious nodes create a tunnel to forward packets to each other. Usually, this path has a lower latency, for example, passing through a wired link. Therefore, other nodes tend to choose this new path to send their packets allowing the attackers to have access to all packets
- Byzantine: malicious nodes manipulate control routing messages to create routing loops and non-optimal routes
- False routing messages: attackers send false routing messages to overflow the routing table or to poison it with fake routes
- Misdirection: the goal of this attack is to cause a denial of service in a certain node. In the misdirection attack, a malicious node creates routing messages to direct the network flows towards the target region or node
- Grayhole or selective forwarding: in this attack, malicious nodes do not forward all the packets they receive. For example, an attacker decides not to forward packets to

a specific destination or zone of the network. The blackhole attack occurs as a special case of the grayhole attack where the attacker does not forward any packet at all

Developing a proper threat model to evaluate security properties in mobile ad hoc routing protocols presents a significant challenge. Andel and Yasinsac [30] present an adaptive threat model to evaluate route discovery attacks against ad hoc routing protocols. Their approach permits the evaluation of the reliability of the routing process in ad hoc networks. It is also possible to identify minimum requirements an attacker needs to break a specific routing protocol.

Some authors propose modifications or extensions to existing routing protocols in order to secure or cope with selfish-nodes. Fourati and Al Agha [31] present a shared secret-based algorithm to secure the Optimized Link State Routing protocol (OLSR) [24]. Gawedzki and Al Agha [32] address the problem of nodes that intentionally drop data traffic but behave correctly with respect to control messages. They address the problem of dealing with the presence of such nodes in networks that use a proactive routing protocol. Their solution is well suited for existing proactive routing protocols because it can use existing control messages and does not require any synchronization between the nodes. Perrig *et al.* [33] present the Secure Efficient Ad hoc Distance Vector (SEAD), which is based on DSDV. The goal is to provide a robust protocol against the routing table poisoning attack. The key feature of the proposed protocol is authenticating route update messages using hash functions. ARIADNE, based on DSR protocol, ensures point-to-point authentication of a routing message by combining a shared key between the two parties [34]. Secure Routing Protocol for Ad Hoc Networks (ARAN) is another extension to on-demand routing protocols to provide authentication to nodes in the routing process [35].

2.2 Collaboration in ad hoc networks

Due to the lack of infrastructure, protocol and applications conceived for ad hoc networks are based on the collaboration of nodes to work properly. As showed in Section 2.1, all nodes participate in the routing process. Thus, routers of an ad hoc network are under the control of the users instead of administrators. The same problem exists for access control mechanisms, authorization and authentication techniques, key distribution protocols, service location algorithms, etc. Therefore, the collaboration among nodes is a crucial issue in ad hoc networks.

Although the collaboration of nodes is an usual assumption in ad hoc networks, it is not so obvious that the collaboration exists in practical networks. Several reasons can lead a node not to collaborate. In MANETS, nodes have to cope with several resource constraints like energy consumption and storage capacity [36]. Consequently, nodes must forward packets for other neighbors spending energy without receiving any direct gain for this act. Therefore, there is no interest for nodes to participate in the routing and forwarding process. In this context, the concept of selfish behavior is introduced. Selfish behavior is characterized by a node that does not participate in the routing process, and/or does not forward other nodes traffic, and/or does not contribute to other network services, or applications, in order to save energy. A malicious node can also decide not to forward traffic for other nodes or not to participate in the routing task causing significant damage to the network performance. The difference between selfish and malicious behaviors is the goal. Malicious nodes intend to disrupt the network whereas selfish nodes seek to spare its own resources. Nevertheless, both behaviors have the same effect on the network performance, such as, low efficiency and high energy consumption. In [37] the authors use a game theory approach to evaluate the effect of malicious users in non-structured networks, such as ad hoc networks. They conclude that the impact of bad behavior is strong related to the topology, that is, the proportion of malicious neighbors of a given node.

Yu and Liu [38] state that before ad hoc networks can be successfully deployed in autonomous ways, the issues of cooperation stimulation and security must be resolved first. Several work propose mechanisms to stimulate the cooperation among nodes. Their goal is to avoid selfish and malicious behavior in order to guarantee the right implementation of routing and forwarding tasks by all nodes of the networks. There are different models for encouraging cooperation in ad hoc networks. One common approach is to use a system based on credit to stimulate cooperation and avoid selfish behaviour [39, 40, 41, 38, 42, 43]. The basic idea consists of nodes that send or receive a message must deduct one unit of credit while forwarding nodes should increment one unit. Zhong *et al.* [39] proposed a cheat-proof credit-based system for ad hoc networks with selfish nodes. There is an entity (Credit Clearance Service - CCS) responsible for all the credit and transactions. Each node must buy a certain amount of credit from this entity. When a node wants to send a message it has to pay to the CCS an amount of credit that depends on the number of forwarding nodes and whether the message has reached the destination or not. Each forwarding node must keep a receipt for each forwarded message. The CCS will pay the forwarding nodes that reported a receipt for that particular message. They show, through a game theory analysis, that the best strategy is no cheating. This means that there is no way of a node or a set of nodes to cheat and get more than it would get if it has not.

Buttyán and Hubaux [41] assume that every node has a tamper resistant security module, which keeps a nuglet counter. When a node wants to send its own packet, it must estimate the number n of intermediate nodes necessary to reach the destination. If the nuglet counter is less than n then the node cannot transmit its packet. Otherwise, the node sends its packets and decreases by n the nuglet counter. The counter is increased by one whenever a packet is forwarded for the benefit of other nodes.

A model that incorporates incentives for users to act as relay nodes and to be rewarded with their own ability to send traffic is proposed in [42]. They use a pricing mechanism in which nodes define their own cost of transmitting a packet based on their power and bandwidth constraints. Therefore, nodes can make decentralized decisions concerning the choice of the flows on potential routers. The nodes make such decisions based on prices announced by relevant nodes.

Yu and Liu [38] propose an attack-resilient cooperation stimulation system for autonomous ad hoc networks that provides mechanisms to stimulate cooperation among selfish nodes in adversarial environments. The basic idea of their mechanism is that if a packet can be successfully delivered to its destination within the specified delay constraint, the source of the packet will get some payoff, otherwise, it will be penalized. Once a node has successfully forwarded a packet on behalf of another node, it will request a receipt from its next node on the route and send this receipt to the source of the packet to claim credit.

One of the first propositions for detecting misbehaving nodes presented in [44] is based on the monitoring of direct neighbors in order to identify misbehaving users and avoid routes with these nodes. They propose a mechanism that uses two applications: Watchdog and Pathrater. The first one runs on every node monitoring the behavior of the other nodes of the network. The Pathrater application uses the information collected to calculate the route with the highest reliability. Although this approach allows the identification of misbehaving nodes, there is no reward/punishment mechanism to well-behaving and misbehaving nodes, respectively.

Some work focus on identifying selfish behaviors and analyze the proposed mechanisms using game theory [45, 46]. Capra *et. al* [47] present a game-theoretic model to facilitate the study of the non-cooperative behaviors in ad hoc networks. Using the proposed model, they analyze incentive schemes to motivate cooperation among nodes to achieve a mutually beneficial networking result. Al-Karaki *et. al* [48] offer a new scheme that can stimulate and also enforce nodes to cooperate in a selfish ad hoc environment. They also present a mechanism to detect and exclude potential threats of selfish nodes.

Nevertheless, all these work are restricted to collaboration of nodes to relay traffic for other neighbors. We are concerned with all kinds of distributed mechanisms and applications, such as authentication, key distribution, access control, management, etc.

2.3 The trust concept

The trust concept possesses several definitions in literature [49]. McKnight and Chervany [50] present a conceptual typology of trust constructs.

- Disposition to trust: it means the extent to which one has a consistent tendency to be willing to depend on others across a broad spectrum of situations and persons
- Institution-based trust: people can rely on others based on previous structures, situations or roles that provides assurances that everything is going to be fine
- Trusting beliefs: one believes that the other person has the ability or power to do what one needs in a situation where negative consequences are possible
- Trusting intention: one is willing to depend on other persons to execute a given task not based on having control or power over other party

McKnight and Chervany [51] argue that there are different types of trust and that every work should specify which one is being addressed. In computer networks, we are basically concerned with Trusting belief and Trusting intention.

Trust can be defined as an entity belief about another party (a person, an organization or a device) based on a set of well-established rules and its expectations. Trust can be considered a fuzzy notion across persons or across areas of competence. When we say “I trust you”, in fact this means that I trust you more than a threshold value that I consider to be complete trust. No matter how close they are to each other, different people may trust very different things, even in front of the same evidence. Trust is conditional transitive because it depends on the scope, namely, it applies to a specific purpose or domain of action. For example, a person A may trust a person B to accomplish a specific task, but not another task. Most people, for example, trust their mother in general, but rarely for piloting a helicopter. In this way, trust can vary in face of different factors.

Trust can be classified in Functional trust and Referral trust [52]. Functional trust represents the trust on someone doing something for you while referral trust denotes a recommendation, that is, an opinion of a trusted person about a third party. For instance, if Alice trusts Bob as a good recommender of dentists, Alice's referral trust on Bob for dentist recommendation is considered direct. When Bob recommends Eric as a good dentist to Alice, because Eric has proven to be a good dentist to Bob, Alice has a indirect functional trust on Eric. Therefore, extending the previous example, a chain of direct referral trust can be formed with a direct functional trust at the end. Accordingly one person can derive trust based on direct trust (recommendations) always respecting the same context, as illustrated in Figure 2.2.

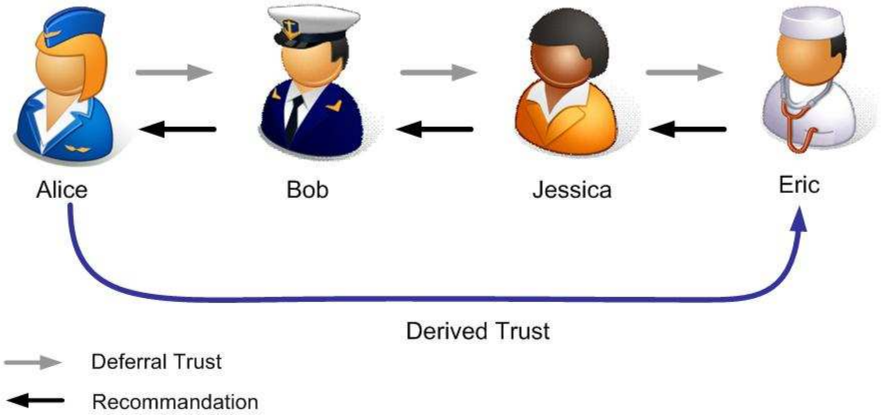


Figure 2.2: Referral trust and derived trust.

Trust pre-exists security. Trust is a “natural phenomenon”, and it has existed for millennia, before any concept of security was invented. Security is a set of techniques designed for providing or improving the trust on systems, services, or devices. Therefore, adding security techniques helps to derive trust. For example, if I trust that my personal computer is not compromised and that the cryptographic algorithm running on both sides is not (yet) broken, then I can trust that what I see on my screen is indeed a Web page corresponding to my bank; hence, in this case, I can carry out my e-banking transactions with the legitimate belief that I will not be defrauded. This simple example illustrates that any security mechanism requires some level of trust in its underlying components.

For network security, the notion of trust corresponds to a set of beliefs among entities that participate in various protocols [53]. Trust influences decisions like access control, choice of public keys, etc. Trust relationships are determined by rules that evaluate the evidence generated by the previous behavior of an entity within a protocol. What is meaningful depends on the specific protocol (application), and on the entity that evaluates the trust relation. The application determines the exact semantics of trust, and the entity determines how the trust relation will be used in the following steps of the protocol.

Reputation mechanisms are sometimes incorrectly considered as trust models. Reputation can be considered as a collective measure of trustworthiness, in the sense of reliability, based on the recommendations or ratings from members in a community [2]. Reputation values result from knowledge, information, and evidences about the evaluated person or thing. Trust models can use reputation values to take decisions about the trustworthiness; however, trust phenomenon exists independently of reputation mechanisms. Reputation values are one of all evidences whose trust models can apply. Trust can be derived from a combination of received recommendations and personal experience. The major challenge for distributed reputation systems consists in assessing the truthfulness of the recommendations [54]. Obreiter [54] introduces the concept of gathering evidences to verify the recommendation.

Kinateder and Rothermel *et al.* [55] present an architecture and algorithms for a distributed reputation system, most specifically for online recommendation systems. The recommendations are separated in different categories and sub-categories, like books, books-security, cars-sports, and etc. An entity must assign a trust value for its neighbors based on previous experiences for each recommendation category. Entities publish these trust values and the reputation of a given entity A is the average trust of all other entities towards A.

2.4 Trust applications in computer science

The concept of trust has been widely studied in several domains in computer science [1], specially in computer networks [2]. Some work has been done to formalize, represent, and manage trust in computing environments. In a common approach, trust is based on the concept of public key certificates, such as PGP [56] or X.509 [57]. In this approach, a certification authority issues a digital certificate to assure that a specific public key belongs to a specific entity. The goal is to provide an access control mechanism based on the user authentication to reduce the risk of interaction with other entities. The basic idea is to create a web of trust in which users and service providers can authenticate each others. Therefore, the trust concept in this kind of systems refers to mechanisms to verify the identity of an entity, namely, if an entity is really the entity it claims to be.

Another possibility of formalizing trust is using formal logic concepts. Such approach uses simple relational formalism and logic operators to express trust rules and to reason about trust properties. They also use modal logics to express possibility, necessity, belief, knowledge, etc. Demolombe [58, 59] defines trust as a mental attitude of an agent with respect to another agent. In his work, an agent can be human or artificial, like a machine, a sensor, a program, etc. He presents formal definitions of trust in a framework based on modal logic.

Trust can also be expressed by analytical expressions based on the probability of success of a given operation. This approach allows using probabilistic theory to infer trust and to take decisions. Josang and Lo Presti [60] analyze the relationship between risk and trust. They propose a probabilistic trust model that considers the probability of success and the risk of taking a decision. Some authors [61, 62] apply game theory concepts to derive analytical expressions to infer trust and to capture the behavior of entities in situations, in which the success in making choices depends on the choices of others. Morselli *et al.* [61] propose a game-theoretic framework for analyzing trust-inference protocols. In addition, trust is inferred using linear equations [4].

2.4.1 Multi-agent systems

There is a considerable effort for developing trust models in multi-agent systems. A number of dynamic properties for trust dynamics were identified [63], mainly on the basis of intuition and common sense. Jonker *et al.* [64] perform human experiments on the dynamics of trust over time depending on positive or negative experiences to verify these properties. The results show that positive experiences produce an increasing or at least nondecreasing effect on trust, while negative experiences have a decreasing or at least non-increasing effect. They also notice that is easier to destroy trust than to build trust: a single negative experience can provoke a stronger negative effect on trust than the positive effect shown by a single positive experience.

Mui *et al.* [65] present a probability based computational model of trust and reputation. They introduce the concept of reciprocity among agents in a social network to derive trust and reputation. The reciprocity means that agents tend to cooperate with other agents after receiving positive actions, or favors. In the same way, agents respond to negative actions with negative responses, like revenge.

A common application of multi-agent systems is to automate commercial transactions in electronic market places. Michalakopoulos and Fasli [66] consider trust as a concept that agents in a market place can use to take decisions on who they are going to transact with.

The paradigm of mobile agents is also based on multi-agent systems. “Mobile agents are programs that help users by acting on their behalf in performing a number of tasks in the network”. These agents migrate to specific hosts to get the information that the user needs, saving time, bandwidth, and money. This technique is common used in network management applications [67]. Buttyan *et al.* [68] proposed a trust system to mobile agent platforms.

2.4.2 Electronic commerce

Trust or reputation management is an important issue in electronic business [69]. Trust can stimulate users and organizations to accept online activities as safe places for interacting and doing business [70]. The authors in [70] are concerned about bringing trust to online activities. They study trust management approaches in order to assess their potential for stimulating on line activities.

For Carbone *et al.* [71] the collaborations between humans and organizations are enabled by the concept of trust. Their work aims to transfer these forms of collaborations to modern computing scenarios. They focus on the foundations of formal models for trust in Global Computing environments, like the internet, for the use of trust-based security mechanisms as an alternative to the traditional ones. The basic idea consists of a “trust engine” and a “risk engine” coupled together as part of a “principal”. The trust engine is responsible for updating trust information based on direct and indirect observations or evidence, and to provide trust information to the risk engine as input to its procedures for handling requests. The risk engine will feed back information on the behaviors of principals as updating input to the trust engine.

In trust management systems, like e-Bay [72] or Amazon marketplace [73], there is no reward for providing feedback. Therefore, only users that want to increase their reputation by advertising positive feedbacks about each other or users that seek revenge against a bad experience are prone to participate. This excludes all users that have had average experiences with other users [74]. In [75] the authors propose a framework for providing incentives for honest participation in distributed trust management. They present a reward model to benefit users that participate reporting their experiences about the interactions with other users. A probabilistic honesty metric to avoid malicious users that submit inaccurate or random statements to obtain rewards is proposed.

2.4.3 Recommender systems

Recommender systems are used to suggest items that users might be interested in [76]. In these kinds of systems two techniques can be employed to generate all the recommendations. The first one uses a content based system which requires manual intervention and do not scale to large data bases [77]. The other technique consists of a collaborative filtering system which is based on user opinions. The basic idea is that users rate specific items and the system collects these opinions. Therefore, the recommendations are based on the identification of similar users and the suggestion of items that have a good rate from these users. In [77], the authors propose an extension to collaborative filtering systems that consider trust relationship between users to improve the efficiency of recommender systems.

Augmented reality systems provides relevant navigational information and recommendations to surveyors, tourists and any one that uses city streets [78]. In such systems users can create notes or recommendations that are available for all other users. For instance, users can leave notes about touristic places, recommendations about restaurants and stores. Ingram [79] proposes a trust-based filtering for augmented reality.

2.4.4 Other applications

The explosion of the semantic web-based social networks, such as Facebook [80], Orkut [81], among others, motivated Golbeck and Hendler [82, 83] to study the accuracy of metrics for inferring trust and reputation in such systems. They have created a trust extension to FOAF (Friend Of A Friend) project that allows people to rate each other. Ziegler and Lausen [84] introduce a new classification scheme for trust metrics and propose a new method for local group trust computation for semantic web trust management.

Jiang and Baras [85] propose a trust evaluation model for autonomous networks. They model trust relationship as a directed graph in which nodes are entities and links represent trust relationships. Their model considers local interactions to obtain a confidence value for a link. The trust level of a node is obtained by a weighted sum of the confidence values of its neighbors.

Nixon *et al.* [86, 87] propose trust based architecture for interaction and collaboration in global computing systems. Their purpose is to build an architecture that comprises trust formation, trust evolution and trust exploitation, forming a basis for risk assessment and taking decisions. The architecture is based on evidences observed by the entity itself. Trust formation might consider the recommendation of other entities, but only first hand experiences are considered as valid recommendations.

Taylor *et al.* [88] present the Practical Architectural approach for Composing Egocentric trust (Pace) provides detailed design guidance on where and how developers can incorporate trust models into decentralized applications. In addition, Pace's guiding principles promote countermeasures against threats to decentralized systems.

2.5 Trust in Ad Hoc Networks

Ad hoc networks rely on collaborative behavior of nodes to work properly. Therefore, nodes must trust each other at some level to allow distributed applications, including routing and admission control. The lack of a fixed network infrastructure, high mobility of the nodes, limited-range, and unreliability of wireless links are some of the characteristics of MANETs that limit and make more complex the existence of a trust establishment scheme. In contrast with fixed networks where trust is generated in centralized ways and cached in certificates, MANET nodes need to generate trust evidences among themselves and evaluate "on the fly" their beliefs to form trust relationships dynamically.

A naive cooperation in MANETs might lead to low efficiency, high energy consumption, and network attacks. The behavior of the nodes is dynamic and depends on their goals and constraints, which might lead to distinct behaviors. Nodes must decide what is best for themselves but in a context of minimum collaboration, like in a society. In face of these characteristics, the existence of a trust model helping nodes to manage trust evidences is essential.

Trust can stimulate cooperation among nodes and consequently avoid selfish behavior. Trust can also be used to minimize the effect of malicious nodes. The trust model must provide nodes with mechanisms to generate, manage, and exchange trust information, respecting MANETs characteristics and constraints. Besides, the model must work in a totally distributed way.

In general, the mechanisms proposed to provide trust establishment between nodes in ad hoc networks apply to the network layer trying to protect or enforce the two basic functions of this layer: routing and packet forwarding [89].

