HAL
open science

A Trust-based Π-ATMS for the Future Internet

Mondi Ravi, Yves Demazeau, Fano Ramparany

To cite this version:

Mondi Ravi, Yves Demazeau, Fano Ramparany. A Trust-based Π-ATMS for the Future Internet. Encontro Nacional de Intelligencia Artificial e Computacional, ENIAC'14, Oct 2014, Sao Carlos, Brazil. pp.7-12. hal-02189735

HAL Id: hal-02189735

https://hal.science/hal-02189735v1

Submitted on 19 Jul 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Trust-based Π-ATMS for the Future Internet

Mondi Ravi*‡, Yves Demazeau†, Fano Ramparany‡
*Univ. Grenoble Alpes, LIG, F-38000 Grenoble, France
†CNRS, LIG, F-38000 Grenoble, France
‡Orange Labs, F-38243 Meylan, France

***Abstract*—In this paper we propose a simple model of trust for the agents in a distributed environment. In the absence of sufficient information about the source and the data produced by it, we argue that the uncertainty in the data is proportional to the trustworthiness of the source. We model an agent consisting of a Π-ATMS. The agent assigns uncertainty values to the hypotheses and justifications about the world and inputs it to the Π-ATMS to obtain a list of conclusions with varying levels of uncertainty that help it to make a decision. We show how this approach can solve the problem applied to a crowd sourced garbage collection use case.**

I. Introduction

Digital society is moving towards making devices smarter. A smart device can communicate with other peers (homogeneous or heterogeneous) and can take decisions on its own. Future Internet is envisioned as interconnecting such smart devices. Connecting all uniquely identifiable objects via the Internet is "Internet of Things" (IoT) [1]. In such an interconnected world, the biggest challenge for a device is to be able to make a correct decision based upon the inputs from different sources. The sources can have contrasting levels of trust and there can be different types of uncertainty [17] associated with the measured data. In this work, we propose a solution for uncertainty that arises due to the lack of trust on the information source. We model trust as a combination of different components (cf. section IV) which we think are important for the Future Internet type of applications such as IoT. In the presence of multiple sources providing information about a hypothesis, there can be two possibilities: either the sources support the hypothesis or they contradict against it. In order to solve this, we propose a discounting function that considers the number of sources supporting a hypothesis and credulity of an agent. Maintaining both support and contradictions against a hypothesis can lead to inconsistencies. The Π-Assumption-based Truth Maintenance System (Π-ATMS) enables to maintain a consistent belief base for uncertain hypotheses. Furthermore, it provides justifications for conclusions and dependency driven backtracking in the advent of failed hypotheses. We illustrate the functioning of the entire agent model with a garbage collection use case. Thus, the objectives of this work are: (i) to explain the problems of trust and uncertainty in Future Internet applications, (ii) to propose a model of trust for IoT, (iii) to explore Trust-based Π-ATMS as a solution for resolving the problem, and (iv) to explain the solution approach with a real world garbage collection use case.

The rest of the article is organized as follows. Section II explains related works and section III introduces the garbage collection use case. The trust and uncertainty models used in the work are explained in the sections IV. Section V explains our solution approach and we finally conclude in VI.

II. Related Work

The respective domains of trust and uncertainty are quite huge and they had been topics of great interest even before the computers came into existence. A large amount of research has been done to deal with the problems of trust and uncertainty separately. Very few researchers have focused on the dual aspects of trust and uncertainty together. [22], [20], [21] are some recent work where the authors present similar problems. Venanzi et al. [22] construct a likelihood model of user trustworthiness by scaling the uncertainty of its multiple estimates. They use this model to their data fusion application. The type of uncertainty considered by them is one that arises due to imprecise measurements. This is inappropriate in a scenario like ours where the data is more subjective and we need to select a single agent out of many based upon the trustworthiness of the agent. Sensoy et al [20] present a solution for a similar problem by exploiting conflicts from trustworthy sources to revise trust in information. Their formalism consists of a subset of Description Logic and is based on Dempster Shafer's Theory (DST) of beliefs and disbeliefs. However, there have been issues relating to correctly interpreting the counter-intuitive results produced by DST [8]. There are quite a few existing data fusion technologies to deal with inputs from crowd sourcing applications [16], [22], [11], [13]. While some consider all sources being equally trustworthy [16], others take trustworthiness of information sources by scaling uncertainty measurements of the estimates [22]. Though simplicity of assuming all sources as being equally trustworthy may be worthwhile, it is not logical to assume this in a real scenario (section III) as ours. Further, it may be a requirement to classify the sources based upon the level of their trustworthiness. Rather than fusioning the information and form an aggregate, there may be a need to select the best source.

