T2D: A Peer to Peer trust management system based on Disposition to Trust

Rachid Saadi, Jean-Marc Pierson, Lionel Brunie

T2D: A Peer to Peer trust management system based on Disposition to Trust

Rachid Saadi
ARLES Project-Team
INRIA Rocquencourt
Paris, France
rachid.saadi@inria.fr

Jean-Marc Pierson
IRIT Lab
University Paul Sabatier
Toulouse, France
pierson@irit.fr

Lionel Brunie
LIRIS Lab
INSA de Lyon
Lyon, France
lionel.brunie@liris.cnrs.fr

ABSTRACT

While the trust paradigm is essential to broadly extend the communication between the environment's actors, the evaluation of trust becomes a challenge when confronted with initializing the trust relationship and validating the transitive propriety of trust. Whether between users or between organizations, existing solutions work to create for peer to peer networks, flexible and decentralized security mechanisms with trust approach. However, we have noticed that the trust management systems do not make the most of the subjectivity, more specifically, the notion of **Disposition to Trust** although this aspect of subjectivity has a strong influence on how to assess direct and a transitive trust. For this reason in our study, we tackle this problem by introducing a new distributed trust model called **T2D (Trust to Distrust)** which is designed to incorporate the following contributions : (i) A behavior model which represents the Disposition to Trust ; (ii) Initialization of trust relationship (direct and transitive) according to the defined behavior model.

Categories and Subject Descriptors

C2.0 [**Computer Communication Networks**]: General—*Security and protection*; K.4.4 [**Computers and Society**]: Electronic Commerce—*Security*

General Terms

Security, Trust, Theory

Keywords

P2P network, Disposition to trust, Subjectivity, Trust Propagation, Trust Transitivity

1. INTRODUCTION

In the domain of social sciences, there has been substantial research regarding the concept of trust. The findings have been applied in areas including economics, finance, management, government, and psychology. In recent years, trust has acquired considerable interest in the computer science community as the basis of security solutions to extend communication in peer to peer topology.

The majority of trust management systems overlook the initialization of trust relationship processes, they consider that is not a problem of the model or else in the best of circumstances they start all trust relations from zero. But effectively the evaluation of the trustworthiness is not a straightforward task. For instance on a range of 0 to 9, with 0 representing blind trust, should a trustworthy node be rated as 1, 2 or 3? The task is further complicated when the number of nodes to be evaluated is large.

Moreover, in peer to peer networks, existing approaches compute similarly trust evaluation processes for computing a direct or transitive trust relation. We believe that this evaluation should be adjusted according to the personality of each actor (trustful or distrustful). In fact, a trustful personality will give more confidence for a trust relation than a distrustful one.

Thus, we should develop methods that work under the assumption that the Disposition to trust (personality) has a strong influence for both initialization and propagation trust relationships. In this case, the main difficulties are to define a generic model which is able to represent a disposition to trust from a trustful to distrustful behavior. Then, use this model to initialize direct trust and evaluate transitive trust relation.

In our work, we aim to provide a customizable trust management system which makes the most of the Disposition to trust property. Thus, the paper is organized as follows: after exploring the related work (in next section) concerning the disposition and the evaluation of trust, in section 3 we introduce the basics of our T2D model and show how the Disposition to Trust behavior in defined to customize the evaluation of direct trust (in section 4) and transitive trust (in section 5). Finally before concluding and introducing future works, in section 6 we define a new simulator for generating a trust peer to peer networks in order to perform some experiments.

2. BACKGROUND

2.1 Disposition to trust

In trust management models the subjectivity, more specifically the disposition to trust is perceived at different degrees. This is particularly an issue in trust management

SAC'10 March 22-26, 2010, Sierre, Switzerland.