There are some well known management systems like KeyNote [90], PolicyMaker [91], Simple Public Key Infrastructure (SPKI) [92], and Simple Distributed Security Infrastructure (SDSI) [93] that attempt to manage security in large-scale distributed networks. However, all these systems are based on the use of credentials that delegate permissions, which is not suitable for ad hoc networks due to the lack of any kind of infrastructure. There are three essential differences from distributed trust management in ad hoc networks and traditional centralized networks. In ad hoc networks trust evidence is provided by peers, which can be incomplete and even incorrect. The second is that trust information is exchanged locally through individual interactions. The last one is that trust evaluation is performed in a distributed manner. Ren *et al.* [94] propose a probabilistic solution for securing ad hoc networks. Their goal is to provide an authentication mechanism using a modified version of a traditional distributed trust establishment approach based on public-key cryptography. They introduce a secret dealer to solve the problem of the bootstrapping phase in authentication. All nodes trust the secret dealer *a priori*, which keeps the node identification and a public key associated (Node ID, Public_key), for each node. In the initial phase the secret dealer distributes a secret list, containing a list of node IDs and public keys for all nodes. After then, nodes authenticate new members in a distributed manner without the interference of the secret dealer that is no longer needed.

Several papers propose trust models for ad hoc networks. An important characteristic in a trust model is what kind of information is used to evaluate the trustworthiness of other nodes. Nodes can use local information and remote information. Another important characteristic is how nodes obtain the necessary information.

Basically, there are two approaches to gather information about other nodes. The first one is based on the observation of neighbors, in which nodes monitor the behavior of their neighbors. Therefore, in this approach nodes use only local information, namely, the information they have collected by themselves. The other technique consists of exchanging information with other nodes. Thus, nodes consider the opinion of their neighbors in the trust level evaluation. We can separate existing trust models into two groups according to the kind of information they use. One group, with trust models that use only local information. The other group comprises trust models that use both, local and remote information.

2.5.1 Trust models based on local information

The basic idea is to generate trust values describing the trustworthiness, reliability, or competence of individual nodes, based on some monitoring schemes. Albers *et al.* [95] propose a general intrusion detection architecture to improve trust based approaches on ad hoc networks.

Theodorakopoulos and Baras [96, 97] analyze the issue of evaluating the trust level as a generalization of the problem of short path in a oriented graph, where the edges correspond to the opinion that a node has about other node. They consider that nodes use just local information to establish their opinions. The opinion of each node includes the trust level and a value that represents the precision of the trust level. The main goal is to enable nodes to indirectly construct trust relationships using exclusively local information.

Sun *et al.* [98] have developed one framework capable of measuring the trust level and propagating it through the network in order to make routing more secure and to assist intrusion detection systems. The framework also includes a defense mechanism against malicious nodes. They use a probabilistic model based on the uncertainty of a neighbor to execute one specific action and consider only local information.

He *et al.* [99] propose an architecture for stimulating the collaboration based on the reputation of nodes. The system is based only on the local information to evaluate the reputation of nodes. The goal is to detect and to punish nodes that do not participate in the routing process.

Gray *et al.* [100] propose to develop a trust framework that enables access control based on trust-based admission control policies that define the trust relationship between entities in collaborative ad hoc applications. They use an interaction monitor to gather local information about interactions between nodes. Based on this information they calculate a trust value, expressed as the probability that a certain node behaves correctly. In another work, Gray *et al.* [101] propose a solution to provide security on ad hoc networks which is based on the human notion of trust, risk, and recognition in human ad hoc collaborative networks. They describe a trust based security architecture that considers small world characteristics and includes entity recognition, trust-based admission control, risk assessment and trust management.

The main difference of these works and our trust model is that they use only local information on the trust evaluation process. Our trust model considers local and remote information. We show in Chapter 4 that the recommendations of other nodes can improve the trust evaluation even in the presence of liars.

2.5.2 Trust models based on local and remote information

In probabilistic-based models, a common approach consists of using Bayesian networks, which is a probabilistic tool that provides a flexible means of dealing with probabilistic problems involving causality [102, 103]. Buchegger and Le Boudec [104] investigate the trade-off between robustness and efficiency of reputation systems in mobile ad hoc networks. A mechanism based on Bayesian statistics is used to filter slanderer nodes. The proposed system considers local information and the recommendation of other nodes to compute the reputation of a specific node. They show that taking into account the recommendations of other nodes can speed up the process of discovery of malicious nodes.

Chinni *et al.* [105] offer a distributed trust model for certificate revocation in ad hoc networks. The proposed model allows trust to be built over time as the number of interactions between nodes increase. Furthermore, trust in a node is defined not only in terms of its potential for maliciousness, but also in terms of the quality of the service it provides. The trust level of nodes where there is little or no history of interactions is determined by recommendations from other nodes. If the nodes in the network are selfish, trust is obtained by an exchange of portfolios. Bayesian networks form the underlying basis for this model.

Liu *et al.* [106] propose a trust model for ad hoc networks based on the distribution of threat reports. The goal is to make security-aware routing decisions, where nodes use the trust level as an additional metric for routing packets. Nevertheless, they assume that nodes cooperate with each other, which is not always the case. They also assume that all nodes are capable of detecting malicious behavior by means of Intrusion Detection Systems (IDS). This assumption leads to high energy consumption, which is clearly not an appropriate option for ad hoc networks.

Another approach consists of using linear functions to infer trust. Schweitzer *et al.* [107, 108] present a distributed mechanism for trust propagation and consolidation for ad hoc networks. They use a linear function based on the concept of trust, distrust and uncertainty.

Pirzada and McDonald [109, 110] propose another trust model for ad hoc networks to compute the trustworthiness of different routes. Nodes can use this information as an additional metric on routing algorithms. Although the authors present an interesting approach, the model presents several disadvantages. For instance, it is currently restricted to the Dynamic Source Routing (DSR) protocol. It also relies on using promiscuous mode ignoring the energy constraints of mobile nodes. Finally, it requires each node to store information for all other nodes in the network, which is clearly non-scalable.

Virendra *et al.* [111] present an architecture based on trust that allows nodes to make decisions on establishing keys with other nodes and forming groups of trust. Their scheme considers trust self-evaluation and recommendation of other nodes to compute trust. Although we have a similar approach, our trust model differs in the following way. Their trust self-evaluation is based on monitoring nodes and a challenge-response system. We propose a self-learning and context based approach in which nodes evaluate their neighbors based on their own goals, current state, present location, and network conditions.

Morvan and Sené [112] propose and evaluate a distributed protocol to manage trust diffusion in ad hoc networks. The protocol uses local information (own knowledge) and exchanged information with all other nodes (external knowledge) to calculate the trust level. Nodes store information about the interactions between all other nodes to build its own experience. They use an index to weight the recommendation of other nodes to minimize the effect of slanderer nodes.

In a previous work, Wang *et al.* [113] propose a trust-based incentive model on a self-policing mechanism to make nodes evaluate the trust of their neighbor. That approach uses direct observation and other neighbor's information. Later, they consider that the information derived from other neighbors might be spurious. Therefore, they add to the trust-based incentive model a trust evaluation method with a trust scaling factor in order to relieve the influence of fake information on the accuracy of trust value [114].

Liu and Issarny [115] propose a reputation model for ad hoc networks based on recording node's experience about its neighbors and the recommendation of other nodes. They define a context-dependent reputation in which nodes might have different reputations for different services. They propose two different services category: helpfulness reputation and recommendation reputation. They assign more weight to recent experiences using a fading factor. Thus, nodes that misbehave can be detected faster. The problem related to this approach is that a malicious node willing to pretend a good behavior to gain the confidence of its neighbors is also benefit.

Yan *et al.* [116] propose a security solution for ad hoc networks based on a trust model. They suggest using a linear function to calculate the trust according to a particular action. The function considers different factors that can affect the trust level, including intrusion black lists, previous experience statistics, and recommendations. Nonetheless, the influence of such factors on the trust evaluation is not defined. Although mentioning general trust concepts, the work focus on specific routing issues.

McGibney *et al.* [117] present a biological and social-based approach in which nodes keep a trust score for each other node of which it is aware and distributes these to its neighbors. Each service has a trust threshold associated, the more sensitive the service, the higher is the threshold. They define a decentralized dynamic system involving nodes, services and trust scores that help to quickly and reliably locate potential sources of attacks and their threat level.

Some authors present trust models specifically designed to work with a particular routing protocol. Komathy and Narayanasamy [62] add to AODV routing protocol a trust-based evolutionary game model to cope with selfish nodes. Rizvi *et al.* [118] introduce two extensions to the DSR routing protocol. The goal is to mitigate the effects of routing misbehavior. They use the watchdog and the pathrater. Using these two approaches, the proposed system will update the trust table of each node with the ranked values of other nodes.

Cooperation Of Nodes Fairness In Dynamic Ad-hoc NeTworks (CONFIDANT) [119] protocol is proposed by Buchegger and Le Boudec as an extension to reactive source-routing protocol such as DSR. It is a reputation-based protocol where the components interact with each other for monitoring, reporting, and establishing routes by avoiding misbehaving nodes. SORI [120] and CORE [121] are reputation-based mechanisms that monitor the behavior of neighbors concerning the routing process.

Balakrishnan *et al.* [122, 123, 124] present a Trust Integrated Cooperation Architecture which consists of an obligation-based cooperation model known as fellowship to defend against both flooding and packet drop attacks. In their architecture, fellowship enhances its security decisions through a trust model known as secure MANET routing with trust intrigue (SMRTI).

Kostoulas *et al.* [125] propose a decentralized trust model to improve reliable information dissemination in large-scale disasters. The proposed model includes a distributed recommendation scheme, incorporated into an existing membership maintenance service for ad-hoc networks. In addition, trust-based information is propagated through a nature-inspired activation spreading mechanism.

The main differences of our work from all the related work are that we consider the resource restrictions and topology dynamics of mobile ad hoc networks. Thus, in our model, the interactions among nodes are limited to direct neighbors only. This characteristic implies a significant lower energy consumption and processing to calculate the trust level for all the network and more memory space since node do not keep trust information for all nodes in the network. In addition, nodes exchange recommendations with direct neighbors exclusively. Consequently, we minimize the effect of false recommendations, as we explain in Chapter 4, and decrease the number of messages sent. Another important issue is the introduction of the concept of relationship maturity in our model. This new concept can improve the efficiency of the trust model in MANETS. At last, only a small number of the work analyzes the robustness of their trust model against liars, as we do.

2.5.3 Attacks against trust systems

Trust establishment mechanisms improve the efficiency and security of ad hoc networks. However, trust models present some specific vulnerabilities that can be exploited by malicious nodes. The main attacks against distributed trust systems are:

- bad mouthing attack: consists of nodes that send false recommendations about its neighbors. The slander attack occurs when a node send a false recommendation intended to injure another node's reputation. This attack can be aggravated by the collusion of malicious nodes;
- on-off attack: a malicious node can continuously change its behavior between good and bad in order to cause damage to the network without being detected. This is considered as a time domain attack;
- Sybil attack: an attacker can create fake identifications and pretend to be another node. In such situation, the attacker can misbehave and ruin the reputation of other nodes or it can send credible false recommendations;
- newcomer attack: a malicious node can repeatedly change its identification for a new one and pretend it is a new user. The goal of this attack is to erase all the past from malicious nodes, so they can continue to misbehave with a new identification, until it is detected;
- conflicting behavior attack: attackers can behave differently according to each neighbor, creating conflicting recommendations from well-behaving nodes.

The defense against Sibyl and newcomer attacks does not rely on the design of trust model, but on authentication and access control. Sun *et al.* [126] investigate the benefits of using trust models in distributed networks, the vulnerabilities in trust establishment methods, and the defense mechanisms. They develop defense techniques for different kinds of attacks against trust establishment methods. Effectiveness of the attacks and the defense is demonstrated using scenarios of securing routing protocols and detecting malicious nodes in MANETs.

In an attempt to minimize the influence of the bad mouthing attack, Buchegger and Le Boudec [127] present a distributed trust system for peer to peer and mobile ad-hoc networks. The proposed system can cope with false disseminated information. In their approach, every node maintains a reputation rating and a trust rating about all nodes they care about. Recommendations that differ largely from the current reputation rating are considered incompatible, and consequently, are ignored.

2.6 Résumé du Chapitre

Les communications sans fil dans le standard IEEE 802.11 doivent employer un des deux modes d'opération disponibles. En mode d'infrastructure, tous les noeuds communiquent par un point d'accès. Par conséquent, quand un noeud veut envoyer un message à un autre noeud dans le même réseau, le message est envoyé au point d'accès (AP) qui transmet le message au noeud destinataire. En mode ad hoc, cependant, il n'y a aucune infrastructure et les noeuds se communiquent directement entre eux. Les réseaux ad hoc de communication directe se composent de noeuds qui se communiquent strictement avec leurs voisins directs, soit, noeuds dans la portée radio. Dans ce genre de réseau il n'y a aucun besoin de routage. Les réseaux mobiles ad hoc multisaut se composent de noeuds qui se communiquent par un réseau radio multisaut de façon entièrement décentralisée. Ainsi, chaque noeud doit jouer le rôle à la fois de client et de routeur. Dans les réseaux ad hoc mobiles (Mobile Ad Hoc Networks - MANETs), la topologie du réseau est dynamique en raison de la mobilité des noeuds.

La nature de l'environnement sans fil et mobile rend les réseaux ad hoc vulnérables à différentes sortes d'attaques. Dans les réseaux ad hoc mobiles, le scénario est encore plus problematique. Les dispositifs portables ont des ressources très limitées en termes de puissance, traitement et mémoire. Ces contraintes imposent plusieurs restrictions aux solutions déjà connues, disponibles strictement pour les réseaux filaires. Fournir le même niveau de sécurité dans les réseaux ad hoc est difficile, à cause de leur nature distribuée, et aussi crucial de maintenir le réseau fonctionnel et à l'abri des noeuds malveillants. Le routage dans les réseaux ad hoc est un défi important. En raison du manque d'infrastructure et de mobilité, les protocoles de routage pour réseaux filaires ne s'adaptent pas aux réseaux ad hoc. Ainsi, un nouvel ensemble de protocoles de routage consacrés à ce genre de réseau a été proposé. Néanmoins, le processus de routage présente plusieurs faiblesses, à cause de la mobilité et du manque de centralisation. Des noeuds malveillants peuvent, ainsi, exécuter plusieurs types d'attaques comme : *Spoofing*, *Sybil*, *DoS*, faux messages de routage, entre autres. Quelques auteurs proposent des modifications ou des extensions aux protocoles de routage existants afin de sécuriser les réseaux ad hoc.

En raison du manque d'infrastructure, les protocoles et applications conçus pour les réseaux ad hoc sont basés sur la collaboration entre noeuds pour un bon fonctionnement. Bien que la collaboration des noeuds soit une prétention habituelle dans les réseaux ad hoc, il n'est pas aussi évident que la collaboration existe dans des réseaux réels. Des noeuds égoïstes peuvent décider de ne pas participer aux tâches du réseau pour économiser de l'énergie et des noeuds malveillants peuvent ne pas coopérer à fin de perturber le réseau. Ces deux comportements exercent le même effet sur la performance du réseau, soit, une faible efficacité et une haute consommation d'énergie.

Plusieurs travaux proposent des mécanismes pour stimuler la coopération entre les noeuds. Leur but est d'éviter les comportements égoïstes et malveillants afin de garantir la bonne implémentation des tâches de routage et d'envoi par tous les noeuds des réseaux. Ces mécanismes sont basés sur des systèmes de récompense/punition, de crédit, et de réputation. Néanmoins, toutes ces approches sont limitées à la collaboration des noeuds pour transmettre des informations à d'autres voisins. Nous sommes concernés par toutes sortes de mécanismes distribués et d'applications, telles que l'authentification, la distribution de clés, le contrôle d'accès, la gestion, etc. A cette fin, nous employons le concept de confiance.

La confiance peut être définie comme une entité de croyance au sujet d'une autre partie (une personne, une organisation ou un dispositif) basée sur un ensemble de règles bien établies et de ses attentes. Pour la sécurité du réseau, la notion de confiance correspond à un ensemble de croyance parmi les entités qui participent à divers protocoles. Les mécanismes de réputation sont quelques fois inexactement regardés comme des modèles de confiance. La réputation peut être considérée comme mesure collective de fidélité, dans le sens de la fiabilité, basé sur les recommandations ou les estimations des membres d'une communauté.

Le concept de confiance est appliqué à plusieurs domaines de l'informatique, particulièrement pour les réseaux informatiques. Les principales applications sont, notamment, systèmes de multi-agent, commerce électronique, systèmes de recommandation. Puisque les réseaux ad hoc se fondent sur le comportement collaboratif des noeuds pour fonctionner correctement, les noeuds doivent se faire confiance de façon à permettre des applications réparties, y compris le contrôle de routage et d'admission. Par conséquent, l'existence d'un modèle de confiance qui aide les noeuds à gérer les évidences de confiance est essentielle.

Plusieurs travaux proposent des modèles de confiance pour réseaux ad hoc. Ces modèles sont basés sur l'observation locale et l'information à distance. Ce qui différencie notre travail est le fait que nous considérons les restrictions de ressources et la dynamique de la topologie des réseaux mobiles ad hoc. Ainsi, dans notre modèle, les interactions parmi les noeuds sont limitées aux voisins directs. Une autre question importante est l'introduction du concept de maturité du rapport. Cette nouvelle notion peut améliorer l'efficacité du modèle de confiance dans les réseaux ad hoc mobiles - MANETs.

Chapter 3

The trust model

In our work, trust can be viewed in two different ways. The trust concept that relates to reliability, which is defined as the subjective probability by which an individual, A, expects that another individual, B, performs a given action on which its welfare depends. [52]. Trust can also relate to making a decision on depending in something or somebody in a specific situation even though negative consequences are possible. For example, in a multi-hop ad hoc network, shown in Figure 3.1, node A needs the collaboration of node B to communicate with node C, node B acts as an intermediate forwarding node. Nevertheless, node A knows that node B implements a selective forwarding scheme that drops, on average, half the packets it receives. Therefore, node A will not trust node B as a relay for communicating with node C, because node B is not a reliable relay.

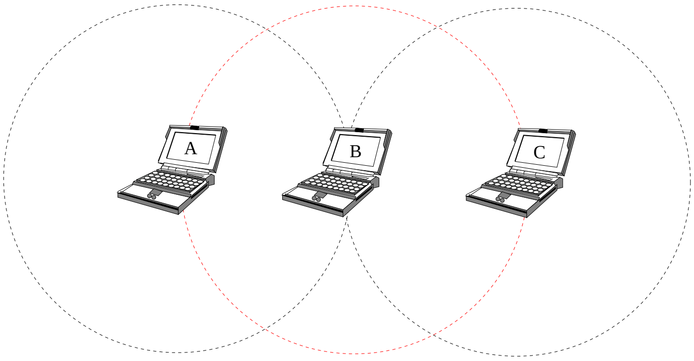


Figure 3.1: Two hops communication in ad hoc networks.

Using the same scenario, but in a different situation, where node A is in danger and needs to communicate with node C to ask for help. In this case, node A may decide to trust node B to serve as a relay. Although the reliability trust in node B as a relay is the same in both situations, the decision trust changes as a function of the comparatively different utility values associated with the different situations.

The basic idea consists of building a trust model that provides nodes with a mechanism to evaluate the trust level of its direct neighbors. The ability of assessing the trustworthy of its neighbors brings several advantages to nodes in ad hoc network. A node can predict its neighbors behavior which allows him to decide on which neighbor it is willing to depend to serve as relay. Nodes can use the trust information to choose the neighbors with which they are ready to collaborate, thus stimulating the cooperation among nodes. Nodes can learn based on information exchanged with trustworthy neighbors to build a knowledge plane [128, 129]. Moreover, nodes can use trust information to detect and isolate malicious behaviors. We do not guarantee a reliable environment but we aim at providing nodes means of knowing who they are dealing with. We expect nodes to use trust information to attempt achieving their goals.

Mui *et al.* define trust as the "subjective expectation an agent has about another's future behavior based on the history of their encounters" [65]. We extend this definition appending the recommendations of others. Therefore, similar to the concept of human trust, the computation of the trust level of a given neighbor is based on previous experiences and also on the opinion of other neighbors about this specific neighbor. By previous experiences, we mean that a node keeps track of the good and bad actions taken by other neighbors. As a result, previous experiences allow a node to have a personal "opinion" about each one of its neighbors. The Learning layer is responsible for monitoring and judging other's neighbor actions. Neighbor nodes can further share their own opinions in order to improve the trust level evaluation. The transmission of a personal opinion about a specific node i is defined as a recommendation. Neighbor nodes take into account this recommendation while calculating the local trust level for node i. For that purpose, we introduce the concept of relationship maturity, which is based on the age of the relationship between two nodes. This concept allows nodes to give more importance to recommendations sent by long-term neighbors rather new neighbors. Nodes willing to consider the recommendation of other nodes use the proposed Recommendation Exchange Protocol (REP) to keep the trust level of each neighbor up to date. It is important to mention that we assume the existence of an authentication mechanism, though we present a brief discussion on this subject.