III. Use case

Smart city is a revolutionary concept where information sharing (from a crowd of people, sensors or things) along with the existing physical infrastructure of the city enables a sustainable economic development in the city leading to a well managed natural resources and high quality of life [4]. Some examples include Smart energy management, Smart public transportation system, Smart noise and pollution control etc. We consider one such smart city application "Smart Garbage Collection", to explain the need of trust and uncertainty management.

Usually an agency or a company is responsible for gathering all the garbage from the city and for disposing it to an appropriate location. The agency has a static itinerary (daily, weekly) to send its vehicles and employees to each and every corner of the city and collect the garbage. To make this process smarter the agency makes use of "intelligent containers"[1]. These containers have a specially designed wireless sensor that detects the amount of garbage in them and relay the information in real-time to one of data collector, which in turn relays it to the central information system. Thus, these containers form a wireless sensor network. The information obtained from this network allows the agency to plan a better itinerary and to improve its efficiency for garbage collection. However, the sensor network cannot be made omnipresent. Perhaps, because of the cost constraints or because garbage may be thrown randomly on the streets. In some cases, the citizen may be more careless and just let the garbage on the road, not even close to the container.

In order to deal with the problem of omnipresence of the sensors, the agency makes use of crowd-sourcing mechanism such as telephone calls from citizens, or a web-portal to report the random garbage and/or garbage near the garbage collection point. Thus, the problem of smart garbage collection can be further supported by crowd-sourcing.

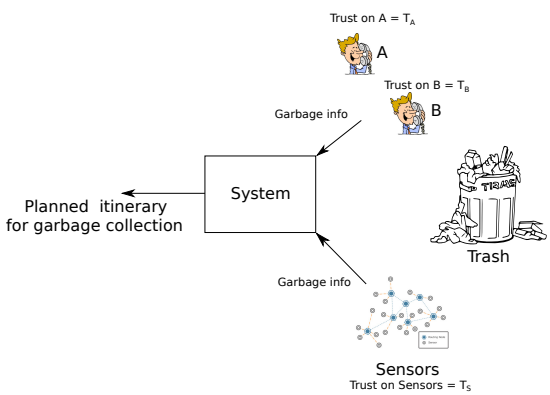

Fig. 1: A smart garbage collection system

Figure 1 shows a block diagram of such a system. Citizens A and B call the system and inform about garbage laying near their neighborhood. Additionally, there may be some information from the deployed sensors S in the streets. Based on the information received from the sources, the system then instructs the garbage men to get the garbage collected from that location.

The problem with such a setting is that there can possibly be more than one source reporting an information, there is always an uncertainty on which information source to rely on to take a decision. Another issue is how does the number of sources that report the same information affect the trust of the agent on the information. We seek to find the solutions to these problems in this work.

[1]http://www.urbiotica.com/

IV. Trust

Trust is too ambiguous to be defined uniquely. It may have as many types as the number of principals involved. Further, it may vary according to the context of consideration. It is understood and interpreted differently by people/parties. From the available literature on trust, the researchers have seen it as composed of different components, differing from domain to domain. We list them in the following section and discuss their relevance from a IoT perspective.

1) Computational aspects: Marsh's work [18] forms the foundation for computational aspects of trust. He distinguishes trust into three categories: basic, general and situational. A basic trust represents the general trust disposition of an agent (whether optimist or pessimist). A general trust is one that an agent has on other without any specification of a situation and a situational trust is specific to a situation. He also presents a numerical formalism for computing aspects such as importance, risk and cost/benefit ratio for a situation.
2) Cognitive aspects [5]: These aspects model the mental state of an agent. Competence, Willingness, Dependence, Reciprocation, Benevolence etc. come under cognitive aspects.
3) Reputation [15], [14]: It is a numeric value that an agent earns from another agent on accomplishment of a task. Reputation of an agent evolves over time, depending on the various feedback values it received from different agents.
4) Recommendation/Trust transitivity [2]: Recommendation for a data source is the communicated trust values for it by other agents.