models that employ a numerical range for the quantification of trustworthiness. For instance, on a range of 0 to 9, with 0 representing blind trust and 9 representing a blind distrust, should a trustworthy node be rated as 1, 2 or 3? This evaluation mainly depends on the disposition to trust of each node (personality). The disposition to trust is the inherent propensity of an individual to trust or distrust others. An individual's disposition to trust does not vary for specific entities but is a stable characteristic of their personality that governs how they view the trustworthiness of every other entity that they encounter. McKnight et al [11] define disposition to trust as the "extent to which a person displays a tendency to be willing to depend on others across a broad spectrum of situations and persons". Rotter [12, 13] notes that an individual's "generalized attitude" towards trust is a product of life experiences, such as interactions with parents, peers, and authorities. Boone and Holmes [2] suggest that good experiences lead to a greater disposition to trust and vice versa. A study in the context of ecommerce by McCord and Ratnasingam [10] has demonstrated that there is a strong relationship between an individual's disposition to trust and the trust related decisions that they make. A thorough examination of the literature on disposition to trust is provided by Kaluscha [6]. We will now revisit a previous example which evaluate trustworthiness in the range of [0,9]. Alice and Bob are two individuals with different dispositions to trust. Alice has a high disposition to trust (trustful personality) and thus assigns a high trust value of 1 to Carol. By contrast, Bob who has a lower disposition to trust (distrustful personality), rates Carol's trustworthiness as only 5. This subjectivity shows the difference between the evaluation of Bob and Alice regardless of the fact that Carol exhibits the same behavior in her interactions with Alice and Bob.

2.2 The trust evaluation :

How does one represent the amount of trust that one individual associates with another? A common approach is to evaluate the direct trust relation with known entities by the spectrum of trust quantitatively as a numerical range. Marsh's formalism [9] represents trust as a continuous variable over an interval of [- 1,1]. Golbeck's FilmTrust [5] defines an integer range of 1 to 10. Gambetta [4], Griffiths [7], and Toivonen [17] utilize an interval of [0.1] for this purpose. An alternative approach is to divide the span of trust into strata and assign them qualitative labels. The stratification used by Abdul-Rahman and Hailes [1] is given as the set Very Trustworthy, Trustworthy, Untrustworthy, Very Untrustworthy). Josang [8], and Theodorakopoulos [16] consider the subjectivity as the uncertainty propriety and represent the trust evaluation by more then one value in multidimensional model.

The trust propagation is as the result of the transitive property of the relation trust. It plays an important role in networks with peer to peer or adhoc topology. It allows any entity to extend its confidence from its local knowledge by building a trust path from trusted peer to trusted peer. In the field of computers, the majority of the trust model approaches compute a trust average value or rescale their trust evaluation between $[0, 1]$ (if it's not the case) then they aggregate all of direct evaluation by multiplication operator. For instance, if Alice trusts Bob as 0.5 and Bob trusts Carla as 0.4, then Alice trusts bob as $0.2 = 0.5 * 0.4$. This approach is most used due to its flexibility; however it does take into account the subjectivity for evaluating the trust transitivity. In fact, a distrustful entity should evaluate a trust path more aggressively than a trustful entity.

The models that have been previously shown mainly present the following limits:

On one hand, it is so easy to assign semantic labels (eg Very Trustworthy, Trustworthy, Untrustworthy) ([1], [18]) as it is more complex to give a numerical evaluation. Thus, most solutions that address the trust evaluation make one of the following assumptions:
(i) The trust initialization is not a problem of the model, it is the responsibility of the actor of the system. However, this task is not necessarily obvious, especially when it comes to evaluating numerically a trusted third party (eg 0.1, 0.2, or 0.15).
(ii) All trust relations are initially evaluated with the value zero [8, 16]. This solution is, in some cases, interesting but does not reflect reality, because a user has given an identity, status, ownership etc.. ; These are the attributes that should be the starting point of any trust model.

On the other hands, the presented solutions consider the subjectivity as the uncertainty propriety, despite the fact (as seen before) that the disposition to trust is an important parameter which must be considered for any trust evaluation process (direct or transitive).

3. OUR APPROACH: T2D MODEL

Trust is an important element to build relationships and establish collaborations and exchanges between different entities in the environment. In peer to peer network, the trust model allows to build and expand by the transitivity, from a local of knowledge (direct trust), the circle of trust for every peer.

We define a new decentralized trust management system called T2D (Trust to Distrust) which evaluate trust relationship on a scale start from max trust threshold (0) to max distrust threshold ($max > 0$). Each actor is free to define his max value and then evaluate his trust relationship according to the defined scale.

For our trust model we represent a peer to peer network as a **Trust graph** noted as $T_g(N, E)$, a valued and directed graph such that:

- N represents the set of the peer to peer network's nodes.
- Each Trust relation between two nodes is represented by a directed edge e. The set of edges is consequently identified with the set of relations, E.
- Each edge is valued by the trust degree between the nodes represented by the source and destination nodes of this edge.

