A Probabilistic Dynamic Network Trust Model for IoT Systems with Lost Messages and Cyberattacks

Erol Gelenbe
Inst. of Theoretical & App. Informatics
Polish Acad. of Sciences (IITIS-PAN)
44100 Gliwice, PL,
& CNRS I3S, Université Côte d'Azur, 06103 Nice, FR
ORCID:0000-0001-9688-2201

Qixian Ren
Inst. of Theoretical & App. Informatics
Polish Acad. of Sciences (IITIS-PAN)
44100 Gliwice, PL
email: qren@iitis.pl

Zheng Yan
State Key Lab of ISN
Xidian University
Xidian, CN
email: zyan@xidian.edu.cn

***Abstract*—This paper introduces a new dynamic networked trust model, the Random Neural Network Trust Model (RNNTM), which incorporates the dynamics of trust formation in a network through a sequence of "votes" from each entity regarding all other entities. The model assures fairness among entities through a fixed replenishment rate of "voting rights" for each entity, whose voting rights are reduced each time the entity votes. A positive vote received by an entity increases its voting rights and its trustworthiness, while a negative vote reduces its voting rights and also its trustworthiness, and a non-negative integer represents each entity's instantaneous "trust value". An important property of the RNNTM is that an entity cannot express its trust or distrust of the other entities, and hence affect their trust values, when its trust level is down to zero, so that untrustworthy entities are not allowed to express trust or distrust. After developing the theoretical characteristics for the RNNTM model, this paper details its use to evaluate the trust value of multiple entities in a network of Internet of Things (IoT) devices and gateways, where cyberattacks against the gateways, and messages that should be received from IoT devices at regular intervals, modify the parameters that express the trust or distrust between entities. To illustrate its use for a network of interaction gateways, servers and user equipment, several detailed time-dependent simulations of the RNNTM are conducted in the presence of cyberattacks.**

***Index Terms*—Dynamic Trust Models, Random Neural Networks, G-Networks, Analytical Methods for Trust**

I. Introduction

In the large peer-to-peer (P2P) wireless systems of the future, the multi-disciplinary concept of trust will be needed as a tool to determine whether user entities (UE) connect with each other, and how each UE may evaluate the services or data that they request and receive from each other [1]. However, trust is a complicated concept which has emotional [2] and cognitive [3] foundations that have been studied from a multidisciplinary perspective [4]. It is also of great practical value, and is widely discussed in the business world [5] where it is used to establish the competence and believable integrity of a trustworthy person or enterprise, including key aspects such as:

Erol Gelenbe's work was partially supported by the EU Horizon 2020 Project DOSS, Grant Agreement No. Grant Agreement No. 101120270. Qixian Ren has been supported by an IITIS-PAN Research Assistantship.

- Connectedness to others who can corroborate trustworthiness and provide their own evaluations, as well as
- Dependability, which requires that data from multiple opinions and instances is available to establish trust.

Among the known business representations of trust, the *American Express* model is based on (i) Consistency, requiring multiple instances and data regarding the behaviour or performance of the trustworthy entity, (ii) Competence, based on the ability to deliver results within expected or agreed Key Performance Indicators (KPI) and (iii) Caring, a softer consideration that requires empathy and listening to the opinions of others, and also brings to bear the issue of participation in collective and shared decision-making processes.

The need to formulate trust in the Internet goes back to the commercial uses of networks that are enabled by cryptography and trust in our ability to identify ourselves, order goods and services online, and make secure payments [6]–[8]. The many-to-many and peer-to-peer connectivity offered by modern wireless networks, with the increased use of mobile connected devices to express opinions and influence decisions, new trust models emerge [9].

A. Contributions of the Present Paper

The purpose of this paper is to develop an operational model of trust for future networks composed of many connected devices, including servers that aim to produce secure and reliable services to all the other entities, in a manner that limits the influence of untrustworthy entities within the model, and which allows all entities to express their trust or distrust of others by the possibility of voting, which is distributed in a rationed manner to each entity. Trust in a given entity, in our model, is based on the opinion expressed by other entities as well as on the frequency with which each entity expresses its trust or distrust of others. Each entity (a device or server) is fed with a flow of "permits" that allow it to dispense its opinion regarding others, and it can indicate its trust about another entity if it observes that this other entity is accomplishing its normal work, or when it receives a normal message from it, while it may express distrust about the other entity if the other entity becomes non-responsive or if experiences a cyber-attack. Entities that express themselves

more frequently than others can also lose their trust value, so that the model discourages excessive "gossip" or attempts to influence opinions, and entities with low trust value will have a smaller chance of influencing the trust level of others.

To this effect, we propose a new time-dependent network model of trust, the RNN Trust Model (RNNTM), which incorporates the dynamics of trust formation through a sequence of "votes" from each entity to all other entities, that mimic the successive expressions of trust and distrust that any entity may formulate about other entities. The instantaneous trust value of an entity e_i at time $t \geq 0$ is represented as a non-negative integer $K_i(t) \geq 0$. Interactions between entities occur asynchronously at different instants of time, and lead to changes in the value of $K_i(t)$ for each e_i, which also limits the "right to express" itself of each entity so that more trustworthy entities may express themselves more frequently. Furthermore, events such as cyberattacks on certain entities will modify the parameters that express the trust or distrust between entities, which in turn affect the values of each $K_i(t)$.