For a distributed system like IoT, the number of agents can be rather large. Due to this constraint, we need to have a computationally simple and a realistic model for trust. Though combining all the aspects discussed above would make the model more realistic, it would be unfeasible in terms of computational complexity. Taking this into account, we propose a simple model for trust composed the reputation and the recommendation for an agent.

Let A and B be two agents. Then trust of agent A on agent B is given by:

$$ {}_AT_B = ({}_A\rho_B,\ {}_ARec_B) \tag{1} $$

where:

- ρ_B represents the reputation of the agent B for accomplishing the task. It is obtained from the evaluation of previous interactions between the agents A and B. Along the lines of Jøsang et al.[15], this reputation can be represented using Beta probability distribution. For r and s numbers of positive and negative interactions respectively, the reputation of agent B for agent A can be given by the expectation value of the β-distribution, i.e.:

$$ {}_A\rho_B \ = \ (r+1)/(r+s+2), \tag{2} $$

  The default value of trust without any interactions between any two agents ($r = 0$, $s = 0$) is 0.5.

- ${}_{A}Rec_{B}$ represents the recommendation of other agents about agent B to agent A. A recommendation is a communicated trust information, that contains reputation information [2]. If S_1, S_2, ... S_n are n recommenders for agent B to agent A, then recommender trust for A is given by weighted mean of their trust values:

$$ {}_{A}Rec_{B} = \frac{{}_{A}T_{S_1} \ * \ {}_{S_1}T_B \ + \ ... \ + \ {}_{A}T_{S_n} \ * \ {}_{S_n}T_B}{{}_{A}T_{S_1} \ + \ {}_{A}T_{S_2} \ + \ ... \ + \ {}_{A}T_{S_n}} $$

where,
${}_{A}T_{S_i}$ is the trust of agent A on agent S_i
${}_{S_i}T_B$ is the recommendation trust for agent B to A by agent S_i
and, $i \in \{1, 2, ..., n\}$

Combination of the reputation and recommendation trust is done by a weighted mean:

$$ {}_{A}T_B = \frac{\omega_1 \ * \ {}_{A}\rho_B + \omega_2 \ * \ {}_{A}Rec_B}{\omega_1 \ + \ \omega_2} \tag{3} $$

where ω_1 and ω_2 are the weights that distinguish whether the reputation or the recommendation is of more importance.

A. The Discounting Function

In a situation where all sources of information have same levels of trust, it is again a difficult proposition to make the selection of the best source. To resolve this we introduce the discounting function: $f(n, \gamma_A)$ where n denotes the number of sources with the same information and γ_A is the discounting factor of agent A. The discounting function is a characteristic of an agent. It represents how easily an agent can be influenced in the presence of multiple sources with the same information and is given by:

$$ f(n, \gamma_A) = (1 - exp(-\gamma_A n)) \tag{4} $$

As seen in figure 2, the function is exponentially growing beginning at zero and peaking up slowly or rapidly according to the value of γ_A. For instance, a value of γ_A equal to 1, 0.5 and 0.1 can correspond to a credulous, a pragmatic and a disbeliever type of agent. A credulous agent can be convinced to trust on an information rather easily (with fewer number of information sources) which is reflected in the graph. Provided all other factors remain constant and the same information coming from two different sources, the credulous will trust 86% while the pragmatist and the disbeliever will have lower trustworthiness (63% and 18% respectively for our example) for the same number of sources $n = 2$. Also, it can be noted that the more the number of sources providing the same information, the higher is the value of the discounting function. Thus, our trust model including the discounting function is given by:

$$ {}_{A}T_B{}^{dis.} = f(n, \gamma_A) \ * \ {}_{A}T_B \tag{5} $$

B. Uncertainty

According to Halpern [12], "Uncertainty is a fundamental - and unavoidable - feature of daily life". Mathematically, uncertainty is the parameter that measures the dispersion of a range of measured values. More often researchers prefer to deal with certainty which is the complement of uncertainty [19]. Since this is an unavoidable commodity, the goals of the researches have been to minimize uncertainty related to data.