In the previous section, we noted that the evaluation models of trust display some discrepancies, especially regarding the disposition to trust. Indeed, the personality (trustful or distrustful) of network's peer has a strong influence on the evaluation of the direct or transitive trust evaluation. Thus,

the evaluation of trust relation (direct and transitive), in T2D model, is performed according to a mathematical function that allows to describe the personality of each node from trustful to distrustful.

In the following sections, we illustrate how the T2D model enable each node to define its own trust policy for evaluating **direct trust** and **transitive trust**.

4. DIRECT TRUST

The direct trust relationship is represented in the trust graph by an edge e. We noted this relation by $Trust$ as follow:

Definition 1. -Trust Relation- Let A and B two nodes ($A \in N$, $B \in N$). If A trusts B then we say that an edge e start form A to B and we note ***A Trust B***. Thus A consider B as **trusted node**.

Definition 2. -Trust Set TS- Each node builds its trust set TS which is composed of the nodes that it can evaluate directly for their trustworthiness (**trusted node**). In other words the members of the set are those nodes with whom the local node has a direct trust relationship.

Definition 3. -Trust function t^0- Each trusted node is evaluated by a trust function noted t^0 which gives to each edge e a trust degree d in a range of $[0, T^0]$. So t^0 evaluate from Trust (**0 represents the maximum trust**) to Distrust (T^0 **is the maximum distrust**), hence the name of our model $T2D$. t^0 is defined as follows:

$$\begin{array}{rccc} t^0: & N*N & \rightarrow & \mathbb{R}^+ \\ & (A,B) & \rightarrow & d \end{array} \qquad t^0(A,B) = d \mid 0 \leq d \leq T^0_A \tag{1}$$

Definition 4. -The Distrust Threshold T^0- This value is a positive number, it can be fixed freely by each node and it represents the maximum tolerated distrust value of a node.

We believe that initialization of the trustworthiness of a node as a numerical value is not a straightforward task. For instance, on a trust scale of 0 to 9, should a "very trustworthy node" be assigned the value 0, 1, 2 ... or 3? This example illustrates the dilemma faced by any node when performing this task. This difficulty occurs due to the difference between the dispositions to trust of each node. To solve this problem we introduce a novel method for the evaluation of trustworthiness of nodes in the trust set. This method comprises of the following two steps:

1. Compute a neutral evaluation using a **Trust Sort** process.
2. Generate the quantitative evaluations according to **the disposition to trust**.

4.1 TrustSort

The objective of this TrustSort is to define a method that enables one to initialize the trust set TS of each node intuitively. Thus, instead of assigning trust values to individual trusted nodes, we propose that each node performs trust evaluations in relation to other nodes.

However, if a node is required to evaluate the trustworthiness of other nodes in relation to other nodes we may have the following scenario. Let's say that Alice rates Cathy as more trustworthy than David. Based on similar experiences with Cathy and David, Bob is also very likely to rate Cathy more trustworthy than David. We thereby make the assumption that this alternate approach we are more likely is more likely to have consistent trustworthiness evaluations.

We call the notion of evaluating nodes in relation to other nodes as "Trust Sort". An administrator is in effect sorting the foreign nodes in terms of their trustworthiness. The product is a sorted list of nodes.

We note $TrustSort_n$ the node n's sorted list. Each trust set's member has an entry in the sorted list. The value contained in each box corresponds to their ranking in the list. For instance $TrustSort_n(A) = 1$ and $TrustSort_n(B) = 2$ mean that n trusts A more than B and consider A as first trusted node and B as the second one.

The sorting unit is fairly intuitive but can be limited when it is used to sort more specifically a large number of elements.

To overcome these limitations, we can define a variant of the last sorting process. In fact instead of applying the Trust Sort process between all TS's members, we can (i) define a sorted groups, for example: (Max Trust) (+)Good, Average, Poor(-) (Max Distrust); (ii) assign each trusted node to the corresponding group ; (iii) sort each group using the TrustSort process.

4.2 Disposition to trust

4.2.1 Node behavior

The next sep of our proposal consists of representing the node behavior. We can broadly classify nodes into two categories based on their disposition to trust. The first category represents nodes that generally exhibit high levels of trust in the members of their trust set. In contrast, the second category represents the nodes that are inclined towards low levels of trust in the members of their trust set.

We define a mathematical function $BV(x)$ which is called the (**B**eha**V**ior) function. This function represents a curve in the Cartesian coordinate system.

The input x is a positive value that represents the order number of a node in the sorted list. The list is numbered from 1 to n where n is the total number of nodes in the list. The node in position 1 is the most trusted node.