In order to know how trustworthy a given neighbor is, each node assigns a so-called trust level for each direct neighbor. We propose a continuous representation for the trust level, ranging from 0 to 1 where 0 means the least reliable node and 1 means the most reliable node.

Our model can be divided in two distinct layers as shown is Figure 3.2. The Learning layer is responsible for gathering and converting information into knowledge. For instance, this layer is responsible for monitoring the behavior of each neighbor. The Trust layer then defines how to assess the trust level of each neighbor using the knowledge information provided by the Learning layer and the information exchanged with direct neighbors. Both layers can interact with all other layers. This means that the learning process considers information from all layers and that the trust information generated by the Trust layer is available for all other layers.

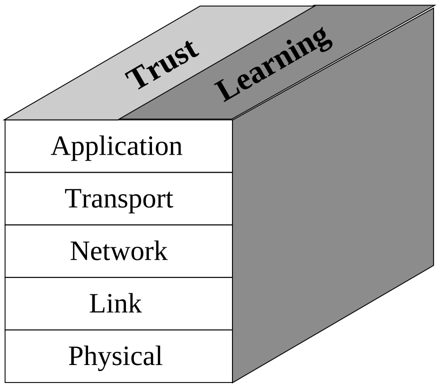


Figure 3.2: The proposed trust model architecture.

A trust value is associated to a particular scope, like forwarding packets, routing process, recommendation, and others application-specific scopes. Consequently, the type of information to be collected by the Learning layer depends on the defined scopes for the trust level. For instance, for routing process, the Learning layer must observe if neighbors respond to route requests, if they send false routes, etc.

The Learning layer relies on three basic components as displayed in Figure 3.3. The Behavior Monitor observes neighbors in order to collect information about their behavior. It must be able to notice other nodes actions and transmit to the Judge. Finally, the Behavior Monitor indicates the presence of new neighbors to the Recommendation Manager. The Judge is the component dedicated to reason about the information collected by the Monitor.

The Judge decides the quality of an action according to a previously defined classification. Then, the Judge sends its verdict to the Experience Calculator. The Experience Calculator estimates a trust value for a given node based on the information received by the Judge.

In this thesis, we focus on the Trust layer and we assume an imperfect Learning layer, which only perceives part of the true behavior of other nodes. The specification of the Learning layer is defined in Section 3.6.

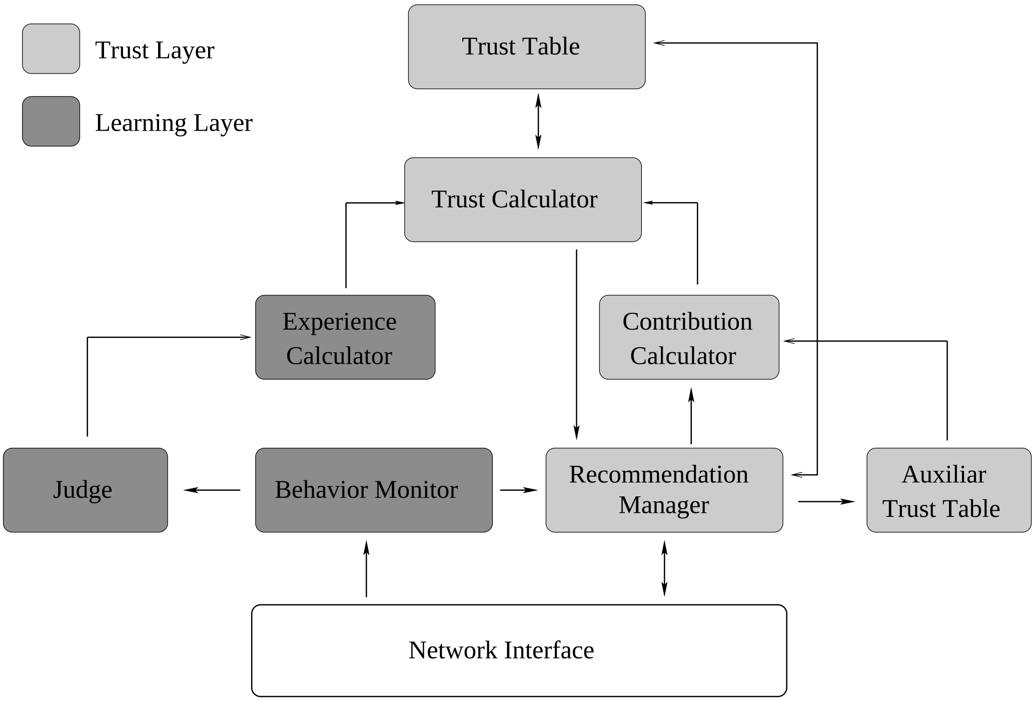


Figure 3.3: The proposed trust model components.

The Trust layer is composed of five main components as illustrates in Figure 3.3. Each node must keep a main Trust Table which contains the trust level for all its neighbors. All nodes that can directly communicate with each other are considered neighbors. Additionally, a node can also store the opinion of its neighbors about other nodes on the Trust Table whenever it is possible. Thus, the Trust Table is a matrix of order n^2, where n is the number of neighbors. Each entry on the Trust Table is associated to a timeout. Therefore, an entry is erased from the Trust Table whenever the node associated to that entry is no longer a direct neighbor or when it expires. All the recommendations related to that entry are erased as well. In our model, nodes can also keep an additional table that is not mandatory.

The Auxiliary Trust Table (ATT) contains the confidence in each trust level and for how long they keep that information (relationship maturity). The goal of the Auxiliary Trust Table is to supply nodes with additional information that improves the trust level evaluation. Nevertheless, this trust evaluation improvement requires more energy consumption and nodes with power or storage constraints can choose not to implement the entire trust system. Therefore, we define three operation modes: simple, intermediate, and advanced. Nodes with low power/storage capacity operate in the simple mode, in which they use just the main Trust Table and REP protocol is optional. Nodes with a medium capacity operate in the intermediate mode, which keeps also the recommendations of other nodes. In the advanced mode, nodes implement the whole trust system with all its features. In the rest of this thesis, we consider that nodes always operate in the advanced mode.

The Recommendation Manager is responsible for receiving, sending, and storing recommendations. The interactions between the Network Interface and the recommendation manager are performed by the Recommendation Exchange Protocol. The reception of a recommendation involves two actions. First, the recommendation is stored in the Auxiliary Trust Table and then it is forwarded to the Contribution Calculator component.

The Contribution Calculator computes all the recommendations for a given neighbor and determines a trust value based on the opinions of other nodes. This value is passed to the Trust Calculator component.

The Trust Calculator evaluates the trust level based on the trust values received from the Experience Calculator and the Contribution Calculator. The Trust Calculator also notifies the Recommendation Manager the need of sending a trust recommendation advertisement (Section 3.4).

The development of new mechanisms for ad hoc networks must take into account the power, processing, and memory restrictions presented by portable devices. In addition, the dynamic topology must not be neglected. Therefore, in our model, the interactions among nodes are limited to direct neighbors only. It means that neighbors do not keep trust information about every single node in the network, but just about direct neighbors. This characteristic implies significant lower energy consumption, less processing, and more memory space. In addition, nodes exchange recommendations with direct neighbors exclusively. Consequently, we minimize the effect of false recommendations in three ways. First, the number of received recommendations is significantly smaller. Second, recommendations are not forwarded, thus there is no intermediate node to increase the uncertainty of the information. Third, a node can always balance the recommendations with its own experiences to calculate the trust level because nodes do not calculate the trust level of neighbors that are not in direct contact. The decrease in the number of messages sent not only alleviates the network traffic, but also decreases the energy consumption.

In the next sections we present details regarding the Trust layer components.

3.1 Trust level evaluation

When a node first meets a new neighbor, it must assign an initial level of trust to this neighbor. This first value depends on the network condition, level of mobility, time, and place. Afterwards, the trust level evaluation process begins with a trust recommendation request and the monitoring of the new neighbor.

We define the trust level evaluation from node a about node b, $T_a(b)$, as a sum of its own trust and the contribution of other nodes, in the same way as defined by Virendra *et al.* [111]. The fundamental equation is

$$T_a(b) = (1 - \alpha)Q_a(b) + \alpha C_a(b), \tag{3.1}$$

where α permits choosing the most relevant factor of the equation. The variable $Q_a(b)$ represents the capability of a node to evaluate the trust level of their neighbors based on its own information and $C_a(b)$ is the contribution of the neighbors. The variable *alpha* is an important parameter in our model, which allows nodes to give more weight to one of the terms, as will be analyzed later in Chapter 4. In order to obtain the value for the capability of a node to evaluate the trust level of their neighbors based on its own information, $Q_a(b)$, we propose the following equation

$$Q_a(b) = \beta E_T + (1 - \beta)T_a(b), \tag{3.2}$$

where E_T represents the trust value obtained by the judgment of the actions of a neighbor, and the variable β allows to give different weights for the factors of the equation, selecting which factor is the more relevant at a given moment. The variable $T_a(b)$ in this equation gives the last trust level value saved in the Trust Table.

Equations 3.1 and 3.2 describe how the Trust Calculator combines the information from the Experience Calculator (E_T), the Contribution Calculator ($C_a(b)$), and the Trust Table ($T_a(b)$) to derive a trust level.

3.2 Contribution computation

The trust level calculation also considers the recommendation of direct neighbors. The set of recommendations is called contribution ($C_a(b)$ in Equation 3.1). Recommendations are obtained using the Recommendation Exchange Protocol.

The contribution, $C_a(b)$, is defined as the sum of the recommendations from all nodes $i \in K_a$ about node b weighted by the trust level of node a about node i, as follows:

$$C_a(b) = \frac{\sum_{i \in K_a} T_a(i) M_i(b) X_i(b)}{\sum_{j \in K_a} T_a(j) \sum_{j \in K_a} M_j(b)}. \tag{3.3}$$

The set of nodes K_a is composed by the group of nodes from which recommendations are considered. It is a subset of the neighbors of node a comprising all nodes that satisfy certain conditions. To increase the confidence of recommendations, a node can select neighbors whose trust level is above a certain threshold to accept recommendations. Let N_a be the set of neighbors of node a that includes all nodes that are known for a period of time longer than the relationship maturity threshold, M_{th}. The subset K_a is then defined as follows

$$K_a = \{\forall i \in N_a | T_a(i) \geq T_{th}\}. \tag{3.4}$$

The contribution considers not only the trust level of other nodes but also the accuracy and the relationship maturity. The accuracy of a trust level is defined by the standard deviation, similar to Theodorakopoulos and Baras [96]. The value in the Trust Table of node a regarding node b is associated to a standard deviation $\sigma_a(b)$, which refers to the variations of the trust level that node a has observed about node b. We use X as a random variable with a normal distribution to represent the uncertainty of the recommendation. It can be expressed as

$$X_i(b) = N(T_i(b), \sigma_i(b)). \tag{3.5}$$

The recommendation of node i about node b is weighted by $M_i(b)$, which defines the maturity of the relationship between nodes i and b, measured at node i. The relationship maturity is a measure of the time that two nodes have known each other. We use the relationship maturity to give more relevance to the nodes that know the evaluated neighbor for a long time. Accordingly, we assume that the trust level of a more mature neighbor has already converged to a common value within the network and therefore its opinion should be more relevant than the opinion of a new neighbor. It is important to notice that maturity is only considered between the recommender, node i, and the node that is being evaluated, node b, namely, node a will never judge the opinions from neighbors that it knows longer more relevant.

Malicious nodes can implement an attack exploiting the concept of relationship maturity by attributing fake trust levels. In order to minimize this effect, each node defines a maximum relationship maturity value M_{max}, which represents an upper bound for the relationship maturity. This value is based on the average time for which a node knows its neighbors.

3.3 The First Trust Assignment

Each node is responsible for its own trust computation. Thus, we divide the trust scheme in two distinct phases. An initial phase, when nodes first meet each other, in which they assign a trust level to each other. The second phase is the trust level update, which assumes that the nodes have already met each other.

When a node first meets a specific neighbor, it assigns an initial level of trust for this neighbor. The first trust assignment depends on several network parameters, such as mobility, node's location, and its current state. We classify the first trust assignment strategy as prudent or friendly/naive. In the prudent strategy the node does not trust strangers and considers that every new neighbor as a possible threat to the network. As a consequence, the node assigns a low value of trust for the new neighbor. On the other hand, the friendly/naive strategy assumes that every node is reliable until proven otherwise. In such case, the node associates a high level of trust for new neighbors. When a node adopts this strategy based on previous experience, we consider it friendly and if the node chooses this strategy due to lack of options it is considered naive. Right in the middle of these two strategies one could think of a moderate strategy, in which the node assigns an intermediate level of trust for strangers.

Different situations may demand distinct strategies. For example, if a node has already a significant number of reliable neighbors it can adopt a prudent strategy because it does not need new reliable neighbors. Further, the addition of a new neighbor may not significantly increase the probability of augmenting its satisfaction level. On the other hand, in a network where topology periodically changes and neighbor relationships are ephemeral, a node can opt for the naive strategy. In hostile environments, nodes can adopt the prudent strategy whereas in well-known cordial environments nodes can select the friendly strategy.

The first trust assignment occurs during the initial phase. The first trust level can also take into account the recommendation of known neighbors weighted by their trust levels. In order to a node a calculate the first trust level of node b, we propose the same approach as Equation 3.1, but replacing the term that reflects the own experience by the First Trust Value (F). Hence the first trust value is given by:

$$T_a(b) = (1 - \alpha)F_a + \alpha C_a(b), \tag{3.6}$$

where F_a is the value used by node a according to the adopted strategy, $C_a(b)$ is the contribution of the trust level of other nodes about node b, and α is the weight factor that allows us to give more relevance to the desired parameter.

Deciding the best strategy to derive F is not a simple task. For instance, nodes must take into account the level of mobility, the current satisfaction, the number of reliable neighbors. As choosing the best strategy evolves several parameters, we suggest a learning approach to select the strategy. This means that the Learning layer is responsible for selecting the best strategy.

3.4 The Recommendation Exchange Protocol

In the previous section, we presented how nodes evaluate the trust level based on their own experiences and on the recommendation of their neighbors. In this section, we propose a protocol that allows nodes to exchange recommendations among them.

The Recommendation Exchange Protocol (REP) is a part of the Recommender Manager in Figure 3.3 and includes three basic messages and is not mandatory for every node. Nodes choose whether to use it or not according to their current goals and constraints. Our proposal only considers interactions with direct neighbors. This restriction simplifies significantly the protocol.

- Trust Request (TREQ): this message is sent by a node to ask its neighbors to give their recommendation about a given node

- Trust Reply (TREP): this message is sent in response to a Trust Request message

- Trust Advertisement (TA): This message is an unsolicited recommendation, used to announce to other neighbors about a change in its trust evaluation about a specific node. This message is associated to a trust level update

When two nodes first meet they broadcast a TREQ to their direct neighbors. When using IP (Internet Protocol) to broadcast the message, the Time to Live (TTL) field is set to 1. Accordingly, this is a one hop broadcast that avoids flooding because TREQ is not forwarded by the neighbors. The neighbors receive the TREQ message and answer it with a TREP message after waiting for a random period of time t_{REP} to avoid collisions and to wait for receiving other TREQs. The TREP message contains the recommendation of a specific node. If the replying node receives more than one TREQ, it opts for sending different unicast messages for each TREQ or sending a broadcast message with all the requested recommendations. A node can set a TREP threshold under which it will not answer the TREQ. The threshold is based on the trust level of the requesting node. This strategy avoids denial of service attacks by non trustworthy nodes that can repeatedly send TREQ messages. Before sending a TREQ message, a node waits for a specific period of time t_{REQ} trying to gather the maximum number of new neighbors. After t_{REQ}, the node will request the recommendations of all the q new neighbors it has collected. Thus, instead of sending q TREQ messages it sends just one with q node IDs.

After sending a TREQ, the trust requesting node will wait for a specific timeout period to receive the TREPs from its neighbors. If a node does not receive any TREP, it ignores the recommendation of its neighbors by choosing $\alpha = 0$ in Equation 3.1.

During a trust level update, the Trust Level (TL) may change. Thus, the node compares the current trust level with the last recommendation sent. In order to avoid nodes to send Trust Advertisement messages after every change in the Trust Level, we defined the TA threshold as a minimum difference between the current TL and the last recommendation sent above which nodes must announce the new TL by sending a TA. Hence, the reception of a TA message does not necessarily imply a recalculation of the trust level.

The recommendation includes the trust level for a particular node, its accuracy and for how long they know each other. For a node that does not implement the Auxiliary Trust Table the recommendation includes just the trust level. It is worth mentioning that recommendation messages can be piggybacked in routing messages.

3.5 A simple authentication mechanism

In trust models an authentication mechanism is essential, because malicious nodes may pretend to be another node. However, our model does not require a sophisticated authentication mechanism. Nodes do not need to know nor recognize any other node *a priori*, namely, a node does not need to identify a new neighbor when it arrives. In our system, nodes must be able to identify neighbors that they already know. Since neighborhood in our trust model is limited to direct neighbors only, nodes must exchange identifiers when they first meet and keep a neighbor identifier during all the period they remain in the radio range of each other. Therefore, each node must have one identifier, which is an additional information on the packet header.

A simple solution is to use the MAC (Medium Access Control) address as the identifier to avoid the addition of information to the packet header. There are two approaches to prevent malicious nodes from pretending to be some other node in the neighborhood when using the MAC address as node identifier. The first possibility consists of using a tamper proof hardware to keep the MAC address out of reach of malicious nodes. This approach guarantees that malicious nodes cannot change their identifier. Tamper proof security modules are also proposed in several solutions for ad hoc networks [41].

Another technique consists of adapting a PKI (Public Key Infrastructure) model for the constraints of our trust system. In this approach, a node keeps its own public key. When a new neighbor arrives, nodes exchange the identifier and the public key. Then, they exchange a secret that is encrypted by the private key of each other. Therefore, when a node must send a recommendation to another neighbor, it encrypts the recommendation and the shared secret with the public key of the destination node. The advantage of this approach lies on the lack of a tamper proof hardware to keep the MAC address. On the other hand, malicious nodes can change their identifiers whenever they want. This is not a real problem in the proposed model if we consider that a malicious node cannot change its identifier to another MAC address that is already in use by another neighbor. The malicious node that has a low trust level and wants to change its identifier to erase its past and begin from the scratch does not represent a problem. For example, a routing protocol that uses the trust level as an additional metric can always consider the relationship maturity as well. Therefore using the maturity relationship can limit the power of short term relationships.

3.6 The trust model implementation

We have developed a simulator, which is specifically designed for our model, in order to evaluate and identify the main characteristics of the proposed model.

In ad hoc networks, nodes can perform several actions, like sending packets, forwarding packets, responding to routing messages, sending recommendations, among others. The set of performed actions define the node's behavior. Therefore, the Learning Layer monitors the neighbor actions trying to evaluate their behavior. In our home-made simulator, each node performs good actions and/or bad actions. Nodes perform actions according to an exponentially distributed variable. The kind of action that will be performed depends solely on the nature of the node. A node with a nature equals to 0.8 means that it performs eight good actions out of ten.

The nature of a node ranges from 0 to 1. Most trustworthy nodes have nature equals to 1 while nodes untrustworthy have nature equals to 0. The nature is used as a reference of the ideal global trust level that a node should receive by its neighbors. We use it here as a metric to evaluate how close the measured global trust level of a node actually gets from its nature.

We emulate the Behavior Monitor (Figure 3.3) by introducing in our simulator the concept of perception. The perception indicates the probability of noticing a certain action. Each Behavior Monitor presents its own perception. Therefore, a node with 0.6 of perception is able of noticing 60% of all the actions performed by its neighbors. Figure 3.4 illustrates the Learning layer components. The Behavior Monitor passes all the perceived actions to the Judge without knowing its nature. In our simulator, we consider the Judge as perfect, which means that the judgment of an action always matches with the original nature of the action. It is worth to mention that noticing and judging an action does not imply using promiscuous mode. We believe that a node should be able to decide whether it will use promiscuous mode or not based on its own constraints and needs. Thus, nodes may decide not to use promiscuous mode at the expense of having a lower perception. Finally, the judgments are transmitted to the Experience Calculator.