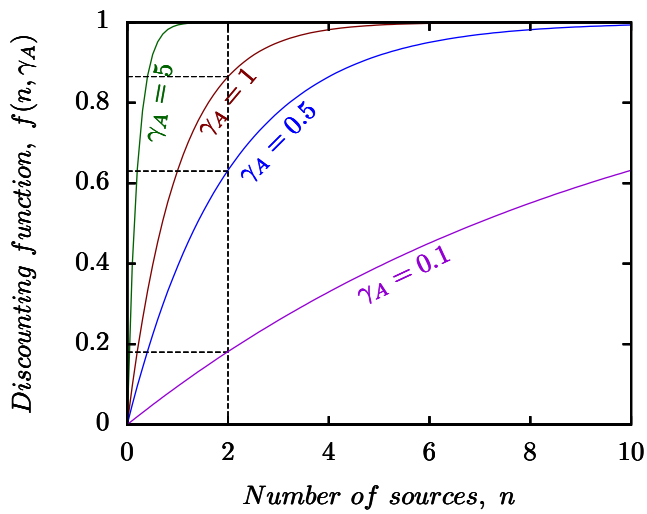

Fig. 2: The plot of the discounting function f(n,γ_A) = 1-exp(-γ_An) for γ_A equal to 5, 1, 0.5 and 0.1.

There can be many different types and sources of uncertainty. Two broad types are: (i) uncertainty due to imprecise sensor measurements (ii) uncertainty due to lack of sufficient knowledge about the domain. In this work, we limit ourselves to finding a solution for the second type. For instance, in the use case explained earlier, the system encounters uncertainty in the provided information due to lack of trust on the source. To solve this, we model uncertainty due to lack of trust on an information source as a possibility measure [9]. According to possibilistic logic, a proposition p can be represented by possibility measure $\Pi(p)$, or necessity measure $N(p)$, which are dual of each other $\forall p \ N(p) = 1 - \Pi(\neg p)$. By definition, a possibility measure $\Pi(p)$ satisfies the following axioms:

1) $\Pi(\bot) = 0, \ \Pi(\top) = 1,$
2) $\forall p, \ \forall q, \ \Pi(p \vee q) = max(\Pi(p), \Pi(q)).$

C. Relationship between trust and uncertainty

In Distributed Systems such as IoT, we have a number of autonomous/semi-autonoumous/non-autonomous agents interacting amongst themselves to achieve the goals that they are assigned. As an agent is limited in capabilities, it is usual to request informations from others. In this scenario, the agents will mostly cooperate and provide the desired information. However, because some agents are autonomous, non-cooperative behavior is also likely. This creates an uncertainty for decision making agent, whether or not the interacting agent will cooperate or defect. This uncertainty arises due to the lack of trust on the interacting agent. In other words, lack of trust on an agent causes uncertainty whether the interacting agent would act in favor or against the decision making agent. This is particularly important for systems like IoT as an agent in such a system may have several agents to obtain the information from. As pointed out as conclusion on uncertainty and trust for distributed systems by Aras et al. [3], a desirable

representation of trust for the problem must have the following three characteristics:

1) It should be able to reflect/integrate different types of uncertainties.
2) It should allow for decision making; i.e., enable ranking of alternatives, allow comparison with own standards and calculate expected risk.
3) It must be able to provide explanation/interaction context.

"A Π-ATMS" provides an answer to all the above requirements. Firstly, we model trust as a combination of reputation, recommendation and personal disposition of the agent. A possibilistic uncertainty representation is known to be effective in such scenarios where there is uncertainty due to lack of knowledge of trust on the sources, or uncertainty due to lack of precision measurements [2], [21], [23], [6]. Secondly, the Π-ATMS is able to rank different possible outcomes given sets of premises, assumptions and justifications based upon the values of trust on their respective sources. It is also possible to rank different contexts based on the trust levels of the input sources. Finally, with the support of the labels of the different nodes of the Π-ATMS, an explanation can be provided as to how the node can be derived from the given nodes, assumptions and justifications. The Π-ATMS is further explained in section V-B.

V. Solution Approach

As explained in the earlier use case, there is a need to be able to rank the different information sources based upon their trustworthiness. In order to solve these issues, we use Π-ATMS [10]. We give a brief introduction of ATMS concepts in the following subsection.