The output $BV(x)$ represents the corresponding quantitative trust value for the node based on the disposition to trust of the local node.

We now present the contrast between nodes that exhibit trustful and distrustful disposition to trust or behavior in terms of the BV function.

Class1 -nodes that exhibit Trustful Behavior- : This class represents the behavior of nodes which are more trusting. We define that this characteristic is represented by the BV function when it takes a hyperbola form. As illustrated in the figure 1 (box A) the projections of the x values are gathered closer to the maximum trust value (zero).

Class2 -nodes that exhibit Distrustful Behavior- : This class represents the behavior of nodes which are less trusting. We define that this characteristic is represented by the BV function when it takes a parabola form. As illustrated in the figure 1 (box B) the projections of the x values are gathered closer to the minimum trust value.

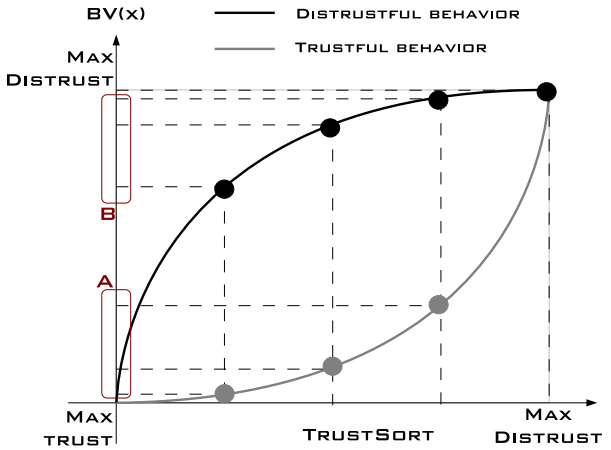

Figure 1: The Trust Behavior

4.2.2 The behavior function

We use a Bezier curve to implement the BV function due to the flexibility it enables plotting geometric curves.

Definition 5. The Bezier Curve is a parametric form to draw a smooth curve. It is achieved through some points $P_0, P_1 ... P_n$, starting at P_0 going towards $P_1 ... P_{n-1}$ and terminating at P_n.

The BV function is expressed by a quadratic Bezier curve that passes through three points where:

- The origin point $(P_0(0,0))$.
- The behavior point $(P_1(b_x, b_y))$
- The threshold point $(P_2(h_x, h_y))$ where the h_x represents the number of sorted node and the h_y represents the trust threshold.

As illustrated in the figure 2, by moving the behavior point P_1 inside the rectangle that is defined by P_0 and P_2, we are able to adjust the curvature.

Based on the Bezier curve, let us now define the "BV function".

As defined previously, the BV function describes the trust behavior of a node. It takes the order number of a node in the sorted list as the abscissa x and returns the corresponding "Quantitative trust value" $BV(x)$.

To apply the BV function with the Bezier curve, we modify the Bezier curve to obtain the ordinate as a function of abscissa, instead of taking a temporal variable 't' as input to compute both abscissa and ordinate.

The BV function curve is drawn through the three points $P_0(0,0)$, $P_1(b_x, b_y)$ and $P_2(h_x, h_y)$ using the Bezier curve.

We assume that it is sufficient to move the point P_1 through the second diagonal of the defined rectangle $b_x = \frac{-h_y}{h_x} * b_y + h_y$ to plot a large panel of behaviors.

Definition 6. -The disposition to trust level l^0- We define the variable l^0 which bounded between 0 and 1 give updated positions for P_1 through the second diagonal. For instance, the value $l^0 = 0$ indicates maximum trustful behavior $(P_1(h_x, 0))$ and $l^0 = 1$ represents maximum distrustful behavior $(P_1(0, h_y))$.

