Managing Trust and Uncertainty for Distributed AI Systems

M.Ravi[1] Yves Demazeau [2] Fano Ramparany [3]

[1] Orange Labs, Meylan & Université Grenoble Alpes, LIG
[2] CNRS, LIG
[3] Orange Labs, Meylan

F-38000 Grenoble, France
mondi.ravi@orange.com

Abstract

The sharing of data and resources in Future Internet applications such as the Internet of Things (IoT) introduces a big challenge of maintaining good quality of information (QoI). Two big factors influencing QoI are (i) Trust on the information source and (ii) Uncertainty in the provided information. For a future Internet type of application, these terms are "Distributed trust" and "Distributed uncertainty". This paper discusses the problems of modeling trust and uncertainty together in a distributed AI environment, provides a survey of the existing state of the art in the domain and puts forward Distributed Π*-ATMS as a possible solution.*

Keywords

Trust ; Uncertainty ; Future Internet ; Multi-agent Systems

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 the interconnection of such smart devices. Connecting all uniquely identifiable objects via the Internet is IoT [1]. In such an interconnected world, the biggest challenge for a smart device is to be able to make a correct decision based upon the inputs from different sources. The sources can have different trust levels and there can be different types of uncertainty [14] associated with the data. Our work relates to finding a solution for management of trust and uncertainty specific to a distributed environment. As a simpler system, we consider the use case of Smart Home - where a centralized smart system in the home is connected to various sensors and actuators. Depending upon the inputs, the system can control devices in the home. We present this use case in detail in section 3.
In order to model the problems of trust and uncertainty for devices we need to understand how we as humans think of them and how we solve them. In our daily life, we make decisions all the time. Taking a bus, a cycle or a metro to work, eating at the cafeteria or trying out a new restaurant, buying clothes at a particular shop, etc. At each moment in time where we take a decision, we consider a lot of things like the source of information, possible consequences etc. The common goal for all our activities and the decisions that we take, is to become happy as a result of what we have done. Two of the most important things that we consider, knowingly or unknowingly, are :

1. Trust on the source of information
2. Uncertainty related to the truthfulness of the information

The best choice is most often the one with the minimal uncertainty (e.g. Choosing between biking to office or taking a bus/metro) or the choice which is obtained from the most trusted information source (e.g. Trusting your friend to go and try out a restaurant) or the combination of the two (e.g. Trusting the radio as a good information source and taking uncertainties related to biking or taking a bus/metro to the office). This is no different for devices of the Future Internet, which consists of multiple intelligent devices being able to take their own autonomous decisions. In fact, trust and uncertainty problems are the same for a smart device or a human-being.
Most of the research across various domains consider trust and uncertainty as independent entities. In IoT and computer networks, trust is more often seen from privacy and security point of view [9], i.e., devices in IoT are considered trustworthy if there exists a secured network to access the sensor input values. Seldom are the trust and uncertainty values associated with the devices and the values provided by them considered together as a part of data fusion or data aggregation mechanism. Since, we believe that the devices can be modeled as a Multi-agent System (MAS), the consideration of trust and uncertainty aspects are utmost important. Like reputation and recommendation systems for people in Internet of People (IoP), there must exist an equivalent trust and uncertainty management system for the objects of IoT. This system should be able to manage and evolve the trust and uncertainty values associated with the different devices of IoT. Building such a

system is the goal of our work.

1 Trust

Trust is too ambiguous to be defined uniquely for an individual. It may have as many types as the number of people or objects involved. Further, it may vary according to the context of consideration. It is understood and interpreted differently by different people at different instants of time. Human trust or Social trust depends on several factors. The experiences with the person, the expertise that he/she has, recommendations for the person etc. stand out as the visible aspects for trusting on a particular person to be able to do some task. However, there also exists some other cognitive factors of trust such as philanthropism, selfishness, reciprocativeness etc. [3] which are not explicitly implied. They are most often hidden and can only be explained by human behavior. This is why trust is very difficult to be uniquely defined and explained. For trust amongst agents in MAS, it is possible to take into consideration all these aspects for modeling it. In table 1, we present a list of desirable trust components from several important works [17, 2, 3] specifically applicable to IoP. However, considering human behavioral aspects for IoT can complicate trust computation. So for the sake of simplicity, we ignore components such as utility, risk and reciprocation for our use case.
In general, there have been several ways to model trust. We have tried to bring them together in following list.