The original features of the RNNTM, as compared to conventional trust models, are:

- RNNTM treats the trust level of each entity as a "right to vote" which is replenished by a constant rate of rights to vote that arrive externally to entity e_i at a positive constant rate $\Lambda_i > 0$, These rights to vote may be also depleted by a constant (non-negative) flow of deplete messages $\lambda_i \geq 0$. Thus, the parameters $\Lambda_i,\ \lambda_i$ can be viewed as the rules imposed by an external regulator to each entity e_i in the system.
- When $\Lambda_i = \Lambda,\ \lambda_i = \lambda$ for all $1 \leq i \leq n$, this means that all entities are placed on an equal footing regarding the number of voting rights they receive per unit time and the rate at which these voting rights are taken away from them.
- The trust level and right to vote is reduced each time the entity expresses trust or distrust towards another entity, or when some other entity expresses distrust towards the given entity. Thus, the "right to vote regarding the trust of others" is modulated by the current trust level of the voter, and entity e_i cannot express its trust or distrust of the other entities, and hence affect their trust values at some time t, if e_i's trust value i$K_i(t) = 0$. More generally, the capability of e_i's to influence the trust level of other entities e_j depends on the probability that e_i itself has a positive trust level $Prob[K_i(t) > 0]$. Thus, non-trustworthy entities are not allowed to affect the trust level of other entities.
- On the other hand, each time an entity receives an expression of trust from another entity, its trust level increases by one.

Thus, in the RNNTM the trust level of an entity is determined by its external replenishment and deletion rate of its rights to vote, the frequency with which it expresses itself about other entities, and the trust or distrust that is expressed by other entities towards itself, for them that is expressed by other entities, Its purpose is to model the manner in which the mutual trust of a set of entities will evolve over time, in response to the opinions expressed by the different entities with respect to each other.

In Section I-B we review some related work, and in Section II we detail the mathematical structure of the RNNTM and its analytical solution. Section III is devoted to the analysis of the collection of entities in the presence of cyberattacks, and the recovery from attacks, which modify the interaction rates between entities. An example of the use of the RNNTM when a collection of devices or servers can be subjected to cyberattacks is detailed in Section IV. Finally, conclusions and suggestions for further work are provided in Section V.

B. Related Work

A recent discussion paper [10] recalls the role of trust as an element of human and institutional identity, and points to an analysis [11] that stresses the importance of Temporal Embeddedness, implying that a trustworthy party can benefit from another entity's trust into the future, and hence value and nurture the trust being placed in itself, while Social Embeddedness allows the trusted entity to benefit from the propagation of its trustworthiness through social networks. On the other hand, Institutional Embeddedness of trust refers to stable social institutions, such as the legal system and the courts of law, regulatory bodies, professional organizations and universities, that can certify trustworthiness within specific contexts through the award of certifications and degrees related to knowledge and professional capabilities, and propose codes of behaviour that can reinforce the role of trust within human society.

Early work has discussed how interactions in social networks can enhance trust relations [12], and an analysis of the links between social networks and trust was examined in the context of medical practice [13]. On-line feedback that replaces human interaction for the establishment and management of reputation was considered in [14], while the manner in which specialized "recommender agents" can be constructed and used is examines in [15].

More recent work has examined how trust representations can enhance collective intelligence and successful search in social systemes [16]. In [17], the analytical techniques that can help evaluate reputation in peer-to-peer systems are discussed, while other work has studied the effects of personalities and human bias on the dyamics of trust [18], [19].. The related computer science literature [20], [21] has often used trust in relation to the Internet of Things (IoT) [22]. Trust was also used for social networks [23], because it influences the manner in which information spreads in peer to peer (P2P) networks [24]. Since trust among n entities is often represented by a directed "trust graph" (TG) where arcs represent the trust of some entity regarding another entity, much work uses sets of TGs to learn the trust values from data with machine learning (ML) [25], and test whether certain TGs conform with the data, or to discover trust relationships which disagree with given datasets [26], [27].

II. Dynamics of the RNN Trust Model (RNNTM)

The RNN Trust Model (RTM) is a computational model for the trust that is associated with a set of n entities $E = \{e_1, \ ... \ , e_n\}$ where the trust for entity e_i at some real-valued time $t \geq 0$ is expressed as a non-negative integer $K_i(t) \geq 0$ which indicates that the entity cannot be trusted at all at time t if $K_i(t) = 0$. On the other hand, if $K_i(t) > 0$ then it is worthy of some trust at time t, described by the value of $K_i(t)$. Thus, the trust in all of the entities at time t is given by the $n - vector \ K(t) = (K_1(t), \ ... \ K_n(t))$, and as we shall see below the trust of each entity depends on the trust value of the other entities. The RTM which concerns all the n entities can be installed as a software API in **each** of the n entities, and each of the entities can use the same rules for updating it based on external events. These external events include periodic broadcast messages, sent for instance every ten seconds, from each entity to all of the other entities. The repeated lack of an acknowledgement message in response to a sent message, or a lack of messages coming from a specific entity, could then be viewed as an indication of a malfunction or cyberattack concerning the non-responsive entity, which would then result in a reduction of the trust that is associated with it.

The trust system we describe is affected by external prior knowledge represented by the real-valued **external trust** parameter $\Lambda_i \geq 0$ and the **external distrust** parameter $\lambda_i \geq 0$, for each entity $e_i, \ i = 1, \ ... \ , n$, and by the parameters that govern the interactions between entities that are defined below. In our system, entity e_i can express an opinion at time t about some other entity as long its trust value is positive, i.e. $K_i(t) > 0$; when it does so, its own trust level drops by one, I.e. $K_i(t^+) = K_i(t) - 1$. Thus each entity has a "number of voting rights" $K_i(t)$ about other entities which is identical to its own trust value. Thus the higher its own the value $K_i(t)$ is, the more votes e_i has to express itself regarding trust or distrust of others.