For the Experience Calculator, we propose a simple approach which consists of evaluating the trust value based on a set of the last i ($i \in \mathbb{N}$) perceived actions from the same neighbor. It implies the existence of a minimum number of actions (i_{min}) that a node must notice from each neighbor to be able of having an opinion about them, based on its own experience. It means that during the initial phase of first contact, nodes use just the recommendations of its neighbors to evaluate the trust level of the new one. The minimum number of perceived actions is crucial for the accuracy of the measure. The larger is this number, the more correct is the result. At the same time, a large number of actions leads to a longer delay for assessing the trust value for new neighbors. In the simulator, we define that the Experience Calculator uses the last 10 actions from a neighbor to estimate the trust value.

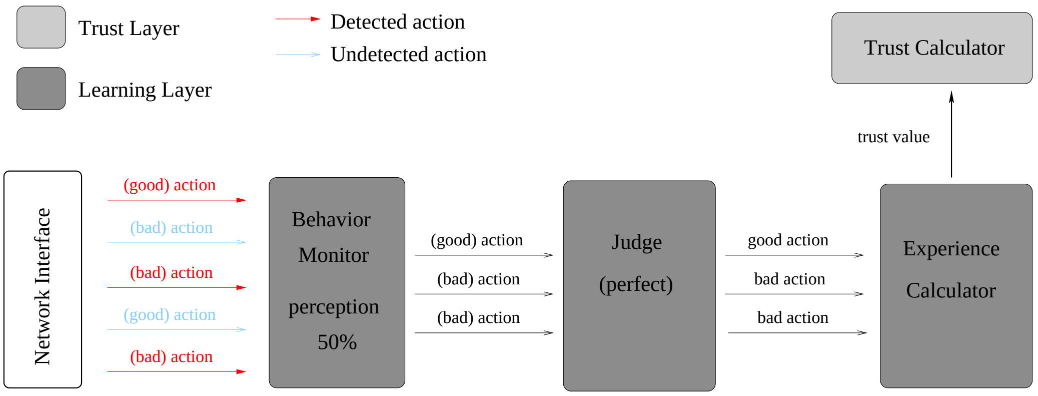


Figure 3.4: The proposed Trust Layer architecture.

There are three different possible status for a given neighbor. When nodes have not yet identified the existence of each other, they consider each other as an "unknown" neighbor. Nodes sense the presence of each other upon the arrival of a event. An event comprises the reception of a message or the perception of an action. From this moment on, neighbors are considered "acquaintance" until the first 10 actions are detected. Meanwhile, the trust level of an acquaintance is only based on the recommendations of neighbors.

Figure 3.5 displays the dynamics of our simulator in respect to the detection of actions. Basically, the sequence of detecting an action of a neighbor depends on the status of the neighbor. If it is a new neighbor, the neighbor status changes to acquaintance. While the neighbor remains an acquaintance, the node just collects the other actions. After noticing 10 actions from the same neighbor, it becomes a known neighbor and all future actions will trigger a trust update.

The trust update process will result in a new trust level, that might differ from the previous one. Therefore, if the difference between the new trust level and the old one is greater than TL threshold then, the node must send a TA message announcing its new value of TL for this neighbor. We consider that only actions provoke a node to send a TA.

The dynamics of the simulator in relation to the arrival of messages is shown in Figure 3.6. The reception of a message makes the node behave as defined by the Recommendation Exchange Protocol, explained in Section 3.4.

It can be seen that the reception of recommendations, through a TA or TREP messages, will trigger a trust update only if the difference between the TL in the recommendation and the previous TL stored in the Trust Table is greater than the TL threshold. Another ob-

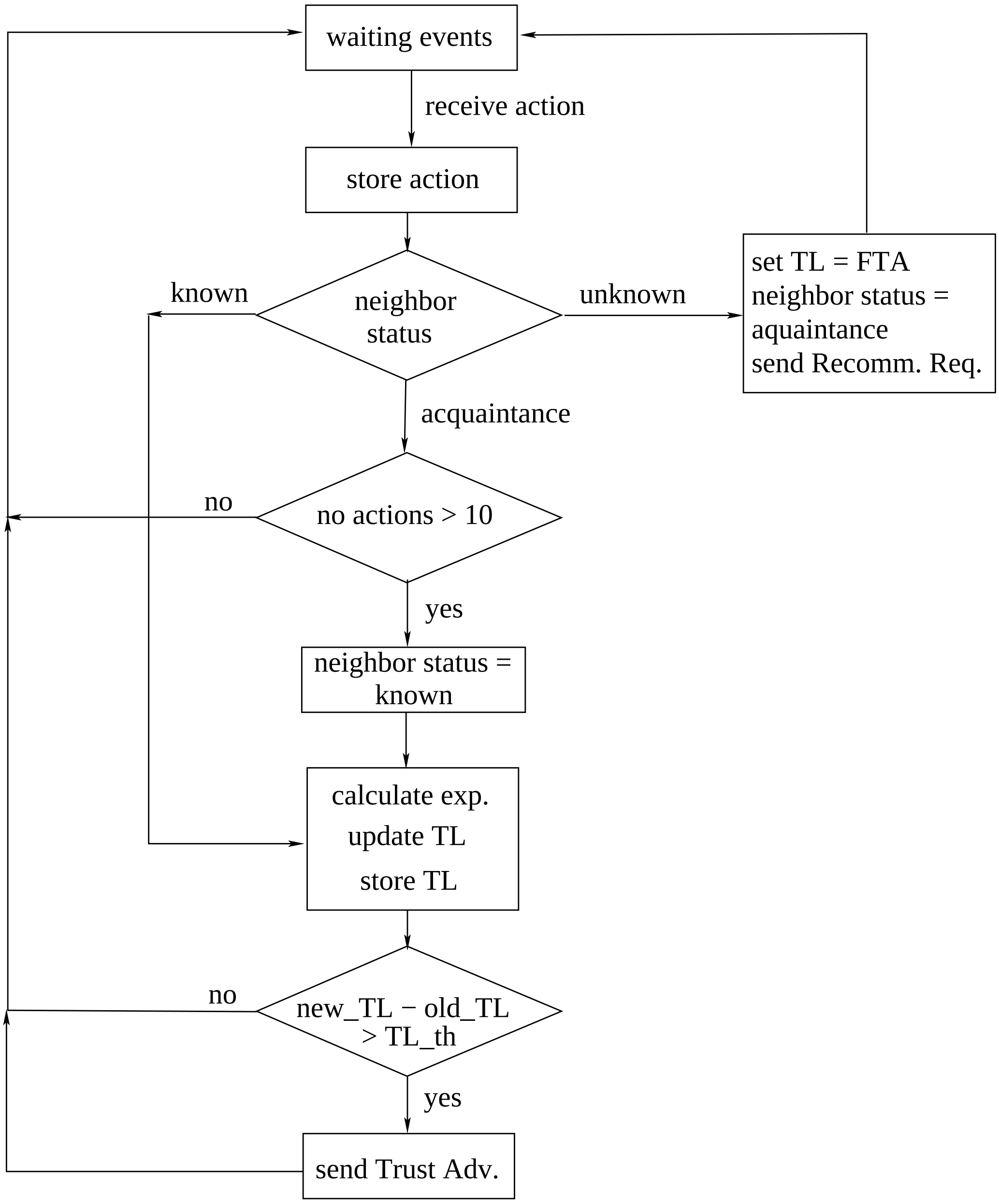


Figure 3.5: Simulator dynamics - detection of an action.

servation is that when the target node of the recommendation is an acquaintance neighbor, the *alpha* parameter is set to 1. It signifies that only the recommendations of other nodes are taken into account by the Trust Calculator (Figure 3.3) because there is not enough previous experiences to consider in the TL calculation.

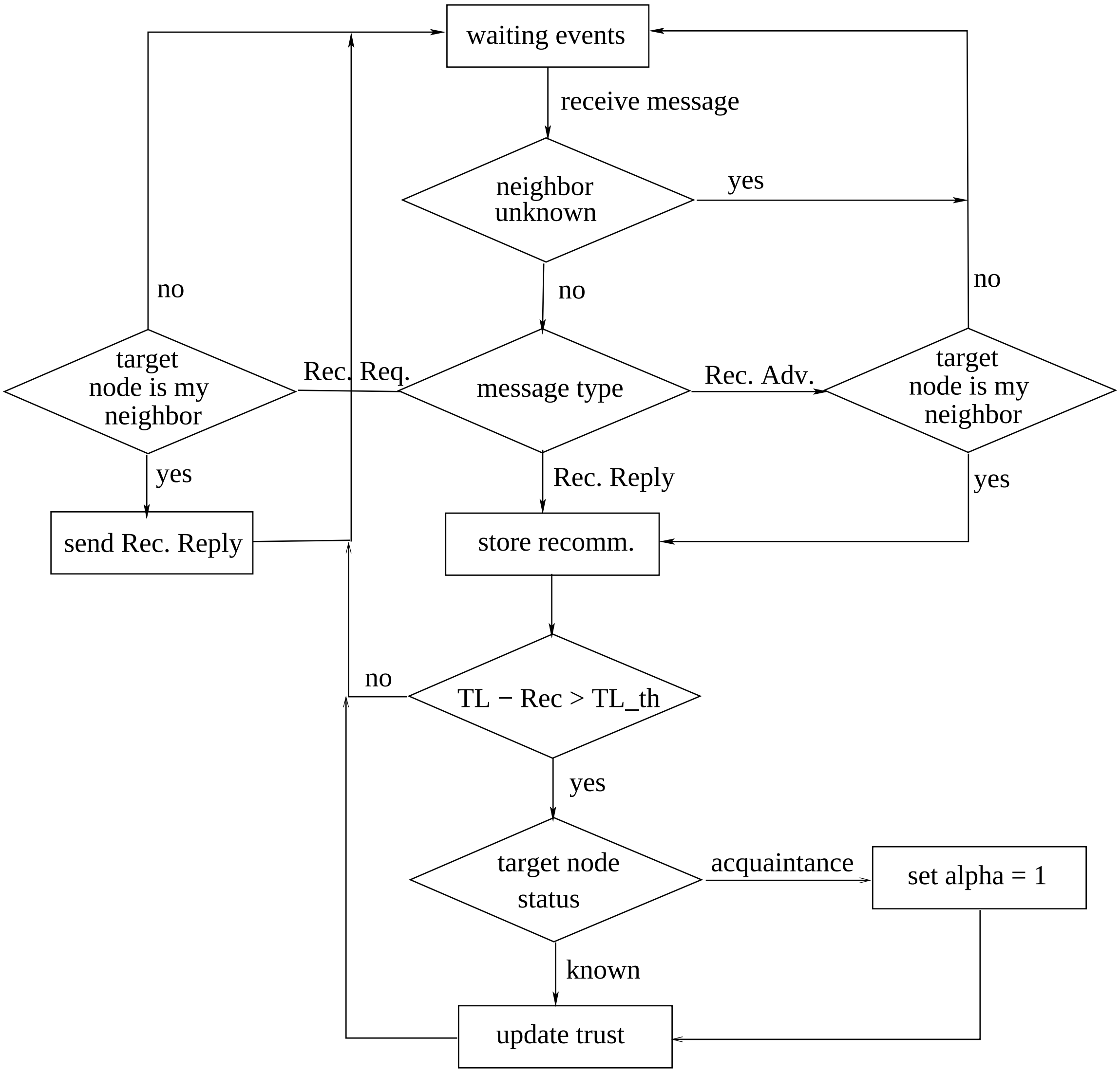


Figure 3.6: Simulator dynamics - reception of a message.

3.6.1 Mobility model

We implement a simple mobility model that allows nodes to move randomly or with specific destinations. It is possible to define a period of time in which nodes remain in the same position. In order to simplify the movement procedure, we implement a discrete mobility model. In our simulator, before moving towards a destination, a node calculates, according to its velocity and distance from the destination, how long it will take. Therefore, the node updates its position after this period of time, instead of updating after the occurrence of every event. This simplification does not compromise our results. First, for our simulations, this approach constitutes a worst case scenario. Second, the time a node maintains contact with new neighbors during the movement is not enough to form a trust level about them, as shown in Chapter 4.

3.7 Résumé du Chapitre

Dans notre travail, la confiance peut être vue de deux manières différentes. D'abord, le concept de confiance relationné à la fiabilité et, deuxièmement, à la prise de décision par rapport aux actions de quelqu'un ou quelque chose dans une situation spécifique, quoique des conséquences négatives soient possibles.

L'idée fondamentale a été de construire un modèle de confiance qui fournit aux noeuds un mécanisme pour évaluer le niveau de confiance de ses voisins directs. La capacité d'évaluer si ses voisins sont dignes de confiance apporte plusieurs avantages aux noeuds dans un réseau ad hoc, puisqu'il peut prévoir le comportement de ses voisins et, ainsi, décider quel voisin lui servira de relais. Les noeuds peuvent employer l'information de confiance pour choisir les voisins avec lesquels ils sont prêts à collaborer, stimulant la coopération entre eux. Ils peuvent aussi apprendre basé sur les informations échangées avec ses voisins dignes de confiance pour construire un plan de connaissance. Additionnellement, les noeuds peuvent employer des informations de confiance pour détecter et isoler des comportements malveillants. Nous ne pouvons pas garantir un environnement fiable mais nous visons à fournir des moyens pour que les noeuds puissent savoir avec qui ils traitent. Semblable au concept de la confiance humaine, le calcul du niveau de confiance d'un voisin donné est basé sur des expériences précédentes et également sur des recommandations d'autres voisins au sujet de ce voisin spécifique. Nous présentons un nouveau paramètre, celui du rapport de maturité, qui définit la maturité du rapport entre deux noeuds. Ce paramètre est une mesure du temps que deux noeuds se connaissent et nous l'employons pour donner plus d'importance aux noeuds qui connaissent leur voisin depuis longtemps. En conséquence, nous supposons que le niveau de confiance d'un voisin plus mûr ait déjà convergé à une valeur commune dans le réseau et donc son avis devrait être plus valorisé que l'opinion d'un nouveau voisin. Il est important de mentionner que nous assumons l'existence d'un mécanisme d'authentification

Notre modèle peut être divisé en deux couches distinctes. La Couche d'apprentissage est responsable par recueillir et convertir des informations en connaissance; elle est responsable, par exemple, par la surveillance du comportement de chaque voisin. La Couche de confiance est, alors, responsable par définir comment évaluer le niveau de confiance de chaque voisin employant l'information de connaissance fournie par la Couche d'apprentissage et par l'information échangée avec les voisins directs. Les deux couches peuvent interagir avec toutes les couches du modèle de TCP/IP. Une valeur de confiance est associée à un context particulière, comme l'expédition des paquets, le processus de routage, l'envoi des recommandations entre autres. En conséquence, le type d'information à rassembler par la Couche d'apprentissage dépend des contextes définis pour le niveau de confiance. Pour le processus de routage, par exemple, la couche d'apprentissage doit observer si les voisins répondent aux demandes d'itinéraire, s'ils envoient des faux itinéraires, etc.

Dans notre modèle, les interactions parmi les noeuds sont limitées aux voisins directs. Cette caractéristique implique une consommation d'énergie et de traitement significativement moindre et à une économie d'espace mémoire. En outre, les noeuds échangent des recommandations exclusivement avec leur voisin direct. Conséquemment, nous réduisons l'effet des fausses recommandations de trois manières. D'abord, le nombre de recommandations reçues est sensiblement moindre. En second lieu, les recommandations ne sont pas expédiées, ainsi il n'y a aucun noeud intermédiaire pour augmenter l'incertitude de l'information. Troisièmement, un noeud peut toujours contrebalancer les recommandations avec ses propres expériences pour calculer le niveau de confiance. La diminution du nombre de messages envoyés peut non seulement alléger le trafic de réseau, mais aussi diminuer la consommation d'énergie.

Nous définissons une première valeur de confiance comme niveau de confiance assigné aux noeuds quand ils font connaissance. Cette valeur dépend de la mobilité, de la localisation du noeud et des conditions du réseau. Nous proposons un protocole qui permet aux noeuds d'envoyer et de recevoir des recommandations de ses voisins. Le Protocole d'Echange de Recommandation (Recommendation Exchange Protocol REP) inclut trois messages de base et n'est pas obligatoire pour tout noeud. Les noeuds choisissent de l'employer ou pas selon leurs buts et contraintes actuels. Le message de Demande de Confiance (Trust Request - TREQ) est envoyé par un noeud pour demander la recommandation de ses voisins. Le message de Réponse de Confiance (Trust Reply - TREP) est envoyé en réponse à un message TREQ. Le message d'Annonce de Confiance (Trust Advertisement - TA) est une recommandation non sollicitée utilisée pour alerter ses voisins d'un changement de son évaluation de confiance.

Dans le simulateur, spécifiquement conçu pour notre modèle, nous définissons le concept d'Action pour émuler des opérations ordinaires exécuté par un noeud d'un réseau ad hoc, comme envoyer des paquets, expédier des paquets, répondre à des messages de routage, envoyer des recommandations, entre autres. Dans notre simulateur, les actions sont classées en tant que bonnes ou mauvaises actions. Les noeuds effectuent des actions selon une variable distribuée exponentielle. Le type d'action qui sera réalisé dépend strictement de la nature du noeud. Qu'il possède une nature égale à 0.8 signifie qu'il effectue huit bonnes actions sur dix. La nature d'un noeud s'étend de 0 à 1. Les noeuds les plus fiables ont une nature égale à 1, tandis que les noeuds moins fiables ont une nature équivalente à 0. La nature est employée comme une référence du niveau de confiance global idéal qu'un noeud devrait recevoir de ses voisins. Nous émulons la Couche d'apprentissage en présentant dans notre simulateur le concept de Perception. La perception indique la probabilité de qu'une certaine action soit notée. Chaque Moniteur de Comportement a sa propre perception. Par conséquent, un noeud avec 0.6 de perception est capable de noter 60% de toutes les actions effectuées par ses voisins.

Chapter 4

The Simulation Results

In this chapter, we present the results of the experiments. First, we expose the results that demonstrate the correctness of our model and the impact of the main parameters on the trust evaluation process. The main characteristics of trust dynamics in an ad hoc networks are emphasized. Then, we evaluate our model in mobile multihop ad hoc networks. We show the effectiveness of the relationship maturity parameter and how the other parameters are tuned to improve the trust evaluation in mobile scenarios. The last results assess the robustness of our model to slander attacks. We take into account the presence of malicious nodes lying about their recommendations. In the first scenario, we consider that malicious nodes collude to hide from other nodes the behavior change of another malicious node. The second scenario considers nodes that collude to slander one of its neighbors, namely, nodes sending false recommendations to depreciate the reputation of other neighbors.

The most relevant simulation parameters are presented separately for each experiment. All the other parameters, which are common for all three experiments are available in Appendix A. All results are presented with a confidence interval of 95%.

4.1 Single Hop Networks

Our main goal in this experiment is to evaluate and analyze the influence of the number of neighbors, the first trust assignment strategy, and the variation of parameters α and perception on the trust evaluation process. The reason for only analyzing single hop networks in this experiment is to isolate all the problems related to multihop networks and focus strictly on the dynamics of our model.

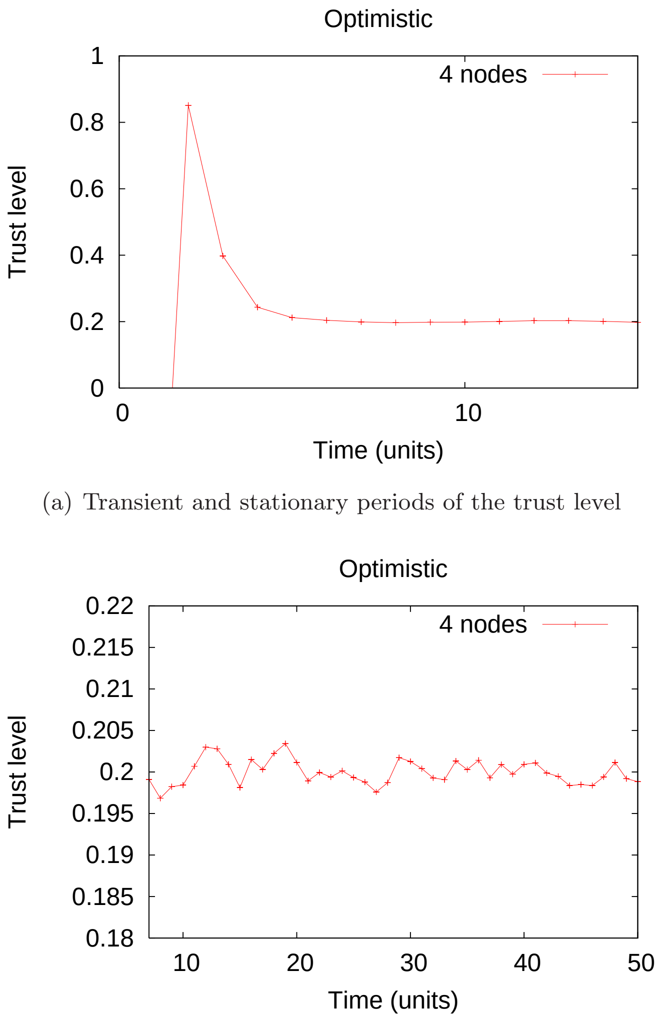
The simulation scenario consists of 16 nodes with 250 m transmission range, which are randomly placed in a 150 m × 150 m area. Under these circumstances, all nodes communicate directly to each other, characterizing a single hop ad hoc network. This scenario makes easier the evaluation of the effect of the basic parameters as already mentioned. All nodes operate in the advanced mode, which means that they implement all the features of the proposed system, as described in Chapter 3. We defined three values for the first trust assignment: 0.1 for the prudent, 0.5 for the moderate, and 0.9 for the friendly/naive strategy, also called optimistic strategy. All nodes adopt the same strategy. We also chose $\alpha = \beta = perception = 0.5$. These are the standard values for the simulations. For each specific configuration, the parameters that differ from these standard values are outlined. At last, in each configuration, all nodes have the same nature.