After computing the Cartesian function from the Bezier parametric format and have fixed the position of the point P_1 according to the disposition to trust level l^0, we obtain the following function:

$$BV : [0, h_x] \longrightarrow [0, h_y]$$
$$X \longrightarrow Y$$

$$BV_{l^0, h_x, h_y}(X) =$$

$$\begin{cases} \frac{(h_y - 2b_y)}{4b_x^2} X^2 + \frac{b_y}{b_x} X & si\ (h_x - 2b_x = 0) \\ (h_y - 2b_y)(\propto (X))^2 + 2b_y \propto (X), & si\ (h_x - 2b_x \neq 0) \end{cases} \quad (2)$$

$$Where \begin{cases} \propto (X) = \frac{-b_x + \sqrt{{b_x}^2 - 2b_x * X + h_x * X}}{h_x - 2b_x} \\ 0 \leq b_x \leq h_x \wedge h_x > 0 \\ b_x = (1 - l^0) \cdot h_x \wedge b_y = h_y \cdot l^0 \end{cases}$$

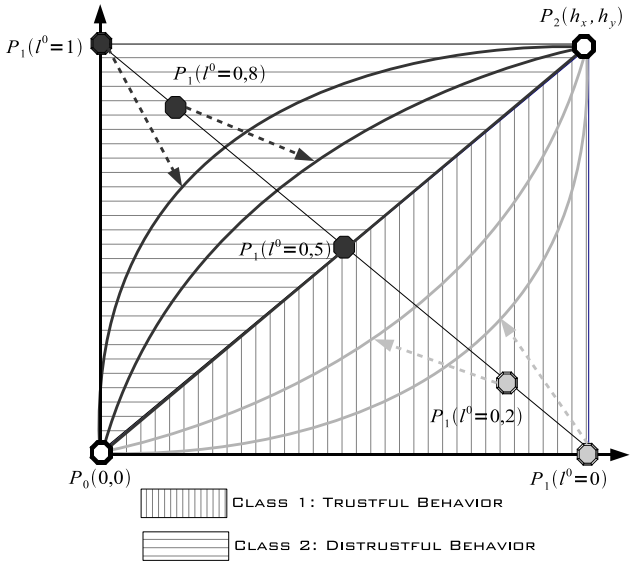

Figure 2: The Behavior curve functions

4.3 Generating quantitative trust values

Given l^0 and the threshold points (P_2), the BV function is able to assign each node in the sorted list a corresponding quantitative trust value as follows:

1. Specifying the P_1 is achieved by selecting the corresponding disposition to trust l^0 between 0 and 1.
2. The P_2 point is specified by assigning h_x and h_y the following values: h_x=(Number of trusted nodes) + 1 and $h_y = T^0$(the trust threshold).
3. Putting the trusted nodes as classified along the abscissa of the BV function.

Thus the evaluation of the trust function t^0 will be done as follows:

$$\begin{array}{rccc} t^0: & N*N & \rightarrow & \mathbb{R}^+ \\ & (A,B) & \rightarrow & d \end{array} \quad (3)$$

$$t^0(A,B) = BV_{l^0,h_x,h_y}(TrustSort_A[B]).$$

Example: Let's consider two nodes where:

- n_1: Trustful node, $l^0 = 0.1$;

- n_2: Distrustful node, $l^0 = 0.8$.

These nodes evaluate five trusted nodes (A,B,C,D,E). The threshold point P_2 has the coordinates: $h_x = 5 + 1 = 6$ *and* $h_y = T^0 = 50$.

The sorted list of both n_1 and n_2 is:

(high trust)(+) **D, C, E, A, B** (-)(low trust)

As illustrated in the figure 3, by performing the BV function the values assigned to the trusted nodes would be as follows:

	node A	node B	node C	node D	node E
node n_1	13.9	24.2	3.9	1.5	7.8
node n_2	46.1	48.6	36.1	25.8	42.2

Table 1: Example of initializing trust set's value.

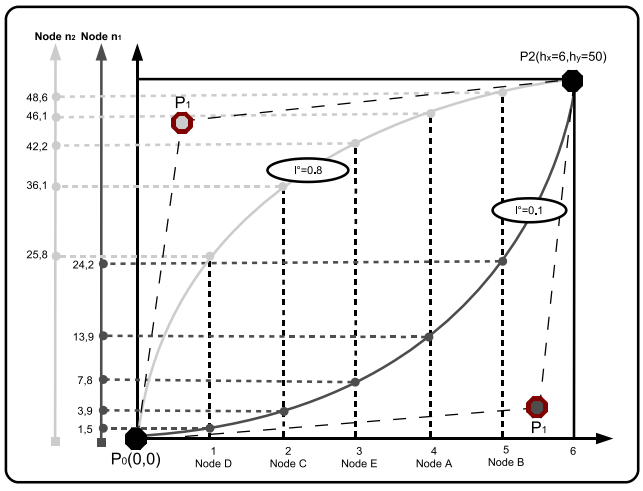

Figure 3: Node classification

5. TRANSITIVE TRUST

With the relation **Trust**, the trust graph is connected like a peer to peer network. This Trust relation can be transitive iff:

$$\forall A,B,C \in N, A\ Trust\ B \wedge B\ Trust\ C \Rightarrow A\ Trust\ C \quad (4)$$