Thus, the trust level $K_i(t)$ of entity e_i is also the number of "votes" or expressions of trust or distrust that it is allowed to express about other entities at a given time; this resembles a "plutocracy of trust", where trustworthy individuals are allowed to more frequently express their trust or distrust of others. In this model, a probabilistic $n \times n$ connection matrix $P^+ = [p^+_{ij}]$ also represents for each entity e_i the probability that it may express trust about another entity e_j, and similarly the probabilistic $n \times n$ connection matrix $P^- = [p^-_{ij}]$ represents for each entity e_i the probability that it may express distrust about another entity e_j. These matrices are constrained as follows for each $i = 1, \ ... \ n$:

$$p^+_{ij} \geq 0, p^-_{ij} \geq 0, p^+_{ii} = p^-_{ii} = 0, \ \sum_{j=1}^{n} [p^+_{ij} + p^-_{ij}] = 1, \quad (1)$$

representing the opinion of each entity e_i regarding its trust or distrust for other entities. Finally, each entity e_i has a specific rate r_i or speed at which it may express its trust or mistrust about another entity. We use these parameters to define the "weights" with which each entity expresses its trust or distrust concerning other entities:

$$w^+_{ij} \equiv r_i p^+_{ij}, \ w^-_{ij} \equiv r_i p^-_{ij}, \ and \ r_i = \sum_{j=1}^{n} [w^+_{ij} + w^-_{ij}]. \quad (2)$$

In general, the weights $w^+_{ij}, \ w^-_{ij}$ may change, and the parameters $\Lambda_i, \ \lambda_i$ may be updated or changed during the long periods of usage of a given model. For instance, a cyberattack on an entity may result in a loss of trust by the other entities towards the entity that has been the victim of a successful cyberattack, because the success of the attack implies that the entity was not well protected to detect or mitigate a cyberattack, and after an attack the entity itself may be compromised.

In the following, we will use the notation $[X]^+$, which is commonly defined as $[X]^+ = X, \ when \ X \geq 0$ and $[X]^+ = 0, \ when \ X < 0$. The n entities interact with each other using the parameters that we have defined, in the following manner at any given time t:

$$\begin{aligned} &K_i(t + \Delta t) = K_i(t) + 1 \\ &with \ probability \ \Lambda_i \Delta t + o(\Delta t), \quad (3) \\ &K_i(t + \Delta t) = [K_i(t) - 1]^+ \\ &with \ probability \ \lambda_i \Delta t + o(\Delta t), \quad (4) \\ &If K_j(t) > 0, \ then \ K_i(t + \Delta t) = K_i(t) + 1 \\ &with \ probability \ r_j p^+_{ji} \Delta t + o(\Delta t), \quad (5) \\ &If K_j(t) > 0, \ then \ K_i(t + \Delta t) = [K_i(t) - 1]^+ \\ &and \ K_j(t + \Delta t) = K_j(t) - 1 \\ &with \ probability \ r_j p^-_{ji} \Delta t + o(\Delta t). \quad (6) \end{aligned}$$

Thus (3) indicates that the external opinion of trust Λ_i regarding e_i increases its trust level by one, while the external opinion of distrust λ_i reduces it by one, as indicated in (4). The expression of trust by some entity e_j for e_i will increase its trust level by one as shown in (5), while the expression of distrust will reduce its trust level by one, as in (6).

Using this definition of the RNNTM, and the results from [28], [29], the following key result allows us to compute the trust value in steady state for a set of n interacting entities:

Theorem Let:

$$q_i = \lim_{t \to \infty} Prob[K_i(t) > 0], \ 1 \leq i \leq n. \quad (7)$$

Then if the solution to the following non-linear system of equations exists:

$$q_i \equiv \frac{\Lambda_i + \sum_{j=1}^{n} q_j w^+_{ji}}{r_i + \lambda_i + \sum_{j=1}^{n} q_j w^-_{ji}} < 1, \quad (8)$$

such that $0 \leq q_i < 1$, for $1 \leq i \leq n$, then:

$$\begin{aligned} &\lim_{t \to \infty} Prob[K_1(t) = k_1, \ ... \ , K_n(t) = k_n] \\ &= \Pi_{i=1}^{n} (1 - q_i) q_i^{k_i}, \quad (9) \\ &and \ \lim_{t \to \infty} E[K_i(t)] = \frac{q_i}{1 - q_i} \ . \quad (10) \end{aligned}$$

Comment: For a particular model or application, one can set thresholds for trustworthiness such as:

- e_i is untrustworthy if $q_i \leq \theta_1$,
- e_i is undetermined if $\theta_1 < q_i \leq \theta_2$,
- e_i is trustworthy if $q_i > \theta_2$, where:

$$0 \leq \theta_1 < \theta_2 < 1 \ . \tag{11}$$

A. Initialization

In an initial situation where we have no evidence regarding whether any of the entities should be trusted or not, we will initialize the parameter values as follows:

- We set the values $w_{ij}^+ = w_{ij}^- = w,\ \lambda_i = \lambda,\ \Lambda_i = \Lambda$ for all distinct pairs of entities $e_i,\ e_j,\ i \neq j$.
- To show "perfect ignorance" we also select $q_i = 0.5,\ i = 1,\ \ldots\ , n$ representing the probability of whether any entity e_i is trustworthy or not.

Thus, using (7), we can seek the set of parameter values that we should take by using (8) and setting:

$$0.5 = \frac{\Lambda + 0.5(n-1)w}{2(n-1)w + \lambda + 0.5(n-1)w} \ , \tag{12}$$

which yields:

$$2\Lambda + (n-1)w = 2(n-1)w + \lambda + 0.5(n-1)w \ , \tag{13}$$

so that:

$$\Lambda = 0.75(n-1)w + 0.5\lambda \ . \tag{14}$$

To simplify the calculations we set $\lambda = 0$ and $w > 0$ can be fixed at any convenient positive value so we obtain:

$$\Lambda = 0.75(n-1)w,\ with\ \lambda = 0,\ and\ w > 0 \ . \tag{15}$$

In the sequel, we will assume that w in (15) is chosen such that:

$$w_{ij}^+ + w_{ij}^- = 2w,\ \ \forall\ i,j = 1,\ ...n,\ i \neq j,\ w = 1\ and$$
$$r_i = 2(n-1),\ \Lambda_i = 0.75(n-1),\ i = 1,\ \ldots\ , n. \tag{16}$$