Figure 4.1 presents the time response of the average trust level from all neighbors about a specific node. In this specific scenario, the nature of nodes is set to 0.2 and the simulation time is 6,000 seconds. We observe in Figure 4.1(a) that the trust level value begins in a certain level but tends to the expected trust level. The expected (correct) level is the nature of the node that is being analyzed. After a specific amount of time $t_1 \approx 5$ time units, the curve oscillates around the correct value. Thus, we verify the existence of a transient period and stationary period. In the transient period (Figure 4.1(a)), nodes are trying to approximate to the expected value, while in the stationary period, the trust level is almost stable, very close to the correct value, varying like a smoothed sine function.

In the other figures, instead of presenting the average trust level, we present the average error for the trust value evaluated, that is, the difference between the trust level and the correct value. At the end, the ideal result is a curve that reaches the zero value, which means that there is no error between the average trust values calculated by the neighbors and the value of the nature of the node.

In Figure 4.2, nodes adopt an optimistic strategy and we vary the number of neighbors. The nature is set to 0,2. We can notice that the greater the number of neighbors the closer to zero the error. It occurs because increasing the number of neighbors results in an increase of the number of recommendations, which implies a greater probability of receiving recommendations closer to the correct value.

Figure 4.3 shows the influence of the parameter *alpha* on the trust level evaluation. Decreasing *alpha* implies that the contribution of other nodes has a minor effect in the trust level calculation. We observe from Figure 4.3(a) that the convergence to the correct value is slower for a higher value of α, namely, the transient is longer. Therefore, although the global



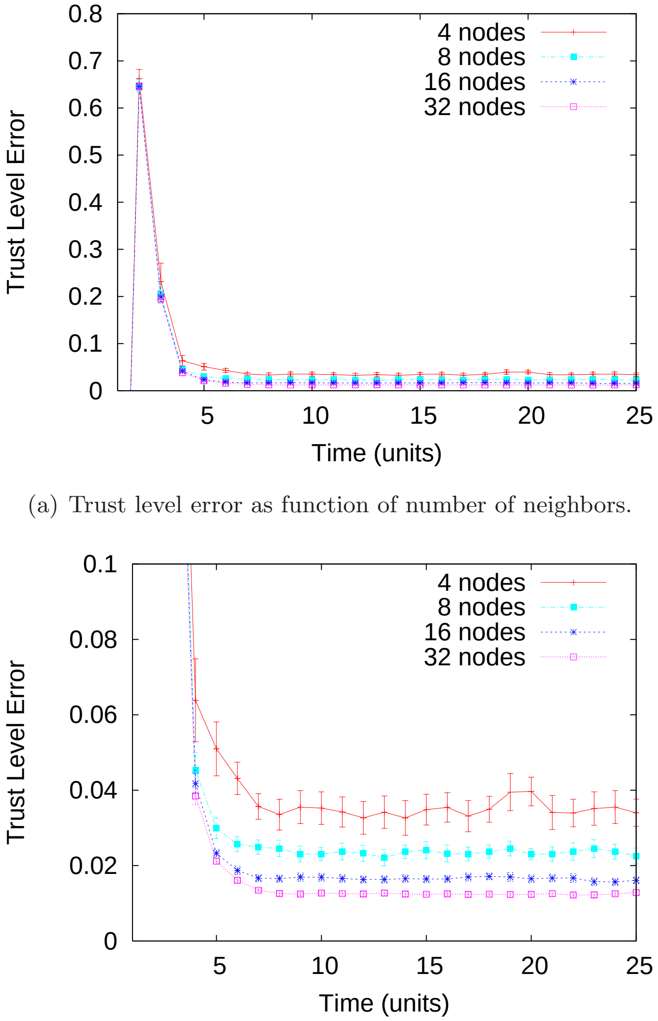
(a) Transient and stationary periods of the trust level

(b) Zoom of the stationary period of the trust level

Figure 4.1: Behavior of the trust level during simulation.

opinion about a specific node changes slower when *alpha* is larger, the convergence value is closer to the expected one and presents a smaller variation, as shown by Figure 4.3(b).

The perception is the fraction of actions a node can notice from its neighbors. Figure 4.4 shows the impact of the perception on the trust level evaluation. It is clear that the perception is strong related to the duration of the transient period. It occurs because a node requires a minimum number of actions from each neighbor to consider its own experiences. If we increase the number of actions a node must notice before judging the



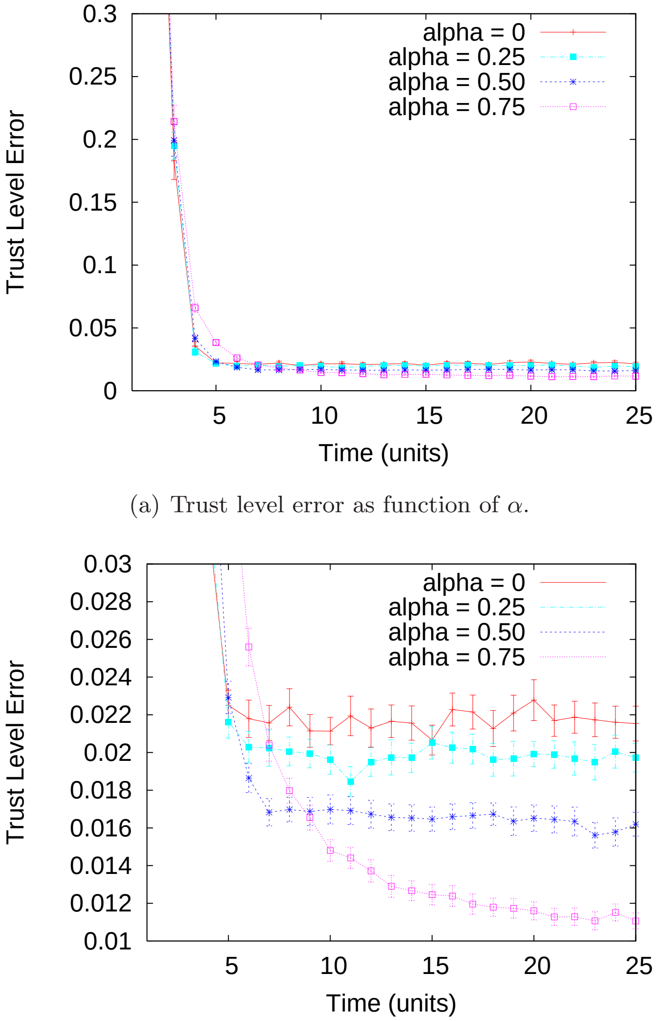
(a) Trust level error as function of number of neighbors.

(b) Zoom of trust level error as function of number of neighbors

Figure 4.2: Influence of the number of neighbors on the error of the trust value evaluation.

nature of a neighbor, it will increase the precision of the judgment, but it will also increase the transient period.

Afterwards, the perception is set to 0.2, varying the number of nodes. Figure 4.5 reveals the significance of a large number of neighbors to reach closer to the expected value for a low perception. It means that the lower is the perception, the lower is the probability of noticing the real nature of a neighbor by the judgment of its actions. Nevertheless, a low perception can be compensated by a larger number of neighbors.



(a) Trust level error as function of α.

(b) Zoom of the trust level error as function of α.

Figure 4.3: The influence of α on the transient period and on the error value.

At last, Figures 4.6 and 4.7 presents the influence of the nature on the trust level evaluation. We aim at analyzing the impact of different nature values in the converging delay. For this purpose, we set the strategy to optimistic (Figure 4.6) and moderate (Figura 4.7), varying the nature. The optimistic strategy assigns trust values of 0.9 for new neighbors and the moderate one assigns 0.5. We assume three nature values for nodes: node of "good" nature with trust level value of 0.8, node of regular nature with trust level value of 0.5, and node of evil nature with trust level value of 0.2. We can observe, from Figure 4.6,

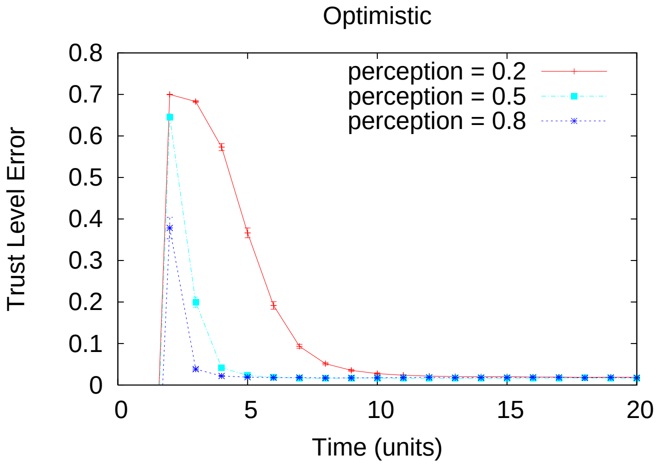


Figure 4.4: The influence of perception on the time required to attain the stationary period.

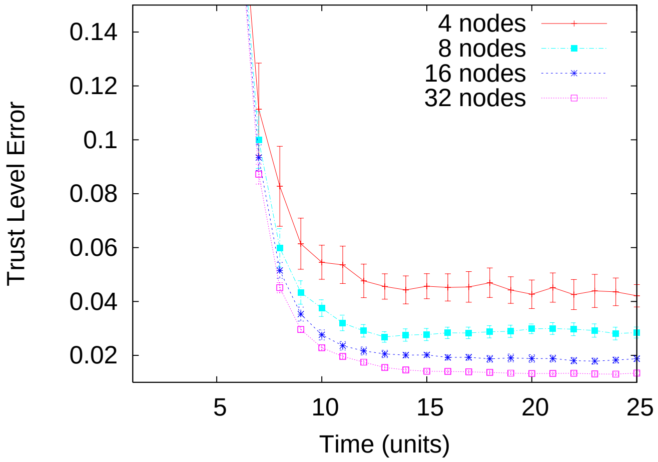


Figure 4.5: Time required for convergence as function of the number of neighbors for nodes with a low perception.

that the nature does not affect significantly the duration of the transient, only the peak, according to the chosen strategy. On the other hand, Figure 4.7 shows that it is easier for nodes to find the correct value when the nature is in the extremities. It happens because when a node produces the same amount of good and bad actions, the probability of sensing the exact proportion of good and bad actions decreases, considering that perception is less than 1.0.

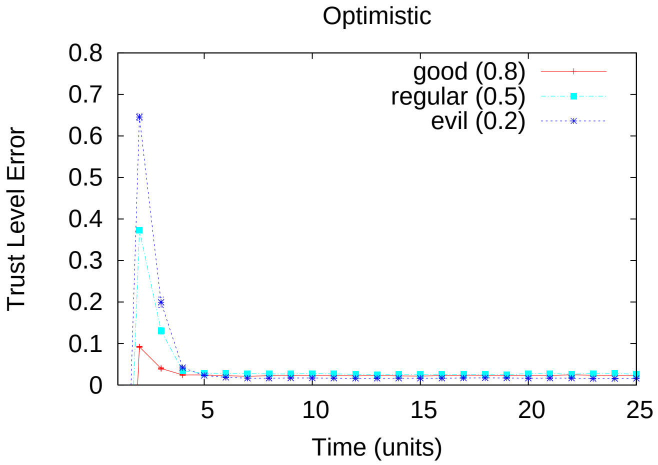


Figure 4.6: The influence of the nature using a optimistic strategy.

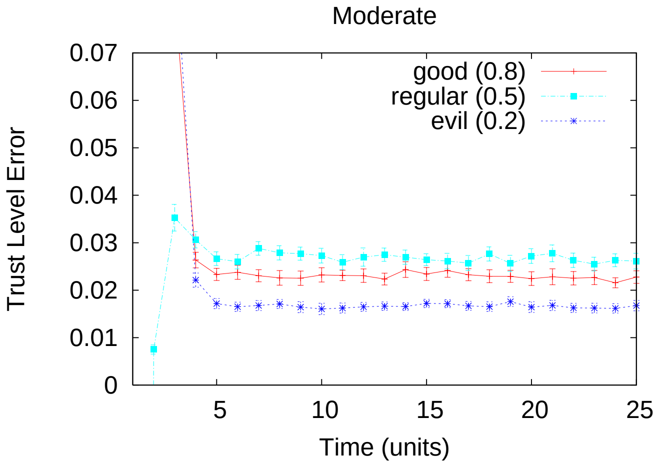


Figure 4.7: The influence of the nature using a moderate strategy.

4.2 Multihop Networks

Our main goal with this experiment is to evaluate the trust system performance in mobile multihop networks. We are also interested in analyzing the impact of the relationship maturity and the influence of the variation of parameters α and perception. All figures present the trust level error (TLE) in time, which stands for the difference between the evaluated trust level and the correct value during the simulation time. The correct value is given by the nature of the node. In an ideal trust system the TLE would reach zero, which means that the node was able to perfectly evaluate the character of its neighbors.

The simulation scenario consists of 21 nodes with 250 m transmission range, which are placed in a 1000 m × 400 m area, as shown in Figure 4.8. The distance between nodes is 150 m. We defined the first trust assignment equals to 0.9 for every node in the simulation. We also chose $\alpha = \beta = perception = 0.5$. These are the standard values for the simulations. For each specific configuration, the parameters that differ from its standard values are outlined. At last, in each configuration, all nodes have the same nature equals to 0.2.

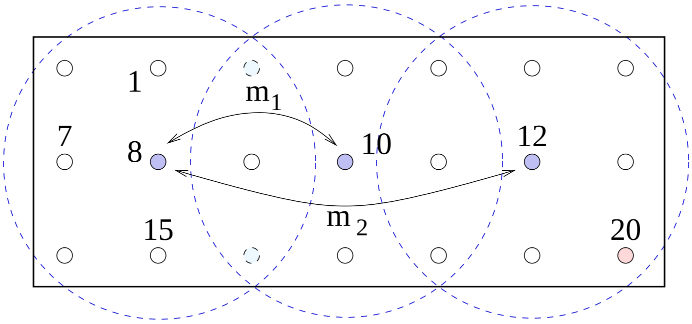
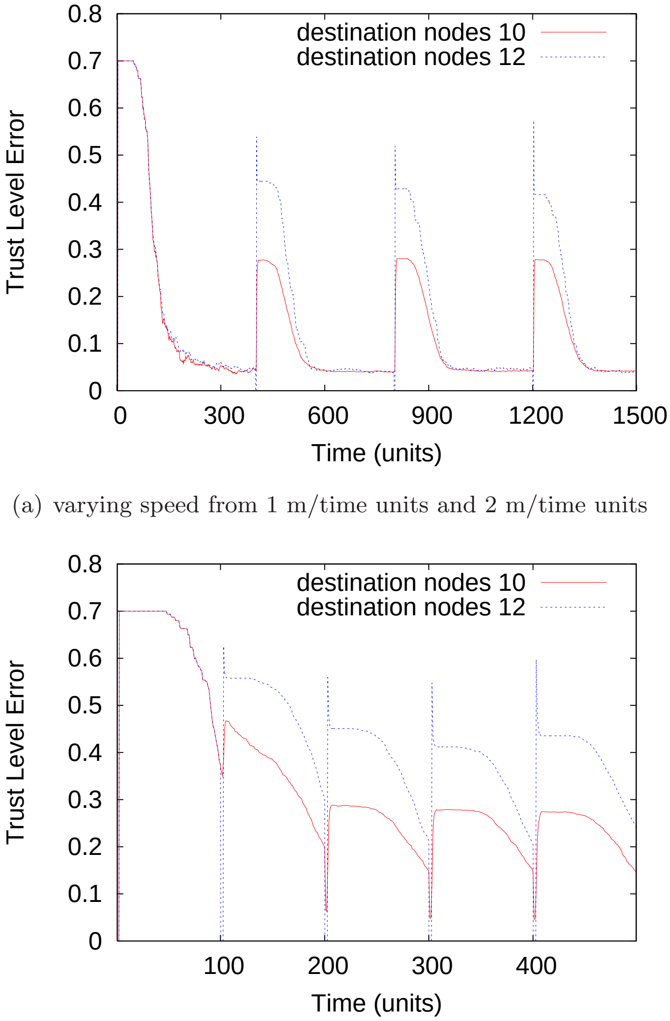


Figure 4.8: The experiment scenario.

In the first configuration, node 8 moves away to a specific place and then, after a pause, comes back to its origin, and it goes back and forth during all simulation. Figure 4.9 presents the average TLE for all neighbors of node 8. The lower curve shows the result when node 8 goes to the same place as node 10 (m_1 in Figure 4.8) and the other one when it moves to node 12 (m_2). The main difference is the number of new neighbors. In the shorter movement (m_1), node 8 keeps 3 old neighbors while in m_2 all neighbors are new ones. We set the speed equals to 1 m/time units and 2 m/time units respectively. Thus, node 8 takes the same amount of time to move to both destinations. We observe in Figure 4.9(a) that the TLE begins in a certain level, tends to zero, but never reaches it.

As we have noticed for single hop networks in Section 4.1, there is a transient period and a stationary period. In the transient period nodes are trying to approximate to the expected value, while in the stationary period, the trust level is almost stable, very close to the correct value. When the simulation starts and nobody knows each other, we notice a certain transient, because the first trust assignment is 0.9 and the nature of all nodes



(a) varying speed from 1 m/time units and 2 m/time units

(b) varying speed from 3 m/time units and 6 m/time units without pause

Figure 4.9: TLE in the presence of mobility with different velocities.

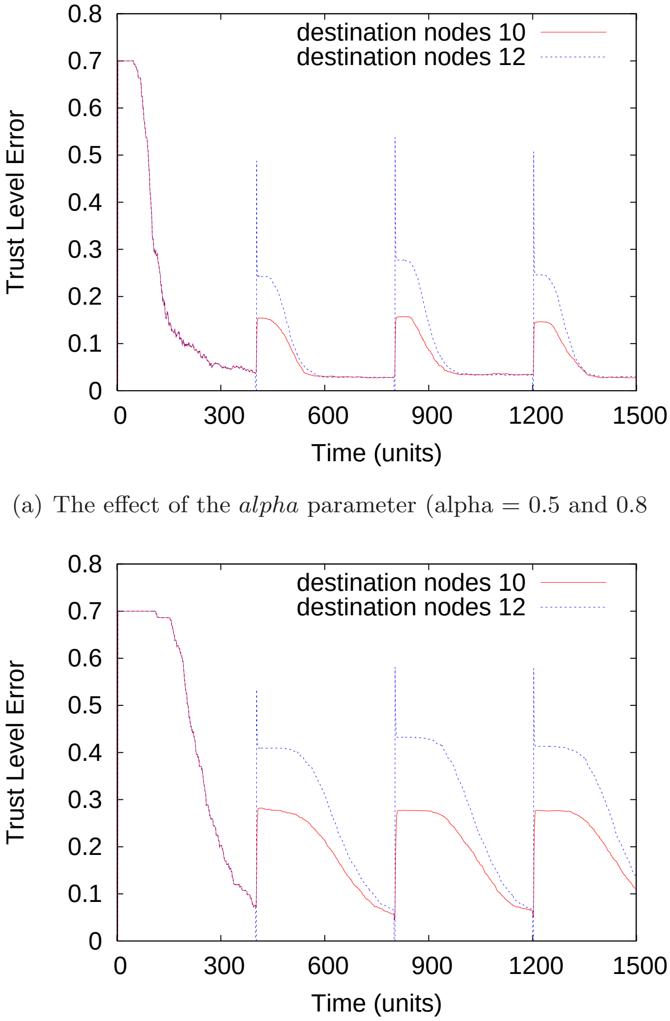
is equal to 0.2. These initial values were chosen as worst case parameters. When node 8 moves away and meets new neighbors, we observe a peak in the TLE. This peak is lower than the first one, because node 8 receives, since its arrival at the new destination, "correct" recommendations from its new neighbors since they already know their old neighbors. The difference between the two curves is the destination place. The lower one represents a situation in which node 8 moves to a place where it already knows 3 neighbors, while in the other one, it moves to a place where it knows nobody. Figure 4.9(b) shows the exact scenario but node 8 moves three times faster. It is clear that in these conditions, node 8 does not stay long enough to evaluate the trust level of its neighbors.