A. Assumption-based Truth Maintenance System (ATMS)

An Assumption-based Truth Maintenance System (ATMS), also called *Belief Revision System*, is a system for maintaining consistent set of beliefs in the knowledge-base. It is attached to a problem solver, which provides inputs to the ATMS in the form of nodes and justifications. A node is associated with a real world information called $datum$. One of the node types is *Premise*, that is used to represent a statement which is always true. Some nodes are designated as *assumptions*. They are considered true unless otherwise proven false. A *justification* relates how a node can be derived from any other node(s). A justification J written as $a_1 \wedge a_2 \wedge ... \wedge a_n \rightarrow c$, expresses that the node c can be justified from the nodes $a_1, a_2, ..., a_i$. An *environment* E of a node n is a set of assumptions ($E = \{a_1, a_2, ...a_i\}$) the disjunction of which derives the node n, i.e., $a_1 \wedge a_2 \wedge ...a_i \rightarrow n$. An environment becomes inconsistent if false, denoted as $\perp$, is derivable from its set of assumptions. A *Context* is a set of assumptions of a consistent environment and the nodes that can be derived from them. Each node n of the ATMS is represented by its $Label$. It is a set of environments associated with the node. A more detailed literature on ATMS is [7].

The task of ATMS is to maintain beliefs, or in other words to check whether the assumptions hold good on the arrival of nodes and justifications and then inform this to the problem solver. A set of assumptions that lead to contradiction is called a $nogood$ environment. As shown in figure 3, the ATMS receives new beliefs, justifications and query requests. The ATMS updates the contexts and responds with whether an assumption is supported (the node corresponding to the assumption is true) or not. The main advantage of ATMS over other belief revision systems is that it facilitates comparison of competing theories to explain a set of data. This matches our scenario that consists incoherent data from different sources, giving rise to competing theories. Also, ATMS functions as a cache for the problem solver.

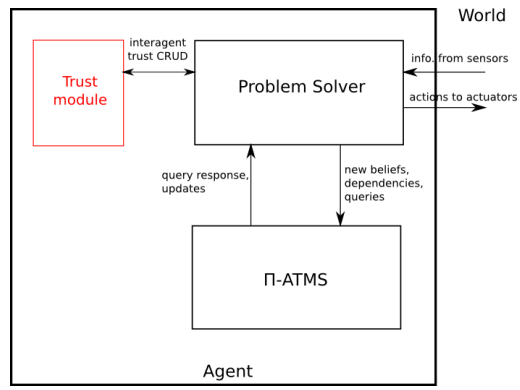

Fig. 3: An agent with a Π-ATMS and a trust module

B. Possibilistic Assumption-based Truth Maintenance System (Π-ATMS)

Π-ATMS (also called Possibilistic ATMS), as introduced by Dubois and Prade [10], is an extension of ATMS. It takes into consideration the uncertainty values that may be related to the datum, assumptions and justifications. In other words, it integrates possibilistic logic with ATMS. Π-ATMS provides a mechanism to rank different environments based on the associated uncertainty values, with the help of which the least certain ones can be safely ignored.

In Π-ATMS, each propositional formula f is associated with a weight $\alpha \in [0, 1]$ and is written as $(f\ \alpha)$. Here, α represents the lower bound of the necessity measure of the formula. A propositional formula $(f\ \ \alpha)$ that is composed of the disjunction of clauses $c_1, c_2, ...c_n$ can be equivalently written as $\{(c_1\ \alpha), (c_2\ \alpha), ..., (c_n\ \alpha)\}$. The resolution rule for resolving between two clauses $(c_1\ \alpha)$ and $(c_2\ \beta)$ is given by $(Resolvent(c_1, c_2)\ min(\alpha, \beta))$. It is also the ATMS's task to check whether the resolvent clause is:

$$\frac{(c_1\ \ \alpha) \qquad (c_2\ \ \beta)}{(Resolvent(c_1, c_2)\ \ \ min(\alpha, \beta))} \tag{6}$$

In a Π-ATMS the definitions of environment, nogood, label and context of a standard ATMS must be modified to take into account the uncertainty of assumptions, facts and justifications. Let J be a set of weighted justifications, H a set of weighted hypotheses, E a subset of H and d a datum. Then we have the following definitions:

- Environments: [E α] is an environment of d iff d can be derived from J $\cup$ E with a certainty degree α. [E α]

	Assertion	Meaning
1.	Say(X, P)	This denotes that agent X informs (says) to the system of the presence of garbage at some location P
2.	Goto(P)	This indicates the garbage collection team to goto location P to collect the garbage.
3.	IsTrue(X, P)	This asserts that the agent X is true about providing garbage information P.