With this initialization, we also need to understand the maximum values that can be allowed for any $w_{ij}^+,\ w_{ij}^-$ for $i \neq j$. We know from (8) that q_i is an increasing function of each w_{ji}^+ when $w_{ji}^+ + w_{ji}^- = 2$ as fixed in (16), and that the maximum value that q_i cannot exceed 1 since it is a probability. Therefore, we compute the value $M,\ 0 \leq M \leq 2$ of each w_{ji}^+ that cannot be exceeded, by setting all the $q_i = 1$ with the parameter values that have been chosen in (15) and (16). We then use (8) to obtain:

$$2(n-1) + (n-1)(2-M) = 0.75(n-1) + M(n-1),$$
$$and \quad M = 1.625 \ . \tag{17}$$

Since we must maintain $q_i < 1,\ \forall i$, we set the maximum and minimum values $\forall i, j,\ i \neq j$ to:

$$0 \ \leq\ w_{ij}^+ \ \leq\ 1.55,\ and\ 0.45 \ \leq\ w_{ij}^- \ \leq\ 2. \tag{18}$$

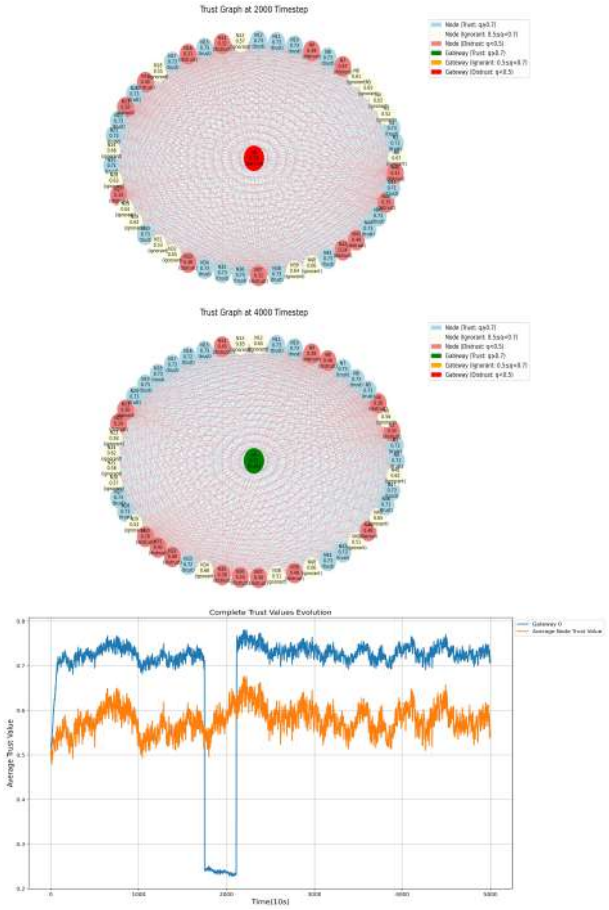

Fig. 1. The top two figures show the RNNTM Trust Graph for a total of 49 IoT devices numbered $N1$ to $N49$ shown around the circle, and a single Gateway $N0$ at the center, sampled at time units 2000 and 4000, where one time unit is 10 seconds. We have set $\theta_1 = 0.5$ and $\theta_2 = 0.7$ in (11). The colour code is Red if the Gateway is deemed untrustworthy, and Green if the Gateway is deemed trustworthy. Trustworthy IoT nodes are colored Light Blue, untrustworthy IoT nodes are colored Pink, and IoT devices whose status is "undecided" are colored Yellow. In these simulations, a randomly chosen 20% of IoT nodes omit to send a message each time unit (i.e., each 10 sec), which affects the trustworthiness of all nodes, since when a message is not received from some node, the assumption is that it has been attacked and other IoT nodes or Gateways will update their weights accordingly as detailed in Section III. The second figure from the top shows that the Gateway has been attacked some time before 2000 time units, and the third figure from the top shows that it has recovered some time later. For exactly the same simulation as the other figures, the bottom figure shows (in Orange) that the **average** trust level of all IoT nodes varies with time for each subsequent 10 second time unit; it also shows that the the instantaneous trustworthiness level of the Gateway (Blue) which suffers a cyberattack some time after 1700 time units, and that it then recovers from the attack several hundred time units later.

III. The RNNTM for Cyberattacks on IoT Gateways

In this section, we present an application of the RNN Trust Model (RTM) that was introduced in the previous section in an environment where denial of service (DoS) attacks occur against some entity e_i. We assume that a DoS attack against an entity e_i occurs at random and unexpectedly on average every $\frac{1}{\alpha_i}$ time units, where $\alpha_i > 0$ can be interpreted as the attack rate, i.e. the average number of attacks per unit time.