Figure 4.10 presents the results from the same movement pattern previously described (speed = 1 m/time units and 2 m/time units), and we vary the *alpha* parameter and the perception of node 8. We can notice that increasing *alpha*, from 0.5 to 0.8 (Figure 4.10(a)), implies a decrease in the TLE during the transient period if compared with Figure 4.9(a). Figure 4.10(b) shows that if node 8, the one that moves, has a lower perception (0.2 instead of 0.5), it takes longer to reach the nature of its neighbors, which restrains the mobility.

4.2.1 Relationship maturity

Afterwards, we analyze the impact of the relationship maturity in the evaluation of the trust level. For this purpose, we use a new configuration in the same scenario of Figure 4.8. In the new configuration, nodes 1, 8, 15 are going to move to the same place as node 12. Instead of monitoring the trust level of all neighbors of node 8, we consider the trust level evaluation of node 8 about node 7 and node 20. Therefore, when node 8 arrives at the destination, nodes 1 and 15 have just arrived there. It means that node 20 has 3 new neighbors and 3 old ones. The old ones have a better idea about the nature of node 20 than the new ones. Without the relationship maturity, when node 8 receives the recommendations of its neighbors, it will treat them all the same manner. Using the relationship maturity allows node 8 to give more importance to the recommendations of the oldest neighbors of node 20. The result can be seen in Figure 4.11. It can be noticed by Figure 4.11(a) that using the relationship maturity the transient is shorter. Figure 4.11(b) shows that with a greater *alpha* the impact of the relationship maturity in the transient is more significant. It improves the efficiency of the system due to the fact that node 8 prioritizes the recommendations of its neighbors.

Figure 4.12(a) displays the impact of the relationship maturity when node 8 has a lower perception (0.2). We can observe that it presents a lower peak when node 8 arrives at the destination, but the difference is not significant as in the other figures. In this case, node 8 has a longer transient caused by the lower perception. It happens because trust updates are triggered only by actions, thus a difficulty of perception implies a longer transient. In Figure 4.12(b) we decreased the perception of node 8 and increased the value of *alpha*. It indicates that this is a good combination for a mobile network. Moreover, the effect of

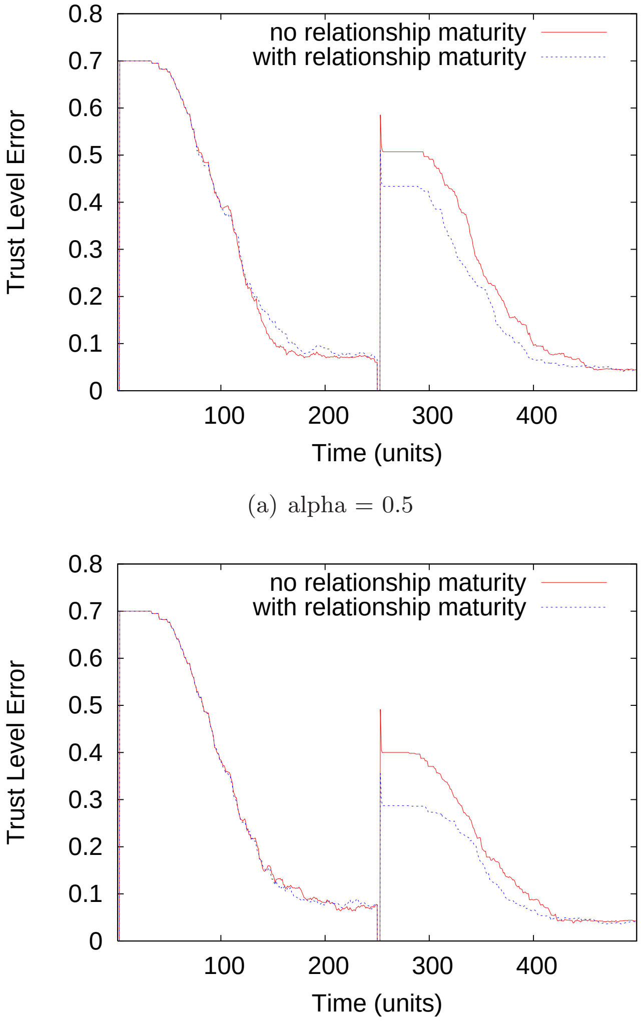


(a) The effect of the *alpha* parameter (alpha = 0.5 and 0.8

(b) The effect of perception (perception = 0.2 and 0.5)

Figure 4.10: TLE in the presence of mobility.

the relationship maturity is more evident. In this case when nodes have some difficulty to notice the actions of its neighbors, expressed by the low perception, the recommendations have greater importance. Therefore, valuing the recommendations from nodes that have a longer relationship with the node being evaluated is more effective. Although node 8 is not able to reach the stationary period, it achieves a lower TLE than without using the relationship maturity. The relationship maturity will play a more important role when nodes start to change their nature.



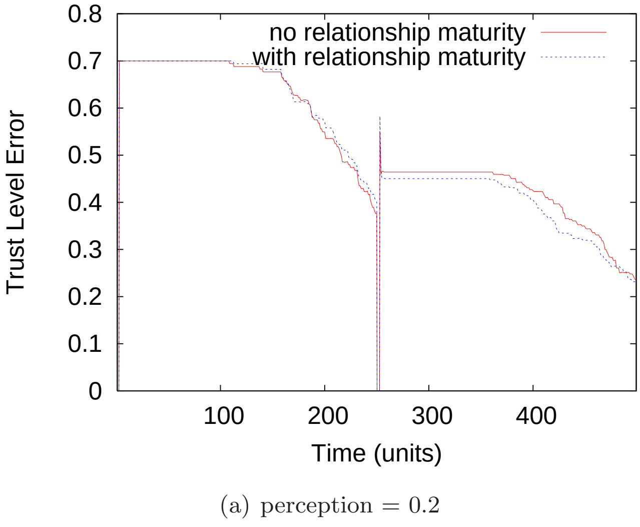
(a) alpha = 0.5

(b) alpha = 0.8

Figure 4.11: The impact of the relationship maturity.

4.3 Lying Attacks

The objective of this experiment is to evaluate the trust system performance under slander and collusion attacks in single-hop ad hoc networks. Mundinger and Le Boudec [130] perform an analysis of a reputation system for mobile ad hoc networks in the presence of liars. They conclude that there is a threshold proportion of lying nodes above which the reputations system cannot work. Below this threshold, liars do not cause a significant impact on the system. Therefore, we aim at finding that threshold above which the trust



(b) perception = 0.2 and alpha = 0.8

Figure 4.12: The impact of the relationship maturity.

model fails to work properly. All figures present the trust evaluation of node 2 about node 1. It means that node 2 is trying to assess the trust level of node 1.

We defined the first trust assignment equal to 0.9 for every node. The first trust assignment is the level of trust that a node assigns to a neighbor, without any previous knowledge. We also chose $\alpha = \beta = perception = 0.5$. These are the standard values for the simulations. For each specific configuration, the parameters that differ from its standard values are outlined. At last, in each configuration, all nodes have nature equal to 0.9, which means one out of ten actions taken is bad on average.

4.3.1 Changing behavior

In Section 4.1, we show that nodes are capable of evaluating their neighbor nature using our trust model. However, a node might change its behavior and consequently its nature during its lifetime. The behavior variation of a node occurs due to several reasons. For instance, a node may behave well at first, but after being compromised it starts to misbehave. Another possibility is a good node that experiences some energy consumption problem and begins to misbehave. A third option, as already mentioned in Chapter 3, is the on-off attack. It occurs when malicious nodes continuously change its behavior between good and bad in order to cause damage to the network. Therefore, it is important for a trust model to provide nodes with the capability of identifying such behavior variations as quick as possible. Thus, in the first set of simulations we analyze the trust evaluation of a node that changes its behavior during the simulation. The scenario consists of 20 nodes with 250 m transmission range, which are randomly placed in a 150 m × 150 m area. In this particular scenario, node 1 changes its nature from 0.9 to 0.2 at 200 units of time.

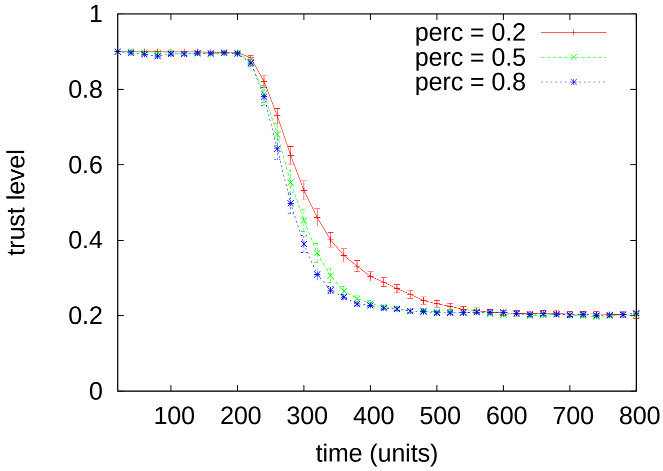


Figure 4.13: Identifying behavior changes.

Figure 4.13 presents the behavior change detection according to the perception of node 2. We can notice that node 2 succeeds in all attempts to remark a change in node 1 behavior. When a node has a low perception means that it has trouble to notice its neighbor actions. This is the reason why a lower perception can slow down the trust evaluation process in the presence of behavior variations, as we can see in Figure 4.13.

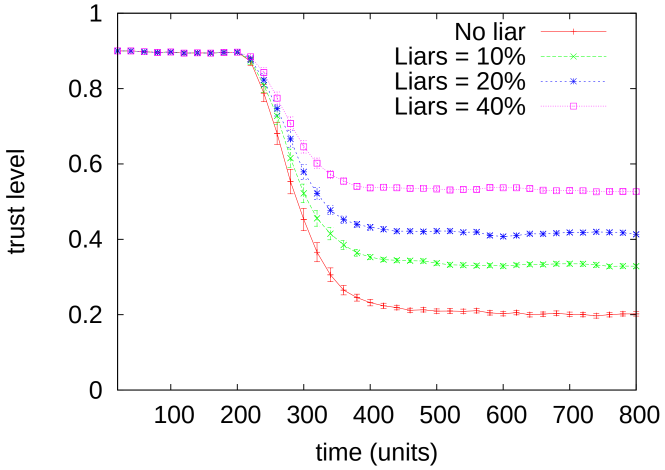
In another scenario, malicious nodes might try to cover the behavior variations of each other in order to keep a good reputation even though they have a bad behavior. Figure 4.14 shows a scenario where node 1 changes its nature from 0.9 to 0.2 and malicious nodes lie about node 1 trying to convince the other nodes that node 1 still has a trust level equals to 0.9. Figure 4.14(a) reveals the effect of a collusion attack varying the percentage of malicious nodes participating in the attack. We observe that malicious nodes can deteriorate the trust evaluation. However, it shows that node 2 manages to identify node 1 as a bad node, namely trust level less than 0.5, if the percentage of malicious nodes is smaller than 40%.

Afterwards, we propose a scenario similar to the last one, but we fixed the percentage of malicious nodes in 40%. In this scenario, we consider that nodes are capable of identifying a change in the behavior of all malicious nodes after a certain amount of time. For instance, nodes can notice that a node is lying by comparing the recommendations it receives with its own experience during a period of time. If there is a significant discrepancy it may classify the node as malicious, and consequently, it can degrade the trust level of the detected neighbor. The results show that detecting liars can improve significantly the trust evaluation performance (curve "ident" Figure 4.14(b)) in the presence of liars. An even better solution is to detect and then to ignore completely the recommendations of malicious nodes, as shown by curve "ident + ignore" in Figure 4.14(b). Ignoring liars is a simple task. Node can simply ignore all recommendations of neighbors with a trust level under a certain threshold. We observe that ignoring liars can neutralize a lying collusion attack. The only damage is during the process of liar detection.

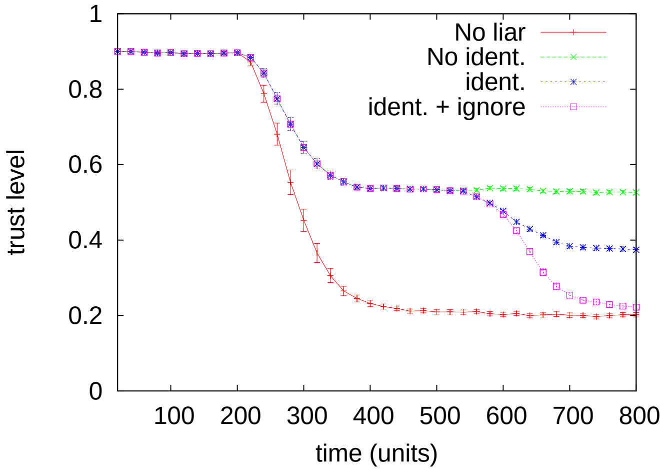
4.3.2 Slander attack

The slander attack consists of sending false recommendations to injure the reputation of a node. Malicious nodes can collude to improve the effect of the attack. In Figure 4.15, node 2 tries to evaluate the trust level of node 1 (0.9). Malicious nodes send false recommendations saying that node 1 has a trust level equals to 0.2. We vary the percentage of liars to show that node 2 can succeed in identifying node 1 as good node (Trust Level > 0.5) for a percentage of liars smaller than 40% as in the result for nodes that lie to cover behavior variations.

Figure 4.16 presents the result for the variation of two important parameters in our model. First, we vary *alpha* (Figure 4.16(a)). The parameter *alpha* is the one that controls



(a) Varying the proportin of liars.



(b) 40% of liars.

Figure 4.14: Nodes try to cover behavior changes.

the weight of recommendations and own experiences in the calculation of the trust level in Equation 3.1. With a higher *alpha* the recommendations of other nodes has a higher weight on the trust level evaluation. It is clear that the more a node considers the recommendations of other nodes, the more it is vulnerable to lying attacks. Therefore, a node might have a low value for *alpha* ($\alpha < 0.5$) in order to be more resistant to liars.

Figure 4.16(b) displays the impact of the perception on the slander attack. The first remark is that the perception does not impact on the trust level evaluation under a slander

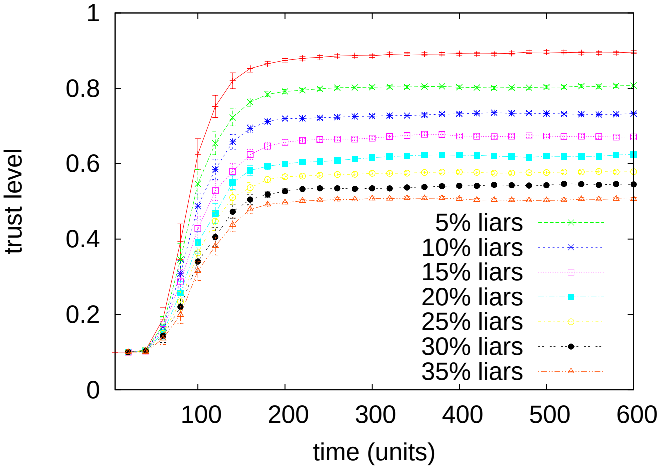
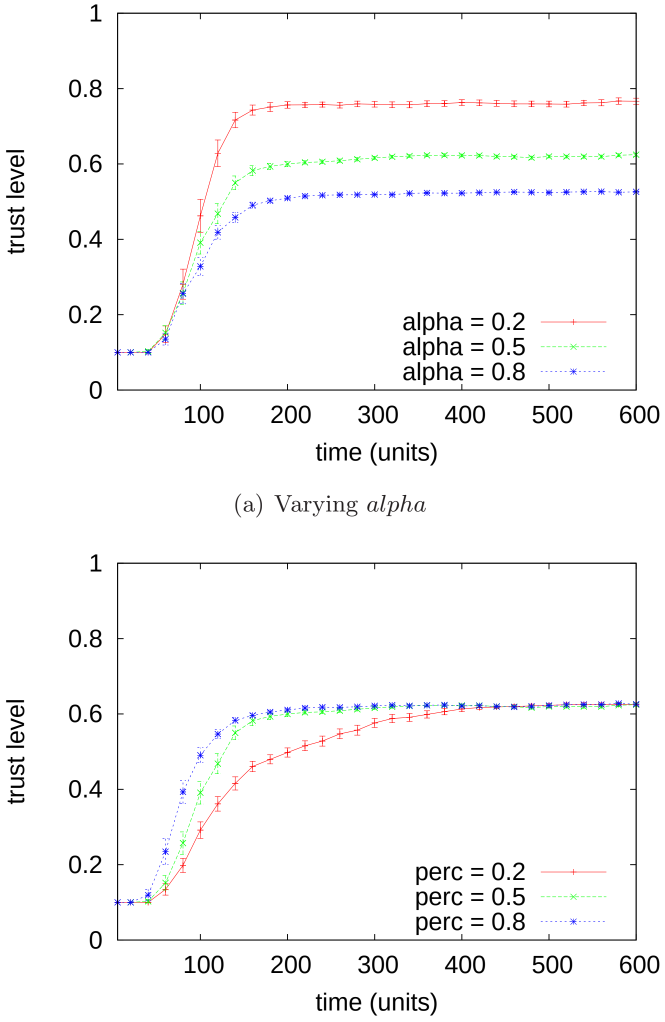


Figure 4.15: Slander attack - varying the percentage of liars.

attack. It can be explained by the fact that the perception has influence only in the duration of the transient period and has no influence on the level achieved after convergence, in the stationary period, as shown in [5]. The transient period nodes are trying to approximate to the expected value, while in the stationary period, the trust level is almost stable, very close to the correct value.

We changed the perception of node 2 to 0.2 and the parameter alpha to 0.8 as a worst case scenario for a slander attack. Figure 4.17 presents the results when malicious nodes begin to lie after 200 time units so they already have a good reputation. We observe that if node 2 detects the misbehavior of the malicious nodes and ignore their recommendations (curve "lying at 200 + ident.") there is no damage to the trust evaluation process, except for the period during which node 2 has not yet notice the liars. This period depends solely on the capacity of the node in detecting a lie.

In Figure 4.18 we vary the duration of the detection of liars. The results show that identifying liars is an important task to avoid damage to the trust system. A fast liar detection mechanism can offer a robust trust system against slander attacks. It can be noticed that the recovery delay, namely, the time a node take to achieve the correct trust value after identifying all liars in the neighborhood, remains the same regardless of the detection delay.



(a) Varying *alpha*

(b) Varying perception

Figure 4.16: Slander attack - varying trust model parameters.

4.4 Discussion

We have learnt from simulation results that, in mobile ad hoc networks, increasing the value of *alpha*, which gives more importance to the recommendations over the experiences, is good strategy to improve the trust model efficiency. Nevertheless, the *alpha* parameter plays an important role in reducing the influence of liars. Nodes with a large *alpha* are more vulnerable to false recommendations. Therefore, we observe the existence of an important trade-off between mobility and vulnerability to slander attacks. A possible solution to

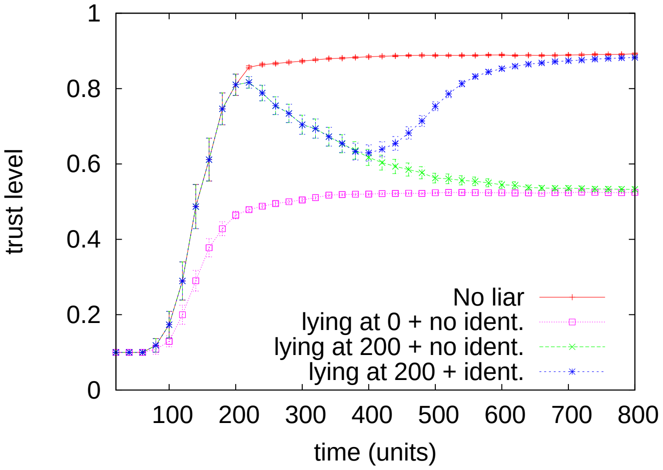


Figure 4.17: Slander attack - worst case.