TABLE I: Assertions used in the example.

is an α-environment of d iff [E α] is an environment of d and $\forall\ \alpha' > \alpha$, [E α'] is not an environment of d (i.e., α is maximal).

- Nogoods: [E α] is an α-contradictory environment, or α-nogood iff J $\cup$ E is α-inconsistent, i.e., $\perp$ can be deduced from J $\cup$ E with α maximal (α is called the inconsistency degree of J $\cup$ E).
  The α-nogood [E α] is minimal iff there is no β-nogood [E$'$ β] such that E $' \subset$ E and $\alpha \leq \beta$.
- Labels: The label of a datum d noted L(d)=$\{$[E$_i$ α_i], $i \in I$ $\}$ is the subset of the set of environments which satisfies the following properties:

C. Obtaining Necessity Weights

As we have no other information regarding the uncertainty of information from different sources apart from their trust values, we convert the numeric trust value to represent necessity. We assume that trust on an information source is equal to the certainty or degree of truth in the information provided by it. While this may not be true in all cases, it can be argued that trust values of the sources are the only way to . If p_A is a proposition put forward by an agent A with a trust T_A as observed by other agent then the necessity value can be written as:

$$N(p_A) = T_A \tag{7}$$

D. A Working Example

Continuing with the figure 1, let us consider three sources of information: A, B and C. The sources A and B are citizens of the city. Also let us assume that in the past A has provided some useful information and that agent B has never provided any information. So, the reputation and hence the trust for A will be greater than that of B. Let us assume these values being T_A=0.65 and T_B=0.5 respectively. The value 0.5 represents the default trust value on an agent without any prior interactions obtained from equation 2. Since the agent C is a sensor, we can assume that it will provide the system with more precise information. Let us assume its trust value be T_C=0.8. Next, we define the assertions used in the example in table I.

Let us suppose that a citizen or a sensor (an agent) X informs the system that there is a garbage at location P_Y. If this information is true (IsTrue(X, P) denotes that the agent X generally tells the truth about garbage information P), then the garbage collecting vehicle should be informed to go to location P_Y and collect the garbage. We represent this information in the following propositional logic statements:

$$Say(X, P_Y) \wedge IsTrue(X, P) \implies GarbageAt(P_Y) \tag{8}$$

$$GarbageAt(P_X) \implies Goto(P_X) \tag{9}$$

Since, these justifications are generic and defined by the system itself, the necessity values associated to them are equal to 1. This is because the system trusts on itself the most. In the example, we assume that the agent A informs of the garbage at location P_A and two agents B and C inform about the presence of garbage at location P_B. We assume these informations are provided in a time instance t, and avoid more complicated situations for simplicity. We use weighted Horn clause representation to illustrate resolution of the statements. The list of clauses and their respective weights are given below.

C1 (Says(A, P$_A$) 1),

C2 (Says(B, P$_B$) 1),

C3 (Says(C, P$_B$) 1),

C4 (IsTrue(A, P) 0.5),

C5 (IsTrue(B, P) 0.65),

C6 (IsTrue(C, P) 0.8),

C7 ($\neg$Says(A,P$_A$) $\vee$ $\neg$IsTrue(A, P) $\vee$ GarbageAt(P$_A$) 1),

C8 ($\neg$Says(B, P$_B$) $\vee$ $\neg$IsTrue(B,P) $\vee$ GarbageAt(P$_B$) 1),

C9 ($\neg$Says(C,P$_B$) $\vee$ $\neg$IsTrue(C,P) $\vee$ GarbageAt(P$_B$) 1),

C10 ($\neg$GarbageAt(P$_A$) $\vee$ Goto(P$_A$) 1),

C11 ($\neg$GarbageAt(P$_B$) $\vee$ Goto(P$_B$) 1)

We use the resolution formula from (6) to resolve the clauses in C1-C11. For instance, using the formula on clauses C1, C4 and C7 yields (GarbageAtP$_A$ 0.5) as the resolvent. A complete deduction is shown in the figure 4. The nodes in white rectangle are premises as indicated by empty set of environments. The white elliptical nodes are the assumption nodes, e.g. IsTrue(A, P). The blue rectangles are the derived nodes, e.g. GarbageAt(P$_A$) is derived from nodes Says(A,P$_A$) and IsTrue(A, P) using the justification clause C7.