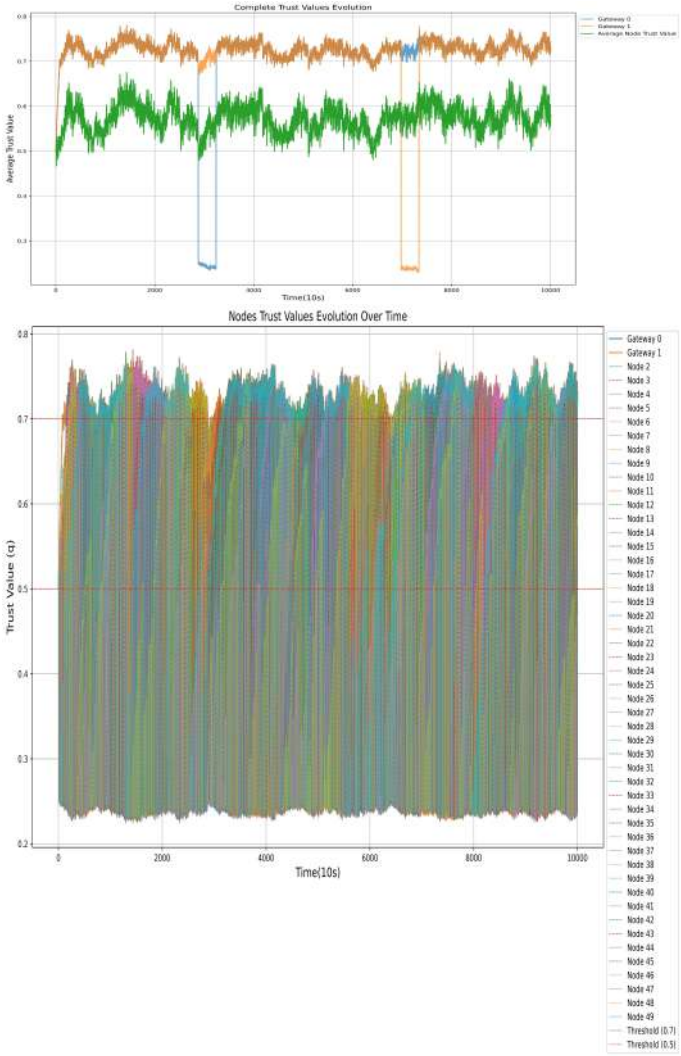

Fig. 2. The simulation results in this figure relate to a 50 node system with two gateways $N0$ and $N1$, where each of the nodes or entities is expected to send a packet into the network at each of the discrete time units (each 10 seconds). We use all the previously defined and fixed parameters throughout the simulation. Each of the two gateways is independently subject to intermittent DoS attacks that affect their trust values. The remaining 48 IoT devices have each (and independently of each other) a probability of 0.2 that any one of them may omit to send a packet into the network. The gateway $N0$ suffers an attack at approximately the 2800 successive 10 second interval, while $N1$ is subject to an attack at approximately the 6900 time instant. The top figure shows (i) the instantaneous average trust value at each time instant of the two gateways when they are NOT under attack (in Brown), (ii) the instantaneous trust value of the gateway being attacked, which dips sharply (in Blue for $N0$ or in Orange for $N1$) during the attack and during the recovery from the attack, as well as trust level of the gateway that is not under attack (in Blue for $N0$ or in Orange for $N1$). The figure below shows the widely varying trust values of the 48 remaining IoT devices throughout the simulation.

Equivalently, we can consider that in a time interval $[t, t + \Delta t[$, the probability of a DoS attack against e_i is $\alpha_i \Delta t + o(\Delta t)$, Note that the time scale of the weight parameters, i.e. $\frac{1}{w_{ij}^+}$, $\frac{1}{w_{ij}^-}$ and hence of the parameters $\frac{1}{r_i}$, as well as $\frac{1}{\Lambda_i}$, $\frac{1}{\lambda_i}$, would be in the range of seconds or tens of seconds, since the RTM is updated frequently (say each 10 seconds) whenever the communication updates are sent and received by the different entities e_i.

On the other hand, the time that elapses between cyberattacks may be hours, days or even weeks (e.g. once every two or three weeks). After an attack, the entity e_i that came under attack will have to recover, and this will take a time of average length T_i which may be as long an hour or more, so that in practice we have $\frac{1}{\alpha_i} > T_i$ and $\frac{1}{\alpha_i} >> \frac{1}{r_i}$, $\frac{1}{\Lambda_i}$, $\frac{1}{\lambda_i}$. Thus, the RNNTM model will typically have reached its steady-state probability distribution between two successive cyberattacks.

A. Effect of the Cyberattack

When a DoS attack occurs against any entity e_i, the attacked entity becomes unavailable and cannot communicate with the other entities. Also, until it recovers from the cyberattack it will not be able to modify its outgoing weights $w_{ij}^+,\ w_{ij}^+,\ j \neq 1$. On the other hand, each (other) entity $e_j,\ j \neq 1$ reacts to the lack of communication from e_i by modifying its connection weights towards e_i as follows:

$$w_{ji}^+ \leftarrow w_{ji}^+ - \eta_{ji},\ w_{ji}^- \leftarrow w_{ji}^- + \eta_{ji}, \tag{19}$$

where $0 < \eta_{ji} \leq w_{ji}^+$. As a result, assuming $\lambda_i = 0$ for all i as suggested in the initialization, after an attack the trust probability q_i of e_i obtained from (8) is updated to the value q_i^u of e_i as follows:

$$\begin{aligned} q_i^u &= \frac{\Lambda_i + \sum_{j=1, j\neq i}^n q_j (w_{ji}^+ - \eta_{ji})}{r_i + \sum_{j=1, j\neq i}^n q_j (w_{ji}^- + \eta_{ji})}, \\ &= \frac{\Lambda_i + \sum_{j=1, j\neq i}^n q_j w_{ji}^+}{r_1 + \sum_{j=1, j\neq i}^n q_j w_{ji}^-} \times \frac{1 - \frac{\sum_{j=1, j\neq i}^n q_j \eta_{ji}}{\Lambda + \sum_{j=1, j\neq i}^n q_j w_{ji}^+}}{1 + \frac{\sum_{j=2}^n q_j \eta_{ji}}{r_1 + \sum_{j=2}^n q_j w_{ji}^-}}, \\ &= q_i \times \frac{1 - \frac{\sum_{j=1, j\neq i}^n q_j \eta_{ji}}{\Lambda + \sum_{j=1, j\neq i}^n q_j w_{ji}^+}}{1 + \frac{\sum_{j=1, j\neq i}^n q_j \eta_{ji}}{r_1 + \sum_{j=1, j\neq i}^n q_j w_{ji}^-}} < q_1\ . \end{aligned} \tag{20}$$