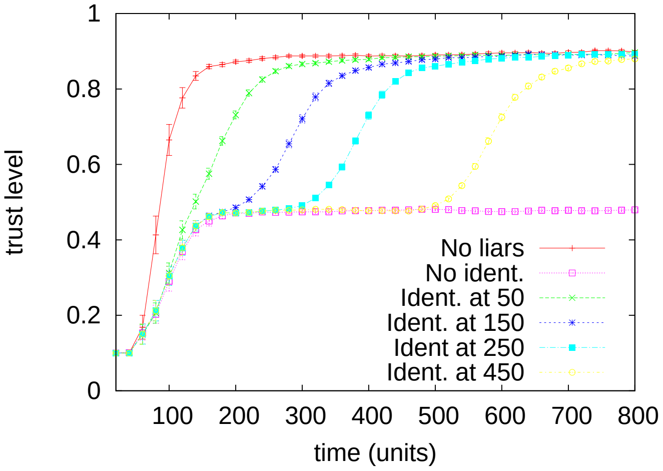


Figure 4.18: Slander attack - detecting slanderer nodes.

overcome the trade-off problem consists of implementing a liar detection mechanism. One feasible approach to detect liars lies in comparing recommendations of all neighbors. Considering that the percentage of malicious nodes is smaller than 50%, a node might assume as a liar every node that keeps sending conflicting recommendations.

4.5 Résumé du Chapitre

Dans ce chapitre, nous présentons les résultats plus importants de nos expériments.

Le premier ensemble d'expériments vise à démontrer l'exactitude de notre modèle et l'impact des principaux paramètres sur l'évaluation de la confiance. Nous montrons également les principales caractéristiques de la dynamique du processus d'évaluation de la confiance dans les réseaux ad hoc. Notre objectif dans cet expériment est d'évaluer et d'analyser l'influence du nombre de voisins, de la stratégie d'attribution de la première valeur de confiance et de la variation des paramètres alpha et de perception sur l'évaluation de la confiance. La raison d'analyser uniquement les réseaux de communication directe dans cet expériment est d'isoler tous les problèmes liés aux réseaux mobiles multisaut à fin de se concentrer strictement sur la dynamique de notre modèle. Les résultats prouvent qu'augmenter le nombre de voisins accélère la convergence de l'évaluation de la confiance.

Dans ce travail, nous analysons notre modèle dans les réseaux mobiles ad hoc multisaut. Dans cet expériment nous analysons l'impact de la maturité du rapport et l'influence de la variation des paramètres alpha et de perception sur notre modèle dans des scénarios mobiles. Nous démontrons l'efficacité du paramètre de maturité du rapport et comment les autres paramètres sont ajustés pour améliorer l'évaluation de la confiance dans des scénarios mobiles. Le paramètre de maturité du rapport peut diminuer jusqu'à 50% l'erreur de la valeur de confiance.

L'objectif de la dernière simulation est d'évaluer l'exécution du système de confiance en présence d'attaques de fausses recommandations et de coalisions de noeuds malveillants dans des réseaux ad hoc de communication directe. Les résultats prouvent la robustesse de notre modèle contre les attaques de noeuds menteurs. Nous prenons en considération la présence de noeuds malveillants qui envoient des fausses recommandations dans deux scénarios différents. Dans le premier, nous considérons que les noeuds malveillants s'entendent à fin de cacher le changement de comportement d'un autre noeud malveillant. Le deuxième scénario considère que les noeuds s'entendent pour diffamer un de ses voisins, c'est à dire, les noeuds envoient des fausses recommandations pour déprécier la réputation d'autres voisins. Les résultats prouvent que notre modèle de confiance tolère jusqu'à 40% de noeuds malveillants. Nous présentons un scénario où les noeuds peuvent détecter ceux qui envoient des fausses recommandations. Dans un tel scénario, après la détection d'un menteur, les noeuds commencent à ignorer ses recommandations, neutralisant l'effet des actes du noeud malveillant. Ainsi, les menteurs qui ont été détectés ne peuvent plus perturber le processus d’évaluation de la confiance de ses voisins.

Chapter 5

Conclusion

Ad hoc networks experiences a significant success due to its advantages over infrastructured networks. At the same time that the lack of infra-structure, an intrinsic characteristic of ad hoc networks, provides flexibility, it also imposes several important obstacles. One of the most relevant issues is that ad hoc networks rely on collaborative behavior of nodes to work properly. Therefore, nodes must trust each other at some level to allow distributed applications, including routing and admission control. Cryptographic mechanisms are important to guarantee authentication of nodes and integrity of messages but they neither avoid nor protect the network against misbehaving nodes. Thus, a trust system is mandatory in ad hoc networks to provide reliable communications and network availability. The ability of estimating the trustworthy of its neighbors provides several advantages to ad hoc networks, such as, behavior prediction, stimulation of cooperation, and exchange of information with trustworthy neighbors. Nevertheless, the trust system must be carefully designed because a naive trust model might lead to low efficiency, high energy consumption, and more vulnerability to attacks.

This thesis addresses the problem of trust evaluation and management in ad hoc networks. Therefore, we propose a trust model based on the concept of human trust, which provides nodes with a mechanism to evaluate the trust level of its direct neighbors. The basic idea consists of using previous experiences and recommendations of other neighbors to appraise the trust level of other nodes. The model is composed by a Learning layer and a Trust layer. The Learning layer is responsible for gathering and converting information into knowledge by monitoring the behavior of each neighbor. The Trust layer then defines how to assess the trust level of each neighbor using the knowledge information provided by the Learning layer and the information exchanged with direct neighbors. We introduce the concept of relationship maturity, which allows nodes to attribute more relevance to the recommendations issued by nodes that know the evaluated neighbor for a long time. We also propose the Recommendation Exchange Protocol (REP) which enables nodes to send and receive recommendations of its neighbors.

Different from related work present in this thesis, our model takes into account the limitations of wireless ad hoc networks, which are usually composed of portable devices with power, processing, and memory constraint. This is accomplished by confining the interactions among nodes to direct neighbors only. Such approach implies significant lower energy consumption, less processing for trust level calculation, and less memory space. In addition, this characteristic helps to minimize the effect of false recommendations. First, it reduces the number of recommendations. Second, there is no intermediate node to increase the uncertainty of the information, since recommendations are sent in one hop and are not forwarded. Third, a node can always balance the recommendations with its own experiences to calculate the trust level.

An important quality of our model is the flexibility due to the possibility of operating in three different modes, depending on the node resource restrictions. Thus, the proposed model is suitable for heterogeneous network, where nodes present distinct constraints. Besides, the presence of nodes that do not implement at all our trust system do not disturb the other nodes that are using the system.

We perform a number of experiments based on simulations, using, most of the time, worst case scenarios. Accordingly, we have developed a simulator, which was specifically designed and developed for our model.

First we show the results related to single-hop ad hoc networks, which demonstrate the correctness of our model and the impact of the main parameters on the trust evaluation process. The results show that a larger number of neighbors accelerate the convergence of the trust evaluation process. We also observe two distinct phases. A transient period in which nodes are trying to converge to a certain trust value and a stationary period where the trust level is stable.

Afterwards, we evaluate our model in mobile multi-hop ad hoc networks. We show the effectiveness of the relationship maturity parameter, which reduces the trust level error in almost 50%, in certain situations. We show that in mobile ad hoc networks, the transient period has a significant relevance because if a node moves before the end of the transient period means that trust level evaluation has not converged. Therefore, we identify that a node can accelerate the transient period by increasing the *alpha* parameter, giving more emphasis to the recommendations of its neighbors. Results also show that in mobile ad hoc networks selecting the strategy for assigning the first trust level can also accelerate the transient period.

Another important result indicates that our model detects behavior changes of nodes and is robust to slander and colluding attacks. We prove the robustness of our model against slander attacks because it takes into account the presence of malicious nodes lying about their recommendations. In the first simulated scenario, we consider that malicious nodes collude to hide from other nodes the misbehavior of another malicious node. The other scenario considers nodes that collude to slander one of its neighbors, namely, nodes sending false recommendations to depreciate the reputation of other neighbors. The results reveal that the proposed model tolerates up to 40% of liars. We presented a scenario where nodes are able to detect nodes that send false recommendations. In such scenario, after detecting a liar, nodes start to ignore its recommendation, neutralizing the acts of the malicious node. Therefore, detected liars can not perturb the trust evaluation process of its neighbors. An intuitive result that is confirmed by our experiments is that decreasing the *alpha* parameter diminish the influence of false recommendations. Nevertheless, this result suggests the existence of trade-off between mobility and protection against liars. Thus, we conclude that mobile nodes tend to be more vulnerable to false recommendations attacks.

5.1 Future works

The next generation of networks needs to take into account the concept of autonomic networks, where nodes have to self-configure, self-manage, etc- [131, 132]. This concept is even more important for ad hoc networks because there is no central entity to help nodes with all these tasks. Therefore, a self-configuring trust model is the next step in our work. An eventual approach must be a context-aware scheme, which means, taking into account the mobility, the number of nodes, the number of liars, the resource constraints, among others. Another interesting idea is to develop a Learning layer considering specific application or service characteristics, for instance, a routing protocol. Then, the routing protocol can use the trust information provided by the system to improve its efficiency and security.

5.2 Résumé du Chapitre

Cette thèse aborde le problème de l'évaluation et de la gestion de confiance dans les réseaux ad hoc. Par conséquent, nous proposons un modèle de confiance basé sur le concept de la confiance humaine, fournissant aux noeuds un mécanisme pour évaluer le niveau de confiance de ses voisins directs. L'idée fondamentale est basée sur la combinaison des expériences précédentes et des recommandations d'autres voisins pour évaluer le niveau de confiance des noeuds du réseau. Nous présentons le concept de maturité du rapport, qui permet aux noeuds de donner plus d'importance aux recommandations envoyé par les noeuds qui connaissent leur voisin depuis longtemps. Nous proposons également le Protocole d'Echange de Recommandation (Recommendation Exchange Protocol - REP) qui permet aux noeuds d'envoyer et de recevoir des recommandations de leurs voisins.

Différemment des autres travails qui abordent le sujet, notre modèle prend en considération les limitations des réseaux ad hoc sans fil, qui se composent habituellement de dispositifs portables possédant des contraintes de puissance, de traitement et de mémoire. Ceci est accompli en limitant les interactions entre les noeuds uniquement aux voisins directs. Une telle approche implique une faible consommation de ressources et une réduction de l'effet des fausses recommandations. Une qualité importante de notre modèle est la flexibilité due à la possibilité de fonctionnement en trois modes différents, selon les restrictions de ressources des noeuds. Ainsi, le modèle proposé convient aux réseaux hétérogènes, où les noeuds présentent des contraintes distinctes. En outre, la présence de noeuds qui n'utilisent pas notre système de confiance ne dérange pas les autres noeuds qui utilisent le système.

Nous exécutons un certain nombre d'expériments basées sur des simulations, en utilisant des scénarios de pire cas. Ainsi, nous avons développé un simulateur spécifiquement conçu pour notre modèle. D'abord nous présentons les résultats liés aux réseaux ad hoc de communication directe, qui démontrent l'exactitude de notre modèle et l'impact des principaux paramètres sur l'évaluation de la confiance. Deux phases différentes sont observées: un état transitoire où les noeuds essayent de converger à une certaine valeur de confiance et une état stationnaire où le niveau de confiance est stable. Ensuite, nous évaluons notre modèle dans des réseaux ad hoc mobiles multisaut. Nous montrons l'efficacité du paramètre de maturité du rapport, qui réduit l'erreur du niveau de confiance de presque 50% dans certaines situations. Nous prouvons que dans les réseaux ad hoc mobiles, l'état transitoire a une importance significative puisque si un noeud se déplace avant que cet état soit fini, l'évaluation de niveau de confiance ne converge pas. Par conséquent, nous identifions qu'un noeud peut accélérer l'état transitoire en augmentant le paramètre *alpha*, accordant plus d'importance aux recommandations de ses voisins. Un autre résultat important indique que notre modèle est capable de détecter des changements de comportement des noeuds et est assez robuste contre les attaques des fausses recommandations. Nous démontrons que le modèle proposé tolère jusqu'à 40% de noeuds malveillants. Un résultat intuitif qui est confirmé par nos expériments est qu'un noeud peut diminuez l'influence des fausses recommandations en diminuant le paramètre *alpha*. Néanmoins, ce résultat suggère l'existence d'un compromis entre la mobilité et la protection contre les noeud menteurs. Ainsi, nous concluons que les noeuds mobiles tendent à être plus vulnérables aux attaques des fausses recommandations. Nous présentons un scénario où les noeuds peuvent détecter les menteurs. Dans un tel scénario, après la détection d'un noeud menteur, les noeuds commencent à ignorer sa recommandation, neutralisant les effets des actes du noeud malveillant. Ainsi, les menteurs détectés ne peuvent pas perturber le processus d'évaluation de la confiance de ses voisins.

Appendix A

Simulation parameters

In this appendix, we present the simulator parameters (Table A.1) used in our experiments. The main parameters are:

- The MeanAction: the actions are exponentially distributed. This parameter is the mean value
- The RadioRange: the maximum distance a node can communicate
- The TL_{th}: the maximum difference between two consecutives trust level values about the same neighbor
- The TA_timeout: the amount of time a node waits until sending a Trust Advertisement (TA) message. It is used to send the maximum number of TAs in the same message
- The TREQ_timeout: the period of time a node waits for a Trust Request (TREQ) message
- The TREP_timeout: the period of time a node waits until sending a Trust Reply (TREP) message. It is used to send the maximum number of TAs in the same message
- The BACKOFF_MAX: the maximum value of the backoff function (uniform(0,BACKOFF_MAX)), which is used before sending a message

Table A.1: Parameter values used in the simulations.

Parameters	Value
$beta(\beta)$	0.5
MeanAction	5 units
Radiorange	250 m
TL_{th}	0.05
TA_timeout	1 unit
TREQ_timeout	2 units
TREP_timeout	1 unit
BACKOFF_MAX	1 unit

References

[1] D. Artz and Y. Gil, “A survey of trust in computer science and the semantic web,” *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 5, no. 2, pp. 58–71, June 2007.

[2] A. Josang, “Trust and reputation systems,” in *Foundations of Security Analysis and Design IV, FOSAD 2006/2007 - Tutorial Lectures*, (Bertinoro, Italy), Springer LNCS 4677, Sept. 2007.

[3] J. O. Kephart and D. M. Chess, “The vision of autonomic computing,” *IEEE Computer*, vol. 36, no. 1, pp. 41–52, Jan. 2003.

[4] P. B. Velloso, R. P. Laufer, O. C. M. B. Duarte, and G. Pujolle, “HIT: A human-inspired trust model for ad hoc networks,” tech. rep., Laboratoire d’Informatique de Paris 6 (LIP6), Université Pierre et Marie Curie, July 2005.

[5] P. B. Velloso, R. P. Laufer, O. C. M. B. Duarte, and G. Pujolle, “HIT: A human-inspired trust model,” in *8th IFIP IEEE International Conference on Mobile and Wireless Communication Networks - MWCN’2006*, (Santiago, Chile), Aug. 2006.

[6] P. B. Velloso, R. P. Laufer, O. C. M. B. Duarte, and G. Pujolle, “Um novo modelo para confiança em rede ad hoc,” in *Brazilian Symposium of Computer Networks - SBRC’2006*, (Curitiba, Brazil), May 2006.

[7] P. B. Velloso, R. P. Laufer, O. C. M. B. Duarte, and G. Pujolle, “Analyzing a human-based trust model for mobile ad hoc networks,” in *IEEE Symposium on Computers and Communications (ISCC’2008), accepted for publication*, (Marrakech, Morocco), Aug. 2008.

[8] P. B. Velloso, R. P. Laufer, O. C. M. B. Duarte, and G. Pujolle, "Análise de um modelo de confiança para redes ad hoc," in *Brazilian Symposium of Computer Networks - SBRC'2008*, (Rio de Janeiro, Brazil), May 2008.

[9] P. B. Velloso, R. P. Laufer, O. C. M. B. Duarte, and G. Pujolle, "A trust model robust to slander attacks in ad hoc networks," in *IEEE International Conference on Computer Communications and Networks (ICCCN'08) ANC workshop, accepted for publication*, (Virgin Islands, USA), Aug. 2008.

[10] D. O. Cunha, R. P. Laufer, I. M. Moraes, M. D. D. Bicudo, P. B. Velloso, and O. C. M. B. Duarte, "A bio-inspired field estimation scheme for wireless sensor networks," *Annals of Telecommunications*, vol. 60, no. 7-8, Aug. 2005.

[11] P. B. Velloso, M. G. Rubinstein, and O. C. M. B. Duarte, "Evaluating voice traffic requirements on IEEE 802.11 ad hoc networks," *Annals of Telecommunications*, vol. 63, no. 5-6, pp. 321–329, June 2008.

[12] L. Zhou and Z. J. Haas, "Securing ad hoc networks," *IEEE Communiations Magazine*, vol. 13, no. 6, pp. 24–30, Dec. 1999.

[13] N. Komninosa, D. Vergadosa, and C. Douligerisb, "Detecting unauthorized and compromised nodes in mobile ad hoc networks," *Ad Hoc Networks*, vol. 5, no. 3, pp. 289–298, Apr. 2007.

[14] M. Abolhasan, T. Wysocki, and E. Dutkiewicz, "A review of routing protocols for mobile ad hoc networks," *Ad hoc Networks*, vol. 2, no. 1, pp. 1–22, Jan. 2004.

[15] J. K. Changling Liu, "A survey of mobile ad hoc network routing protocols," tech. rep., University of Ulm, 2005.

[16] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *ACM Computer Communication Review*, vol. 24, no. 4, pp. 234–244, Oct. 1994.

[17] S. Murthy and J. J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," *Mobile Networks and Applications*, vol. 1, no. 2, pp. 183–197, Oct. 1996.

[18] D. B. Johnson, D. A. Maltz, and J. Broch, "DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks." Ad hoc Networks, 2001. Chapter 5, pp. 139-172, Addison-Wesley.

[19] C. E. Perkins and E. M. Royer, “Ad hoc on demand distance vector routing, mobile computing systems and applications,” in *IEEE Workshop on Mobile Computing Systems and Applications (WMCSA’99)*, (New Orleans, USA), Feb. 1999.

[20] Z. J. Haas, “The zone routing protocol (ZRP) for ad hoc networks,” Nov. 1997. Internet Draft.

[21] M. Joa-Ng and I.-T. Lu, “A peer-to-peer zone-based two-level link state routing for mobile ad hoc networks,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, no. 8, 1999.

[22] M. Jiang, J. Li, and Y. C. Tay, “Cluster based routing protocol (CBRP),” 2001. Internet Draft.

[23] P. Sinha, R. Sivakumar, and V. Bharghaven, “CEDAR: a core-extraction distributed ad hoc routing algorithm,” in *IEEE INFOCOM*, Mar. 1999.

[24] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, “Optimized link state routing protocol for ad hoc networks,” in *IEEE International Multi Topic Conference (INMIC’01)*, Dec. 2001.

[25] J. Li, J. Jannotti, D. S. J. D. Couto, D. Karger, and R. Morris, “A scalable location service for geographic ad hoc routing,” in *Mobicom’00*, Aug. 2000.

[26] C.-K. Toh, “Associativity based routing for ad hoc mobile networks,” *Wireless Personal Communications Journal - Special Issue on Mobile Networking and Computing Systems*, vol. 4, no. 2, Mar. 1997.

[27] H. Deng, W. Li, and D. P. Agrawal, “Routing security in wireless ad hoc networks,” *IEEE Communiations Magazine*, pp. 70–75, Oct. 2002.

[28] R. P. Laufer, P. B. Velloso, D. de Oliveira Cunha, I. M. Moraes, M. D. D. Bicudo, M. D. D. Moreira, and O. C. M. B. Duarte, “Defeating DoS attacks with IP traceback,” in *IFIP Open Conference on Metropolitan Area Networks (MAN’2005)*, (Ho Chi Minh, Viet Nam), Apr. 2005.

[29] R. P. Laufer, P. B. Velloso, D. de Oliveira Cunha, I. M. Moraes, M. D. D. Bicudo, M. D. D. Moreira, and O. C. M. B. Duarte, “Towards stateless single-packet IP

traceback," in *IEEE Conference on Local Computer Networks (LCN'2007)*, (Dublin, Ireland), Oct. 2007.

[30] T. R. Andel and A. Yasinsac, "Adaptive threat modeling for secure ad hoc routing protocols," *Ad Hoc Networks*, vol. 197, no. 2, pp. 3–14, Feb. 2008.

[31] A. Fourati and K. Al Agha, "A shared secret-based algorithm for securing the OLSR routing protocol," *Telecommunication Systems*, vol. 31, no. 2–3, pp. 213–226, Mar. 2006.

[32] I. Gawedzki and K. A. Agha, "How to avoid packet droppers with proactive routing protocols for ad hoc networks," *International Journal of Network Management*, vol. 18, no. 2, pp. 195–208, Mar. 2008.

[33] Y. C. Hu, D. B. Johnson, and A. Perrig, "SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks," in *IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'02)*, June 2002.

[34] Y. C. Hu, D. B. Johnson, and A. Perrig, "Ariadne: A secure on-demand routing protocol for ad hoc networks," in *Mobicom'02*, Sept. 2002.