When the problem solver (the garbage collection system in the example) queries the Π-ATMS about the nodes of interest Goto(P$_A$) and Goto(P$_B$), it receives the labels of the nodes as response. The certainty with which Goto(P$_A$) is supported is 0.5. The explanation for why Goto(P$_A$) is supported is given by the label of Goto(P$_A$) = $\{\{\text{TrueA}\}_{0.5}\}$. The 0.5-environment explains that the node Goto(P$_A$) is supported because of the assumption IsTrue(A, P) and with a certainty value 0.5. For Goto(P$_B$), however, there are two assumptions that support it as indicated by its label. They are IsTrue(B, P) and IsTrue(C, P) with certainty values 0.65 and 0.8 respectively. Thus, the system will instruct the garbage men to go to place P$_B$ that has greater support over P$_A$.

For the use of the discounting factor, let us assume that the trust on all three sources in the example are equal, i.e., T$_A$=T$_B$=T$_C$ and that their value is in the range [0.5,1), say 0.65. In such a case, we have two sources B and C, informing garbage at P$_B$ with a necessity 0.65 and the source A informing garbage at P$_A$ also with a necessity 0.65. Since, all the values are equal, the Π-ATMS responds with two places P$_A$ and P$_B$ be visited with equal certainty. In this scenario, we use the equation 5 to obtain discounted value of certainty for each of the alternatives. Assuming γ_{system}=0.5, the discounting function is equal to 0.43 and 0.63 for n=1

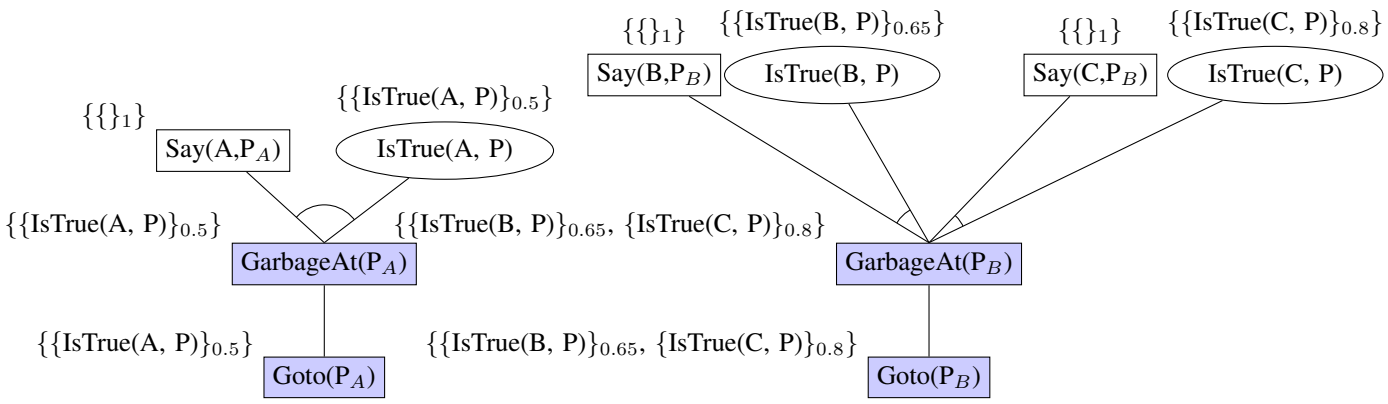

Fig. 4: Lattice showing Environments of the various nodes of the ATMS

and n=2 respectively. Thus, the discounted trusts for Goto(P_A) and Goto(P_B) become (0.43 * 0.65) $\approx$ 0.28 and (0.63 * 0.65) $\approx$ 0.41 respectively. So, the system prefers instructing the garbage men to go to P_B before P_A. Since, the necessity for Goto(P_A) is still 0.63, the system wil still instruct the garbage men to go to P_A.

VI. Conclusion

Smart devices need to consider the aspects of data uncertainty and trust on data sources, when they make their decisions. Also, since they make autonomous decisions by their own, they must be able to reason for the decisions that they make. ATMS is one such technology which helps compare different beliefs simultaneously. But, it still lacks ability to handle uncertainty about trust values associated with the input clauses. Π-ATMS can be used to consider uncertainty into the clauses of ATMS. For the simplest of cases, we can consider trust on a data source as certainty measure. Since in Π-ATMS, we are concerned with comparing the necessity measures of the clauses, their relative values can be helpful to select the most trustworthy source and make the best decision under uncertainty.