We will set $\eta_{ji} = w_{ji}^+$ which is its maximum value. From (16) we have the value $w_{ij}^+ + w_{ij}^- = 2$ for all $i \neq j$. When we also use (15), we obtain the updated Trust Probability for the attacked entity e_i as:

$$\begin{aligned} q_i^u &= \frac{0.75(n-1)}{2(n-1) + 2\sum_{j=1, j\neq i}^n q_j}\ , \\ &= \frac{0.375(n-1)}{(n-1) + \sum_{j=1, j\neq i}^n q_j}\ . \end{aligned} \tag{21}$$

B. Recovery from a Cyberattack

After a cyberattack recovery time of average length T_i, the entity e_i that has been attacked recovers from the attack and sends "all clear" messages to all the other entities. These other entities will then change their weights in successive steps $s =$

1, 2, 3, ..., following each of the messages that arrive from e_i to e_j. These weight changes take the following form:

$$w_{ji}^{[+,s]} = w_{ji}^{[+,(s-1)]} + [\frac{\eta_{ji}}{1+\eta_{ji}}]^s \ ,$$

$$= w_{ji}^{+} - \eta_{ji} + \sum_{s=1}^{\infty} [\frac{\eta_{ji}}{1+\eta_{ji}}]^s,$$

$$w_{ji}^{[-,s]} = w_{ji}^{[-,(s-1)]} - [\frac{\eta_{ji}}{1+\eta_{ji}}]^s \ ,$$

$$= w_{ji}^{-} + \eta_{ji} - \sum_{s=1}^{\infty} [\frac{\eta_{ji}}{1+\eta_{ji}}]^s, \quad (22)$$

$$where\ w_{ji}^{[+,0]} = w_{ji}^{+} - \eta_{ji},\ w_{ji}^{[-,0]} = w_{ji}^{-} + \eta_{ji},$$

$$hence\ \lim_{s\to\infty} w_{ji}^{[+,s]} = w_{ji}^{+},\ \lim_{s\to\infty} w_{ji}^{[-,s]} = w_{ji}^{-}. \quad (23)$$

The simulation results presented in this paper use the settings in (16), $0 \leq w_{ij}^{+} \leq 1.55$ from (18) and $\eta_{ji} = w_{ji}^{+}$ which yield:

$$w_{ji}^{[+,(0)]} = 0,\ w_{ji}^{[-,(0)]} = 2,\ w_{ji}^{[+,(0)]} + w_{ji}^{[-,(0)]} = 2. \quad (24)$$

We can also calculate τ_i, the average trustworthiness of e_i over a long length of time which includes alternating periods when the system has been attacked and is recovering and when it is operating normally:

$$\tau_i \approx \frac{\alpha_i T_i}{(\alpha_i T_i + 1)} \frac{q_i^u}{1 - q_i^u} + \frac{1}{(\alpha_i T_i + 1)} \frac{q_i}{1 - q_i} \ . \quad (25)$$

Note that (25) is an approximate expression which neglects the gradual growth of the trust that each entity has in e_i after an attack occurs, until the "all clear" signal is broadcast by e_i.

IV. Simulation of Trustworthiness in the Presence of Lost Messages and Cyberattacks

In the simulation experiments of this section, we show the evolution of the coupled trust values for 50 entities or nodes over time, in an IoT network. All the simulations proceed in successive time units, each equal to 10 seconds, and the corresponding trust evaluations are recorded and shown in Figures 1, 2 and 3. Note that in all the simulations, all the 50 trust values are all coupled as described in the previous sections. All entities are expected to broadcast messages to all other entities at each time unit, and trust is updated for each entity (gateway and IoT devices) based on this communication behaviour.

The first setup includes one IoT Gateway (denoted node $N0$) and 49 IoT devices ($N1$ to $N49$), and its results are shown in Figure 1. The average trust value is calculated for all of the 49 IoT devices, and the gateway's trust value is also computed at each time unit. At each time unit, 20% of the IoT randomly fail to send messages. Thus, at each time unit, each IoT device may fail to send its message with probability 0.2. This mimics sporadic communication errors or the packet loss of the IoT devices. When a device "forgets" to send a message, its trust value drops. Since all messages are broadcast to all entities, this behaviour is known to all other devices or gateways, modifying the trust assessments of other devices,

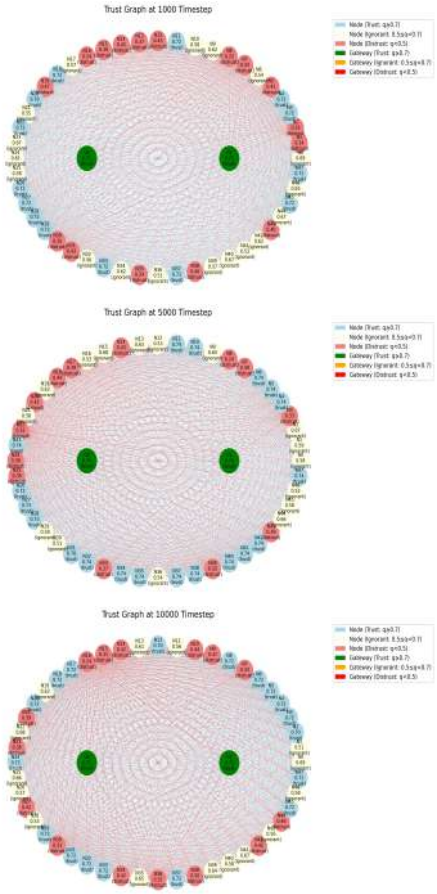

Fig. 3. The three figures shown above represent the RNNTM Trust Graph for a total of 50 entities, including 48 IoT devices numbered $N2$ to $N50$ shown around the circle, and the two gateways $N0$, $N1$ at the center, sampled at the time units 1000, 5000 and 10,000, where each time unit is 10 seconds, and $\theta_1 = 0.5,\ \theta_2 = 0.7$as in (11). The colour code is Red for an untrustworthy gateway, and is Green when the gateway is deemed trustworthy. Trustworthy IoT nodes are cf Light Blue color, untrustworthy IoT nodes are Pink, and IoT devices whose status is the "undecided" trustworthiness levels are colored Yellow. Each IoT device has a probability of 0.2 that it omits to send a message in any time unit, and all other IoT nodes or Gateways update their weights accordingly, as described in Section III. These diagrams allow us to observe the changing trust levels for all of the entities over a total period of 100,000 seconds.