This property is fundamental for the effectiveness of the proposition. It allows defining "trust paths" between nodes that do not know each other as follows:

let $n_0, n_1, ..., n_k \in N \mid \forall i = 0, ..., k-1\ (n_i\ Trust\ n_{i+1})$. We define a Trust path by : $Pt_k = (n_0, n_1, ..., n_k)$ where:

- n_k is called the **target node**. It represents the node which be evaluated.

- n_0 is called the **source node**. It represents the node that wants to evaluate the target one.

The evaluation of each Trust Set is decentralized. Each node evaluates its distrust threshold independently from other ones. This can lead to a divergence in the evolution of the transitive access. For example: one node can value its trusted nodes up to 20 and another can value its owns up to 500. To smooth these differences and to perform a neutral evaluation, we define the Trust propagation function P^0. P^0 uses the sum operator to add all the degree of trust path's edge relatively to the distrust threshold of the source node.

Definition 7. -Distrust propagation function P^0- Let A, B, C 3 nodes (A is the source and C is the Target). The composition of the trust degrees $t^0(A,B)$ and $t^0(B,C)$, noted $P^0(A,B,C) = t^0(A,B) \oplus t^0(B,C)$ is defined as :

$$P^0(A,B,C) = t^0(A,B) + \frac{t^0(B,C)}{T^0_B} * T^0_A$$

Generalization: trust paths

The composition of distrust degrees is generalized to n nodes through the trust path Pt_n by composing two by two the trust degrees t^0:

$$P^0(Pt_k) = \begin{cases} t^0(n_0,n_k) & si\ 0 \leq k \leq 1 \\ P^0(Pt_{k-1}) + \frac{T^0_{n_0}}{T^0_{n_{k-1}}} * t^0(n_{k-1},n_k) & si\ k > 1 \end{cases} \quad (5)$$

Definition 8. -Global distrust threshold- To validate or invalidate a trust path we define the Global distrust threshold θT^0. Each node n has to define its global distrust threshold, corresponding to the maximum tolerated degree for a transitive evaluation. This value is proportional to the distrust threshold T^0 and to the defined maximum authorized path length L_n, as follows:

$$\theta T^0 = T^0 * (1 + (1 - L_p) * (1 - l^0))$$

Consequently, if the evaluation of the trust path $(P^0(Pt_k))$ is less than the distrust threshold, i.e. $0 \leq P^0(Pt_k) \leq \theta T^0_{n_0}$, the source node n_0 trusts the trust path Pt_k and thus trust the target node n_k .

Example:

Let five nodes that build a trust path $Pt_4 = (n_0, n_1, n_2, n_3, n_4)$. To decide if the source n_0 can trust the target n_4, we compute the $P^0(Pt_4)$ progressively (see figure 4).
If $L_{n_0} = 3 \wedge T^0_{n_0} = 70$ then $\theta T^0_{n_0} = 70 * 3 = 210$.

As a consequence, the source n_0 considers the target n_4 as trust node since:

$$0 \leq (P^0(Pt_4) = 152) \leq (\theta T^0_{n_0} = 182)$$

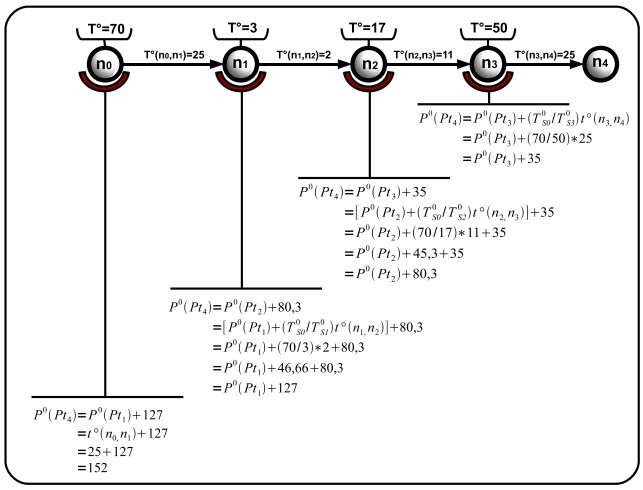

Figure 4: The Trust Propagation

6. SIMULATIONS

For the evaluation of our proposal we have implemented in Python a simulator (TrustSim). We have used this simulator to compute the trust path L_p which further enables calculation of the maximum distrust threshold $\theta T^0 = T^0 * L_p$. To accomplish this task we assume that the interconnection between nodes is higher if the network is composed of trustful nodes. In fact, each node would trust almost of all other nodes if the network is 100% trustful. The TrustSim can be freely downloaded on this link [14].