[35] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer, "A secure routing protocol for ad hoc networks," in *IEEE International Conference on Network Protocol (ICNP'02)*, Nov. 2002.

[36] R. Zheng and R. Kravets, "On-demand power management for ad hoc networks," *Ad Hoc Networks*, vol. 3, no. 1, pp. 51–68, Jan. 2005.

[37] G. Theodorakopoulos and J. S. Baras, "Malicious users in unstructured networks," in *IEEE International Conference on Computer Communications (INFOCOM'07)*, (Anchorage, Alaska, USA), May 2007.

[38] W. Yu and K. J. R. Liu, "Attack-resistant cooperation stimulation in autonomous ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 12, pp. 2260–2271, Dec. 2005.

[39] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *IEEE INFOCOM'03*, (San Francisco, USA), Apr. 2003.

[40] L. Buttyan and J. P. Hubaux, "Enforcing service availability in mobile ad-hoc wans," in *IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHoc)*, (Boston, USA), Aug. 2000.

[41] L. Buttyan and J. P. Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," *ACM/Kluwer Mobile Networks and Applications (MONET)*, vol. 8, no. 5, pp. 579–592, Oct. 2003.

[42] J. Crowcroft, R. Gibbens, F. Kelly, and S. Östring, "Modelling incentives for collaboration in mobile ad hoc networks," *Performance Evaluation*, vol. 57, no. 4, pp. 427–439, Aug. 2004.

[43] J. Pan, L. Cai, X. S. Shen, and J. W. Mark, "Identity-based secure collaboration in wireless ad hoc networks," *Computer Networks*, vol. 51, no. 3, pp. 853–865, Feb. 2007.

[44] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *International Conference on Mobile Computing and Networking (MobiCom'00)*, (Boston, USA), Aug. 2000.

[45] L. Huang, L. Lei, L. Lixiang, Z. Haibin, and L. Tang, "Stimulating cooperation in route discovery of ad hoc networks," in *ACM workshop on QoS and security for wireless and mobile networks (Q2SWinet '07)*, (Chania, Crete Island, Greece), Oct. 2007.

[46] V. Srinivasan, P. Nuggehalli, C. F. Chiasserini, and R. R. Rao, "Cooperation in wireless ad hoc networks," in *IEEE INFOCOM'03*, (San Francisco, USA), Apr. 2003.

[47] L. Yan, S. Hailes, and L. Capra, "Analysis of packet relaying models and incentive strategies in wireless ad hoc networks with game theory," in *IEEE International Conference on Advanced Information Networking and Applications (AINA'08)*, (GinoWan, Japan), Mar. 2008.

[48] J. N. Al-Karaki and A. E. Kamal, "Stimulating node cooperation in mobile ad hoc networks," *Wireless Personal Communications*, vol. 44, no. 2, pp. 219–239, Jan. 2008.

[49] D. H. McKnight and N. L. Chervany, "The meanings of trust," tech. rep., Management Information Systems Reseach Center, University of Minnesota, 1996.

[50] D. H. McKnight and N. L. Chervany, “What is trust? a conceptual analysis and an interdisciplinary model,” in *Proceedings of Americas Conference on Information Systems (AMCIS 2000)*, (Long Beach, USA), Aug. 2000.

[51] D. H. McKnight and N. L. Chervany, “Trust and distrust definitions: One bite at a time,” *Lecture Notes in Computer Science*, vol. 2246, pp. 27–54, Jan. 2001.

[52] A. Josang, R. Ismail, and C. Boyd, “A survey of trust and reputation systems for online service provision,” *Decision Support Systems*, vol. 43, no. 23, pp. 618–644, Mar. 2007.

[53] S. P. Marsh, *Formalizing trust as a computational concep.* PhD thesis, Department of Mathematics and Computer Science, University of Stirling, Stirling, Scotland, UK, 1994.

[54] P. Obreiter, “A case for evidence-aware distributed reputation systems: Overcoming the limitations of plausibility considerations,” *Lecture Notes in Computer Science*, vol. 2995, pp. 33–47, Feb. 2004.

[55] M. Kinateder and K. Rothermel, “Architecture and algorithms for a distributed reputation system,” in *International Conference on Trust Management (iTrust’03)*, (Heraklion, Crete, Greece), May 2003.

[56] P. R. Zimmermann, *The Official PGP User’s Guide.* MIT Press, 1995.

[57] C. Adams and S. Farrel, “Internet X.509 public key infrastructure certificate management protocols,” 1999. IETF, RFC 2510.

[58] R. Demolombe, “Reasoning about trust: A formal logical framework,” in *International Conference on Trust Management (iTrust’04)*, (Oxford, UK), Apr. 2004.

[59] R. Demolombe, “Reasoning about trust: A formal logical framework,” *Lecture Notes in Computer Science*, vol. 2995, pp. 291–303, Feb. 2004.

[60] A. Josang and S. Lo Presti, “Analysing the relationship between risk and trust,” in *International Conference on Trust Management (iTrust’04)*, (Oxford, UK), Apr. 2004.

[61] R. Morselli, J. Katz, and B. Bhattacharjee, “A game-theoretic framework for analyzing trust-inference protocols,” in *Workshop on the Economics of Peer-to-Peer Systems (P2PEcon’04)*, (Cambridge, USA), June 2004.

[62] K. Komathy and P. Narayanasamy, “Trust-based evolutionary game model assisting AODV routing against selfishness,” *Journal of Network and Computer Applications (available online)*, Feb. 2008.

[63] C. M. Jonker and J. Treur, “Formal analysis of models for the dynamics of trust based on experiences,” in *European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW’99)*, (Brisbane, Australia), Sept. 1999.

[64] C. M. Jonker, J. J. Schalken, J. Theeuwes, and J. Treur, “Human experiments in trust dynamics,” *Lecture Notes in Computer Science*, vol. 2995, pp. 206–220, Feb. 2004.

[65] L. Mui, M. Mohtashemi, and A. Halberstadt, “A computational model of trust and reputation,” in *Annual Hawaii International Conference on System Sciences (HICSS’02)*, (Big Island, Hawaii, USA), Jan. 2002.

[66] M. Michalakopoulos and M. Fasli, “On deciding to trust,” *Lecture Notes in Computer Science*, vol. 3477, pp. 61–76, May 2005.

[67] M. G. Rubinstein, O. C. M. B. Duarte, and G. Pujolle, “Scalability of a mobile agents based network management application,” *IEEE/KICS Journal of Communications and Networks*, vol. 5, no. 3, pp. 240–248, Sept. 2003.

[68] U. G. Wilhelm, S. M. Staamann, and L. Buttyan, “A pessimistic approach to trust in mobile agent platforms,” *IEEE Internet Computing*, vol. 4, no. 5, pp. 40–48, Sept. 2000.

[69] H. Li and M. Singhal, “Trust management in distributed systems,” *Computer*, vol. 40, no. 2, pp. 45–53, Feb. 2007.

[70] A. Jösang, C. Keser, and T. Dimitrakos, “Can we manage trust?,” *Lecture Notes in Computer Science*, vol. 3477, pp. 93–107, May 2005.

[71] M. Carbone, M. Nielsen, and V. Sassone, “A formal model for trust in dynamic networks,” in *International Conference on Software Engineering and Formal Methods, SEFM’03*, (Brisbane, Australia), Sept. 2003.

[72] "E-bay." http://www.ebay.com. Online auction and shopping website.

[73] "Amazon." http://www.amazon.com. Online shopping website.

[74] A. Fernandes, E. Kotsovinos, S. Östring, and B. Dragovic, "Pinocchio: Incentives for honest participation in distributed trust management," in *International Conference on Trust Management (iTrust'04)*, (Oxford, UK), Apr. 2004.

[75] A. Fernandes, E. Kotsovinos, S. Östring, and B. Dragovic, "Pinocchio: Incentives for honest participation in distributed trust management," *Lecture Notes in Computer Science*, vol. 2995, pp. 63–77, Feb. 2004.

[76] P. Resnick and H. R. Varian, "Recommender systems," *Communications of the ACM*, vol. 40, no. 3, pp. 56–58, Mar. 1997.

[77] P. Massa and B. Bhattacharjee, "Using trust in recommender systems: An experimental analysis," in *International Conference on Trust Management (iTrust'04)*, (Oxford, UK), Apr. 2004.

[78] K. Cheverst, N. Davies, K. Mitchell, and A. Friday, "Experiences of developing and deploying a context-aware tourist guide: the guide project," in *International Conference on Mobile Computing and Networking (Mobicom'00)*, (Boston, USA), Aug. 2000.

[79] D. Ingram, "Trust-based filtering for augmented reality," *Lecture Notes in Computer Science*, vol. 2692, p. 1072, Jan. 2003.

[80] "Facebook." http://www.facebook.com. Online community website.

[81] "Orkut." Online community website.

[82] J. Golbeck and J. Hendler, "Accuracy of metrics for inferring trust and reputation in semantic web-based social networks," in *International Conference on Knowledge Engineering and Knowledge Management (EKAW'04)*, (Northamptonshire, UK), Oct. 2004.

[83] J. Golbeck and J. Hendler, "Accuracy of metrics for inferring trust and reputation in semantic web-based social networks," *Lecture Notes in Computer Science*, vol. 3257, pp. 116–131, Sept. 2004.

[84] C.-N. Ziegler and G. Lausen, “Spreading activation models for trust propagation,” in *IEEE International Conference on e-Technology, e-Commerce, and e-Service (EEE ’04)*, (Taipei, Taiwan), Mar. 2004.

[85] T. Jiang and J. S.Baras, “Trust evaluation in anarchy: A case study on autonomous networks,” in *IEEE International Conference on Computer Communications (INFOCOM’06)*, (Barcelona, Spain), Apr. 2006.

[86] C. English, W. Wagealla, P. Nixon, S. Terzis, H. Lowe, and A. McGettrick, “Trusting collaboration in global computing systems,” in *International Conference on Trust Management (iTrust’03)*, (Heraklion, Crete, Greece), May 2003.

[87] C. English, W. Wagealla, P. Nixon, S. Terzis, H. Lowe, and A. McGettrick, “Trusting collaboration in global computing systems,” *Lecture Notes in Computer Science*, vol. 2692, p. 1072, Jan. 2003.

[88] G. Suryanarayana, J. R. Erenkrantz, and R. N. Taylor, “An architectural approach for decentralized trust management,” *IEEE Internet Computing*, vol. 9, no. 6, pp. 16–23, Dec. 2005.

[89] A. Adnane, R. T. de Sousa Jr., C. Bidan, and L. Mé, “Autonomic trust reasoning enables misbehavior detection in OLSR,” in *ACM symposium on Applied computing (SAC’08)*, (Ceará, Brazil), Mar. 2008.

[90] M. Blaze, J. Feigenbaum, J. Ioannidis, and A. Keromytis, “The keynote trust-management system, version 2,” Sept. 1999. IETF, RFC 2704.

[91] M. Blaze, J. Feigenbaum, and J. Lacy, “Decentralized trust management,” in *IEEE Symposium on Security and Privacy*, (Los Alamos, USA), Apr. 1996.

[92] C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, and T. Ylonen, “SPKI certificate theory,” Sept. 1999. IETF, RFC 2704.

[93] R. L. Rivest and B. Lampson, “SDSI - a simple distributed security infrastructure,” Oct. 1996.

[94] K. Ren, T. Li, Z. Wan, F. Bao, D. Robert H, and K. Kim, “Highly reliable trust establishment scheme in ad hoc networks,” *Computer Networks*, vol. 45, no. 6, pp. 687–699, Aug. 2004.

[95] P. Albers, O. Camp, J.-M. Percher, B. Jouga, L. Mé, and R. Puttini, “Security in ad hoc networks: a general intrusion detection architecture enhancing trust based approaches,” in *International Workshop on Wireless Information Systems (WIS’02)*, (Ciudad Real, Spain), Apr. 2002.

[96] G. Theodorakopoulos and J. S. Baras, “Trust evaluation in ad-hoc networks,” in *Proceedings of the ACM Workshop on Wireless Security (WiSE’04)*, (Philadelphia, USA), Oct. 2004.

[97] G. Theodorakopoulos and J. S. Baras, “On trust models and trust evaluation metrics for ad hoc networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 318–328, Feb. 2006.

[98] Y. Sun, Z. Han, W. Yu, and K. J. R. Liu, “A trust evaluation framework in distributed networks: Vulnerability analysis and defense against attacks,” in *IEEE INFOCOM’06*, (Barcelona, Spain), Apr. 2006.

[99] Q. He, D. Wu, and P. Khosla, “A secure incentive architecture for ad hoc networks,” *Wireless Communications and Mobile Computing*, vol. 6, no. 3, pp. 333–346, May 2006.

[100] E. Gray, J.-M. Seigneur, Y. Chen, and C. Jensen, “Trust propagation in small world,” in *Proceedings of 1st International Conference on Trust Management (iTrust 03)*, (Crete, Greece), May 2003.

[101] E. Gray, P. O’Connell, C. Jensen, S. Weber, J.-M. Seigneur, and C. Yong, “Towards a framework for assessing trust-based admission control in collaborative ad hoc applications,” tech. rep., Trinity College, Dublin, 2002.

[102] C. T. Nguyen, O. Camp, and S. Loiseau, “A bayesian network based trust model for improving collaboration in mobile ad hoc networks,” in *IEEE International Conference on Research, Innovation and Vision for the Future (RIVF’07)*, (Hanoi, Viet-Nam), Mar. 2007.

[103] J. Li, R. Li, and J. Kato, “Future trust management framework for mobile ad hoc networks,” *IEEE Communiations Magazine*, vol. 46, no. 4, pp. 108–114, Apr. 2008.

[104] S. Buchegger and J.-Y. Le Boudec, "The effect of rumor spreading in reputation systems for mobile ad-hoc networks," in *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt03)*, (Sophia-Antipolis, France), Mar. 2003.

[105] S. Chinni, J. Thomas, G. Ghinea, and Z. Shen, "Trust model for certificate revocation in ad hoc networks," *Ad Hoc Networks*, vol. 6, no. 3, pp. 441–457, May 2008.

[106] Z. Liu, A. W. Joy, and R. A. Thompson, "A dynamic trust model for mobile ad hoc networks," in *IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS'04)*, (Suzhou, Chine), May 2004.

[107] C. M. Schweitzer, T. C. Carvalho, and W. Ruggiero, "A distributed mechanism for trust propagation and consolidation in ad hoc networks," *Lecture Notes in Computer Science*, vol. 3961, pp. 156–165, Nov. 2006.

[108] L. A. Martucci, C. M. Schweitzer, Y. R. Venturini, T. C. Carvalho, and W. V. Ruggiero, "A trust-based security architecture for small and medium-sized mobile ad hoc networks," in *Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net'04)*, (Bodrum, Turkey), June 2004.

[109] A. A. Pirzada and C. McDonald, "Establishing trust in pure ad-hoc networks," in *Proceedings of 27th Australasian Computer Science Conference (ACSC'04)*, (Dunedin, New Zealand), Oct. 2004.

[110] A. A. Pirzada and C. McDonald, "Trust establishment in pure ad-hoc networks," *Wireless Personal Communications: An International Journal*, vol. 37, no. 1-2, pp. 139–168, Apr. 2006.

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

[112] M. Morvan and S. Sené, "A distributed trust diffusion protocol for ad hoc networks," in *International Conference on Wireless and Mobile Communication (ICWMC'06)*, (Bucharest, Romania), July 2006.

[113] K. Wang and M. Wu, "A trust approach for node cooperation in manet," in *International Conference on Mobile Ad-hoc and Sensor Networks (MSN'07)*, (Beijing, China), Dec. 2007.

[114] K. Wang, M. Wu, and S. Shen, "A trust evaluation method for node cooperation in mobile ad hoc networks," in *International Conference on Information Technology: New Generations (ITNG 2008)*, (Las Vegas, USA), Apr. 2008.

[115] J. Liu and V. Issarny, "Enhanced reputation mechanism for mobile ad hoc networks," in *International Conference on Trust Management (iTrust'04)*, (Oxford, UK), Mar. 2004.

[116] Z. Yan, P. Zhang, and T. Virtanen, "Trust evaluation based security solution in ad hoc networks," in *Proceedings of the Seventh Nordic Workshop on Secure IT Systems, (NordSec'03)*, (Gjã¸vik, Norway), Oct. 2003.

[117] J. McGibney, D. Botvich, and S. Balasubramaniam, "A combined biologically and socially inspired approach to mitigating ad hoc network threats," in *IEEE Vehicular Technology Conference (VTC'07 Fall)*, (Baltimore, USA), Oct. 2007.

[118] S. S. Rizvi, S. Poudyal, V. Edla, and R. Nepal, "A novel approach for creating trust to reduce malicious behavior in manet," in *ACM International Conference On Emerging Networking Experiments And Technologies (CoNEXT Student Workshp)*, (New York, USA), Dec. 2007.

[119] S. Buchegger and J. Le. Boudec, "Performance Analysis of the CONFIDANT Protocol Cooperation Of Nodes Fairness In Dynamic Ad-hoc NeTworks," in *IEEE/ACM Symposium on Mobile Ad Hoc Net- working and Computing (MobiHOC'02)*, (Lausanne, Switzerland), June 2002.

[120] S. Bansal and M. Baker, "Observation-based cooperation enforcement in ad-hoc networks," tech. rep., Stanford University, 2003.

[121] P. Michiardi and R. Molva, "CORE: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *IFIP Communication and Multimedia Security Conference*, Sept. 2002.

[122] V. Balakrishnan, V. Varadharajan, U. K. Tupakula, and P. Lucs, “Trust integrated cooperation architecture for mobile ad-hoc networks,” in *IEEE International Symposium on Wireless Communication Systems (ISWCS’07)*, (Trondheim, Norway), Oct. 2007.

[123] V. Balakrishnan, V. Varadharajan, P. Lucs, and U. K. Tupakula, “Trust enhanced secure mobile ad-hoc network routing,” in *International Conference on Advanced Information Networking and Applications Workshops (AINAW ’07)*, (Niagara Falls, Canada), May 2007.

[124] V. Balakrishnan, V. Varadharajan, U. K. Tupakula, and P. Lucs, “Trust and recommendations in mobile ad hoc networks,” in *International Conference on Networking and Services (ICNS’07)*, (Athens, Greece), June 2007.

[125] D. Kostoulas, R. Aldunate, F. P. Mora, and S. Lakhera, “A nature-inspired decentralized trust model to reduce information unreliability in complex disaster relief operations,” *Advanced Engineering Informatics*, vol. 22, no. 1, pp. 45–58, Jan. 2008.

[126] Y. L. Sun, Z. Han, and K. J. R. Liu, “Defense of trust management vulnerabilities in distributed networks,” *IEEE Communiations Magazine*, vol. 46, no. 2, pp. 112–119, Feb. 2008.

[127] S. Buchegger and J.-Y. Le Boudec, “A robust reputation system for P2P and mobile ad-hoc networks,” in *Workshop on the Economics of Peer-to-Peer Systems (P2PEcon’04)*, (Cambridge, USA), June 2004.

[128] D. D. Clark, C. Partridge, J. C. Ramming, and J. T. Wroclawski, “A knowledge plane for the internet,” in *ACM SIGCOMM’03*, (Karlsruhe, Germany), Aug. 2003.

[129] D. F. Macedo, A. L. Santos, and G. Pujolle, “MANKOP: A knowledge plane for wireless ad hoc networks,” in *IEEE/IFIP Network Operations and Management Symposium (NOMS’08)*, (Salvador, Brazil), Apr. 2008.

[130] J. Mundinger and J.-Y. Le Boudec, “Analysis of a reputation system for Mobile Ad-Hoc Networks with liars,” *Performance Evaluation*, vol. 65, no. 3-4, no. 3-4, pp. 212–226, 2008.

[131] R. Braden, D. Clark, S. Shenker, and J. Wroclawski, “Developing a next-generation internet architecture,” tech. rep., Advanced Network Architecture Group, MIT, 2000.

[132] M. Smirnov, “Autonomic communication,” tech. rep., Communication paradigms for 2020, Mar. 2004.

List of acronyms

REP	*Recommendation Exchange Protocol*
AODV	*Ad hoc On-Demand Distance Vector routing protocol*
DSDV	*Destination-Sequenced Distance Vector*
DSR	*Dynamic Source Routing protocol*
HSR	*Hierarchical State Routing*
IP	*Internet Protocol*
LIP6	*Laboratoire d'Informatique de Paris VI*
MAC	*Medium Access Control*
MANET	*Mobile Ad Hoc Networks*
OLSR	*Optimized Link State Routing protocol*
RFC	*Request For Comments*
ZRP	*Zone Routing Protocol*

List of figures

List of tables