and it propagates across the network's trust landscape. After a device "forgets" to send a message, if it resumes its message sending it will take several time units to recover its trust in successive steps. Also, at epoch 1769, a simulated attack targets the gateway, resulting in a shutdown lasting 360 time units. This causes the gateway trust value to decline sharply, reaching a minimum of around 0.25. During the shutdown, the gateway ceases to interact with other entities, simulating real-world conditions such as a DoS (Denial of Service) attack.

The simulation results in Figures 2 and 3 concern a 50 node system with two gateways $N0$ and $N1$, where each of the nodes or entities is expected to send a packet into the network at each of the discrete time units (each 10 seconds). We use all the previously defined and fixed parameters throughout the simulation. Each of the two gateways is independently subject

to intermittent DoS attacks that affect their trust values. The remaining 48 IoT devices have each (and independently of each other) a probability of 0.2 that any one of them may omit to send a packet into the network. When a gateway suffers from a cyberattack, its usage is blocked for some time, it does not send packets, and this results in a loss of trust on the part of the remaining nodes. The algorithms in the previous sections are used to update the trust values of all of the entities. After an attack, the victim gateway will be able to recover after a certain amount of time, and its trust value will be upgraded accordingly. The gateway $N0$ suffers an attack at approximately the 2800 successive 10 second interval, while $N1$ is subject to an attack at approximately the 6900 time instant. The top figure of Figure 2 shows (i) the instantaneous average trust value at each time instant of the two gateways when they are NOT under attack (in Brown), (ii) the instantaneous trust value of the gateway being attacked, which dips sharply (in Blue for $N0$ or in Orange for $N1$) during the attack and during the recovery from the attack, as well as trust level of the gateway that is not under attack (in Blue for $N0$ or in Orange for $N1$). The figure below shows the wide-ranging and varying trust values of each of the 48 remaining IoT devices (nodes) in the system for each successive instant of the simulation. The five diagrams shown in Figure 3 summarize the trustworthiness of all of the 50 entities over a very long period of $100,000$ seconds.

V. Conclusions and Future Work

The concept of trust among a finite number of entities is often represented by a directed graph whose nodes represent the entities, and labels as well as numerical values on the arcs to represent the type and level of trust that is expressed by some entities regarding some other entities. Various algorithmic techniques can then be used to deduce the level of trust that is enjoyed by each of the entities that are represented in the graph.

In this paper, we introduce the RNNTM that aims to represent trust as a dynamic time-dependent quantity of the different entities. In the RNNTM, the trust level of an entity is determined by its external replenishment and deletion rate of its rights to vote, the frequency with which it expresses itself about other entities, and the trust or distrust that is expressed by other entities towards itself, for them that is expressed by other entities. Its purpose is to model the evolving trust level of a set of entities, in response to the opinions expressed by the different entities with respect to each other, and by theirown behaviour

The RNNTM aims to allow the expressions of trust to vary over time as a function of various significant events, such as the exchange of information through message broadcasts, and the possible existence of external adversarial effects, such as cyberattacks, which will affect the trust that can be attributed to different entities. We have therefore constructed a mathematical model where trust levels of each entity are time-varying, where all entities are treated fairly by the attribution of an equal number of "voting rights per unit time" to all of them, and the possibility for each of the entities to express both trust (a positive vote) and distrust (a negative vote) to each other, while each time an entity gives its opinion it also reduces its ability for further votes. In the resulting dynamics, it turns out that trustworthy entities have more impact on the overall resulting "opinion" about other entities. The details of the model are developed, and are then illustrated with simulations regarding a network with IoT devices and gateways, which can be subject to communication errors and cyberattacks. The RNNTM is also a machine learning system [30], [31], so that future work may use existing data about trust evaluations to estimate the parameters of the RNNTM, and match predictions from existing measured data. Furthermore, it will be useful to consider other "voting modalities" which can be obtained by extending the RNNTM to cases where entities are required to express clear preferences, e.g., trust in some of the other entities and distrust in all of the others. Another possible extension may include the rationing of voting rights based on the energy consumption of each entity [32].

Since this model, and the examples we have developed in this paper, show how such a system can result in significant time variations of the trust level, in future work, we plan to show how the RNNTM may be used to dynamically decide about how different IoT gateways may be chosen as a result of their dynamically varying trust metrics. We will also examine how these dynamic trust variations may result in workload imbalance and additional delays in data processing that is needed by the IoT devices from the IoT gateways.

References

[1] E. Gelenbe, "Users and services in intelligent networks," *IEE Proceedings - Intelligent Transport Systems*, vol. 153, no. 3, pp. 213–220, 2006. [Online]. Available: http://www2.ee.ic.ac.uk/publications/p4923.pdf

[2] K. Jones, "Trust as an affective attitude," *Ethics*, vol. 107, no. 1, pp. 4–25, 1996.

[3] R. Hardin, *Trust and Trustworthiness*. Russel Sage Foundation, New York, New York, USA, 2002.

[4] B. G. Robbins, "What is trust? A multidisciplinary review, critique, and synthesis," *Sociology Compass*, vol. 10, no. 10, p. 972–986, 2016.