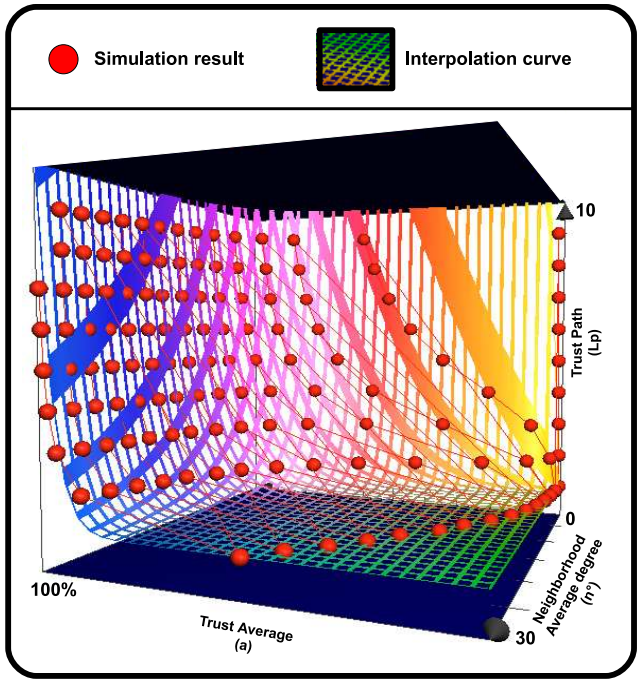

Figure 5: Path simulation: This graph is performed through network size=1000 nodes.

We run the simulation for a set of different networks characterized by their size. The set of the simulation is composed of networks sizes 200, 300, 500, 1000. Let's represent the size by the variable s and the maximum neighborhood degree of a node as n^0.

We run the simulation over networks' size by varying n^0 between 2 and 30 and L_p from 1 to 10. For each increment we compute the percentage of network's connections, i.e. how many nodes is considered as trusted from any given network node (see figure 5). We use the term "Trust average (a)" for this result.

We take the result of the simulation (i.e. the Trust average) and we use it to draw an interpolation curve. This curve is given as the function called the Trust Path TP function. This function returns the corresponding recommended path p according to network size s, trust neighborhood average n^0 and the access average a as follows:

$$\begin{array}{rccc} TP: &]0,1] * [1,N] & \longrightarrow & \mathbb{R}^+ \\ & (a, n^0) & \longrightarrow & L_p \end{array}$$

$$TP(a, n^0) = \alpha(S) * \frac{a^{n^0}}{(a*\ln(n^0+1))^{1,5}} + 1$$

$$Where\ \alpha(s) = 0,05 * s + 32,5.$$

In table 2, we assume that for $a = 75\%$ the network is very highly connected. As illustrated in table 2, by fixing a to 75%, we compute the trust recommended path according to several network characteristics, i.e. the network size and the average neighborhood degree.

$s \backslash n^0$	5	10	15	20	25	30
200	7.5	2	1.5	1	1	1
400	9	2.3	1.3	1.1	1	1
600	10.5	2.5	1.3	1.1	1	1
800	12	2.7	1.4	1.1	1	1
1000	13.5	2.9	1.4	1.1	1	1

Table 2: Maximum recommended path.

7. CONCLUSION

We have proposed a new approach for computing trust in peer to peer topology. A novel feature of this approach is that each node can evaluate its trust set by simply fixing its trust behavior than sorting the trusted nodes based on their trustworthiness.

The TrustSort method can be applied on one or more groups of trusted nodes. Each group would define a certain level of distrust e.g. group 1 can be defined to be trustworthy and group 2 can be defined in comparison to group 1 as being less trustworthy. Thus, each node is able to evaluate a large number of sites by classifying each one of them in the appropriate group.

In relation to future work, we would like to improve the TrustSort approach by a reputation mechanism. In this approach, the initial attributed value may change positively or negatively according to the node behavior. In fact, each node can inquire its trusted node one by one above all other trusted nodes. Then according to the responses received from the nodes, a classification for a node n can change if the nodes that have initially the same perception of the node n, change their evaluation.