In this paper, we considered a simple use case of street garbage collection. We presented trust-based Π-ATMS based central agent for managing decisions with information provided by untrustworthy sources. In real world, the use cases are much complex. As a future work, we plan to implement a network of ATMS-based agents to achieve trust-based distributed truth maintenance.

References

[1] Internet of Things in 2020: A Roadmap for the future. *EPoSS*, 2008.

[2] A. Abdul-Rahman and S. Hailes. A distributed trust model. *Proceedings of the 1997 workshop on New security paradigms*, pages 48–60, 1998.

[3] H. Aras, C. Beckstein, S. Buchegger, P. Dittrich, F. Klan, B. Knig-ries, and O. Wolfson. Uncertainty and Trust. In *Proceedings of the Dagstuhl Invitational Seminar 08421*, pages 1–3, 2008.

[4] A. Caragliu, C. Del Bo, and P. Nijkamp. Smart cities in Europe. 2009.

[5] C. Castelfranchi and R. Falcone. Social trust: A cognitive approach. *Trust and deception in virtual societies*, 2001.

[6] C. da Costa Pereira, A. Tettamanzi, and S. Villata. Changing one's mind: Erase or rewind? In *IJCAI*, pages 164–171, 2011.

[7] J. de Kleer. An assumption-based TMS. *Artificial Intelligence*, 28(2):127–162, Mar. 1986.

[8] J. Dezert, P. Wang, and A. Tchamova. On the validity of Dempster-Shafer Theory. In *Information Fusion (FUSION), 2012 15th International Conference on*, pages 655–660, July 2012.

[9] D. Dubois. Possibility theory and statistical reasoning. *Computational statistics & data analysis*, pages 1–33, 2006.

[10] D. Dubois, J. Lang, and H. Prade. A possibilistic assumption-based truth maintenance system with uncertain justifications, and its application to belief revision. *Truth maintenance systems*, 1991.

[11] A. Etuk, T. J. Norman, and S. Murat. TIDY : A Trust-Based Approach to Information Fusion through Diversity. In *Proceedings of 16th International Conference on Information Fusion*, pages 1188–1195, 2013.

[12] J. Y. Halpern. *Reasoning about Uncertainty*. The MIT Press, Oct. 2003.

[13] A. Jøsang, J. Diaz, and M. Rifqi. Cumulative and averaging fusion of beliefs. *Inf. Fusion*, 11(2):192–200, Apr. 2010.

[14] A. Josang and J. Haller. Dirichlet reputation systems. *ARES '07 Proceedings of the The Second International Conference on Availability, Reliability and Security.*, (April), 2007.

[15] A. Josang and R. Ismail. The beta reputation system. *Proceedings of the 15th Bled Electronic Commerce Conference*, pages 1–14, 2002.

[16] E. Kamar, S. Hacker, and E. Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. *Proceedings of the Eleventh International Conference on Autonomous Agents and Multiagent Systems*, 1, 2012.

[17] K. J. Laskey, K. B. Laskey, P. C. G. Costa, M. M. Kokar, T. Martin, and T. Lukasiewicz. Uncertainty reasoning for the world wide web. mar 2008. http://www.w3.org/2005/Incubator/urw3/XGR-urw3-20080331/.

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

[19] S. Ries, S. Habib, M. Muhlhauser, and V. Varadharajan. Certainlogic: A logic for modeling trust and uncertainty. *Trust and Trustworthy Computing - 4th International Conference*, 6740(2):254–261, 2011.

[20] M. Sensoy, A. Fokoue, and J. Pan. Reasoning about uncertain information and conflict resolution through trust revision. *Proceedings of 12th International Joint Conference on AutonomousAgents and Multiagent Systems (AAMAS)*, 2, 2013.

[21] W. Teacy, M. Luck, A. Rogers, and N. Jennings. An efficient and versatile approach to trust and reputation using hierarchical Bayesian modelling. *Artificial Intelligence*, pages 1–44, 2012.

[22] M. Venanzi, A. Rogers, and N. Jennings. Trust-Based Fusion of Untrustworthy Information in Crowdsourcing Applications. In *Proceedings of 12th International Joint Conference on AutonomousAgents and Multiagent Systems (AAMAS)*, number Aamas, pages 6–10, 2013.

[23] Y. Wang and M. Singh. Formal trust model for multiagent systems. *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, 2007.