[5] J. Blakey, "Comparing trust models based on the nine habits of trust," December 2022. [Online]. Available: https://trustedexecutive.com/comparing-trust-models-based-on-the-nine-habits-of-trust/

[6] S. P. Marsh, *Formalizing Trust as a Computational Concept (PhD thesis)*. University of Stirling, Department of Computer Science and Mathematics, 1994. [Online]. Available: https://www.cs.stir.ac.uk/~kjt/techreps/pdf/TR133.pdf

[7] D. M. Romano, *The Nature of Trust: Conceptual and Operational Clarification (PhD thesis)*. Louisiana State University, 2003. [Online]. Available: https://repository.lsu.edu/gradschool_dissertations/2674

[8] J. Seigneur, *Trust, Security and Privacy in Global Computing (PhD thesis)*. University of Dublin, Trinity College, 2005.

[9] J. Sabater and C. Sierra, "Review on computational trust and reputation models," *Artificial Intelligence Review*, p. 33–60, 2005.

[10] P. Smart, B. Pickering, M. Boniface, and W. Hall, "Risk models of national identity systems: A conceptual model of trust and trustworthiness." [Online]. Available: https://www.turing.ac.uk/sites/default/files/2021-11/technical_briefing_a_conceptual_model_of_trust_and_trustworthiness.pdf

[11] J. Riegelsberger, M. A. Sasse, and J. D. McCarthy, "The mechanics of trust: A framework for research and design," *International Journal of Human–Computer Studies*, vol. 62, no. 3, pp. 381–422, 2005.

[12] B. Esfandiari and S. Chandrasekharan, "On how agents make friends: Mechanism for trust acquisition," in *Proceedings of the Fourth Workshop on Deception Fraud and Trust in Agent Societies*, 2001, p. 27–34.

[13] J. Scott, "Social network analysis as an analytic tool for interaction patterns in primary care practices," *Annals of Family Medicine*, vol. 3, no. 5, p. 443–448, 2005.

[14] C. Dellarocas, "The digitization of Word of Mouth: Promise and challenges of online feedback mechanisms," *Management Science*, vol. 49, no. 10, pp. 1407–1424, 2003. [Online]. Available: http://www.jstor.org/stable/4134013.C

[15] M. Montaner, B. Lopez, and J. De La Rosa, "Developing trust in recommender agents," in *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-02), Part 1*, 2002, pp. 304 – 305. [Online]. Available: https://doi.org/10.1145/544741.544811

[16] P. Dondio and L. Longo, "Trust-based techniques for collective intelligence in social search systems," in *Next Generation Data Technologies For Collective Computational Intelligence. Studies in Computational Intelligence*, vol. 352. Springer, 2011, p. 113–135.

[17] B. Lagesse, "Analytical evaluation of p2p reputation systems," *International Journal of Communication Networks and Distributed Systems*, vol. 9, p. 82–96, 2012.

[18] M. Hoogendoorn and S. W. Jaffry, "The influence of personalities upon the dynamics of trust and reputation," in *2009 International Conference on Computational Science and Engineering*, vol. 3, 2009, p. 263–270.

[19] M. Hoogendoorn, S. W. Jaffry, P.-P. van Maanen, and J. Treur, "Modelling biased human trust dynamics," *Web Intelligence and Agent Systems*, vol. 11, no. 1, pp. 21–40, 2013.

[20] M. Momani and S. Challa, "Survey of trust models in different network domains," *CoRR*, vol. abs/1010.0168, 2010. [Online]. Available: http://arxiv.org/abs/1010.0168

[21] J.-H. Cho, K. Chan, and S. Adali, "A survey on trust modeling," *ACM Comput. Surv.*, vol. 48, no. 2, Oct. 2015. [Online]. Available: https://doi.org/10.1145/2815595

[22] D. Ferraris, C. Fernandez-Gago, and R. Roman, et al., "A survey on IoT trust model frameworks," *Journal of Supercomputing*, vol. 80, p. 8259–8296, 2024. [Online]. Available: https://doi.org/10.1007/s11227-023-05765-4

[23] J. Wang et al., "A survey on trust models in heterogeneous networks," *IEEE Communications Surveys and Tutorials*, vol. 24, no. 4, pp. 2127–2162, 2022.

[24] Z. Yan, P. Zhang, and A. V. Vasilakos, "A survey on trust management for internet of things," *Journal of Network and Computer Applications*, vol. 42, p. 120–134, 2014.

[25] J. Wang, X. Jing, Z. Yan, Y. Fu, W. Pedrycz, and L. T. Yang, "A survey on trust evaluation based on machine learning," *ACM Computing Surveys*, vol. 53, no. 5, p. 1–36, 2020.

[26] J. Wen, et al., "Dtrust: Toward dynamic trust levels assessment in time-varying online social networks," in *Proceedings of IEEE INFOCOM*. IEEEXplore, 2023, pp. 1–10.

[27] T. Luo, J. Wang, Z. Yan, and E. Gelenbe, "Graph neural networks for trust evaluation: Criteria, state-of-the-art, and future directions," *IEEE Network*, March 2025. [Online]. Available: https://drive.google.com/file/d/197wp5FbtS7axaNz9fYW_evMVJIdppbU6/view

[28] E. Gelenbe, "Random neural networks with negative and positive signals and product form solution," *Neural Computation*, vol. 1, no. 4, pp. 502–510, 1989. [Online]. Available: https://doi.org/10.1162/neco.1989.1.4.502

[29] ——, "G-networks with triggered customer movement," *Journal of applied probability*, vol. 30, no. 3, pp. 742–748, 1993.

[30] ——, "Learning in the recurrent random neural network," *Neural Computation*, vol. 5, no. 1, pp. 154–164, 1993.

[31] E. Gelenbe and Y. Yin, "Deep learning with random neural networks," in *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016, pp. 1633–1638. [Online]. Available: https://www.researchgate.net/publication/305581794_Deep_Learning_with_Random_Neural_Networks

[32] E. Gelenbe, "Energy packet networks: Ict based energy allocation and storage," in *Green Communications and Networking*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 186–195.