Using this trust transitivity is particularly attractive for the distributed environment where propagation is computed using summation instead of average. Methods that use av-

erage do not take into account the length of the trust propagation path. In fact, using averages, access can be acquired over a path even if some sub-paths are not valid. However, using summation, distrust would increase when the length of the path increases. Consequently, if we cross the fixed threshold, the trust propagation path would be invalidated and access would not be allowed. Thus, it always holds true for a valid path that it does not contain any invalid sub-paths. In future work, we will also integrate the disposition to transitivity for computing the trust path by increasing the neutral evaluation according to the personality of the node.

Our T2D trust model was successfully implemented for trust requirements in pervasive [15] and grid scenarios for the GeenNet project [3].

8. REFERENCES

[1] A. Abdul-Rahman and S. Hailes. A distributed trust model. In *NSPW: New Security Paradigms Workshop*, pages 48–60, New York, USA, 1997. ACM Press.

[2] S. Boon and J. Holmes. The dynamics of interpersonal trust: Resolving uncertainty in the face of risk. In C. U. Press, editor, *Cooperation and Prosocial Behaviour*, pages 190–211, 1991.

[3] G. Da Costa, J.-P. Gelas, Y. Georgiou, K. Sharma, and J.-M. Pierson. The green-net framework: Energy efficiency in large scale distributed systems. In *HPPAC : High Performance Power Aware Computing Workshop in conjunction with IPDPS 2009, Rome, 25/05/09-29/05/09*. IEEE Computer Society, mai 2009.

[4] D. Gambetta. Can we trust trust? In *Trust: Making and Breaking Cooperative Relations*, pages 213–237. Basil Blackwell, 1988.

[5] J. Golbeck and J. Hendler. Filmtrust: Movie recommendations using trust in web-based social networks. In *CCNC: IEEE Consumer Communications and Networking Conference*, pages 282– 286, Las Vegas, NV, USA, 2006. IEEE Computer Society.

[6] S. Grabner-Kräuter, E. A. Kaluscha, and M. Fladnitzer. Perspectives of online trust and similar constructs: a conceptual clarification. In *ICEC: The international conference on Electronic commerce*, pages 235–243, New York, NY, USA, 2006. ACM.

[7] N. Griffiths. Task delegation using experience based multidimensional trust. In *International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 489–496, 2005.

[8] A. Jøsang and S. Pope. Semantic constraints for trust transitivity. In *APCCM: 2nd Asia-Pacific conference on Conceptual modelling*, pages 59–68, Newcastle, New South Wales, Australia, 2005. Australian Computer Society, Inc.

[9] S. Marsh. *Formalising Trust as a Computational Concept*. PhD thesis, University of Stirling, Scotland, 1994.

[10] M. McCord and P. Ratnasingam. The impact of trust on the technology acceptance model in business to consumer e-commerce. In *The Information Resources Management Association.*, pages 921–924, 2004.

[11] D. H. McKnight, L. L. Cummings, and N. L. Chervany. Initial trust formation in new organizational relationships. *Academy of Management Review*, 23(3):473–490, 1998.

[12] J. B. Rotter. A new scale for the measurement of interpersonal trust. *Journal of Personality*, 35(4):651–665, 1967.

[13] J. B. Rotter. Generalized expectancies for interpersonal trust. *American Psychologist*, 26(5):443–453, 1971.

[14] R.Saadi. Trustsim. `http://www-roc.inria.fr/~rsaadi`.

[15] R. Saadi, O. Hasan, J. M. Pierson, and L. Brunie. Establishing trust beliefs based on a uniform disposition to trust. In *IEEE Conference on Signal-Image Technologies and Internet-Based System*, pages 221–228. IEEE Computer Society, 2007.

[16] G. Theodorakopoulos and J. S. Baras. Trust evaluation in ad-hoc networks. In *3rd ACM workshop on Wireless security*, pages 1–10, New York, NY, USA, 2004. ACM Press.

[17] S. Toivonen, G. Lenzini, and I. Uusitalo. Context-aware trust evaluation functions for dynamic reconfigurable systems. In *Workshop on Models of Trust for the Web*, pages 11–22, 2006.

[18] P. R. Zimmermann. *The official PGP user's guide*. MIT Press, Cambridge, MA, USA, 1995.