1. Discrete trust models : These trust models represent trust of an agent quantitatively by assigning a particular value within a given numerical range for e.g. [-1, +1] as used by Marsh [17]. This value is updated according to the interactions of the agent with other agents in the system. We can further divide these types of models into :
   - Reputation-based trust models([7, 15]) : Reputation is a numeric value that an agent earns from another agent on accomplishment of a task. This is called *evidence-based model* in some literature. Reputation of an agent evolves over time, depending on the various feedback values it received from different agents.
   - Recommendation-based trust models : In [2], Abdul-Rahman describes Recommendation *as a communicated trust information that contains reputation information*. Like reputation, the recommendations evolve over time. An agent can have multiple recommendations for a task from other agents.
2. Socio-cognitive trust models : These types of trust modeling consider the sociological aspects such as : competence, willingness, persistence, motivation for computing the trust values of the agents. The seminal work of Castelfranchi and Falcone [3] is of great importance in this respect. In [6], the authors call the trust as *behavioral trust*. These models are important for modeling trust as human behavior for IoP (e.g. social networks) type of application.

Trust components	IoT	Remarks
Reputation	Y	
Recommendation	Y	
Basic trust [17]	Y	Represents the trust disposition of an agent
General trust [17]	Y	Trust between agents in general
Situational trust [17]	Y	Trust between agents in a situation
Transitivity [12]	Y	Transfer of trust among agents
Importance	Y	Importance of the current interaction
Utility	Y/N	A numerical value of cost/benefit ratio
Risk [3]	N	Involved risk in the interaction
Reciprocation [3]	N	

TABLE 1 – Trust components for IoT

3. Belief-based trust models : These trust models consider belief and disbelief as important aspects for trust calculation. The root of these models lies in the Dempster-Shafer belief theory [24]. Jøsang [11] further considers uncertainty along with the belief aspects to provide an improved belief-based trust model.
4. Security-based trust models : Some models employ cryptographic algorithms to secure the communication among various peers in a network. A public key infrastructure (PKI), Pretty Good Privacy (PGP) and X.509 are some of the examples of such models. These mechanisms do not guarantee trustworthiness in a true sense. These models say that an agent is trustworthy because the underlying communication mechanism is secure and/or a renowned third party has certified the agent to be so.

For a comprehensive model for trust more than one aspect of trust needs to be taken into account. As our research focuses to find a solution for distributed AI systems such as IoT and IoP, we see the need to model trust as composed of components listed in Table 1. The necessary components are marked as 'Y'.

2 Uncertainty

According to Halpern [8], "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 [21] prefer to deal with certainty which is the complement of uncertainty (1 - uncertainty). Since, this is an unavoidable commodity, the goal of the researches have been to minimize uncertainty related to occurrence of an event or minimize uncertainty related to data. If we are provided with

two types of data : one with high uncertainty and the other with low uncertainty, we would always go with the low uncertainty data as we feel that this data is more trustworthy. In other words, lower uncertainty reflects more trustworthy source and vice versa. There can be many types and sources of uncertainty as explained in the section 2.2. We limit ourselves to uncertainty related to data (Data uncertainty). This is sometimes referred to as *Quality of Information* [18].

2.1 Modeling uncertainty

There are various qualitative and quantitative approaches [19, 20] to model uncertainty. Since our work is related to sensors and data from sensors, we are more interested in the modeling of the quantitative analysis of uncertainty. Uncertainty alone (without the consideration of Trust aspect of the information source) can be modeled as one of the following ways :

1. Probabilistic logic : This is the most common, natural and probably the most widely used way of representing uncertainty. Each of the possible outcomes of a proposition is represented by a value in the range [0, 1]. This follows all the laws of probability theory.
2. Fuzzy logic : This approach, as introduced by Zadeh (1965), allows to classify data into different classes called *Fuzzy Sets*, depending upon their relevance or closeness to the set. Halpern [8] calls such modeling as “Possibility Measures".
3. Dempster-Shafer belief theory : Dempster-Shafer Theory basically deals with measures of two main aspects *belief* and *plausibility*. The belief is related to the certainty of the occurrence of an event and the plausibility is related to the possibility of the occurrence of the event. A simplified version of Dempster-Shafer theory is explained in the paper [24].
4. Subjective logic : Based on probabilistic logic and DST, this approach has come up as one of the important ways to model uncertainty. There can be one or more opinions about a given proposition. A binomial opinion is represented as a *Beta* distribution while multinomial opinion is represented as *Dirichlet* distribution. It explicitly takes belief and uncertainty into account. Jøsang’s work [10, 11] provides good examples for this.

2.2 Sources and types of uncertainty

Amongst the latest researches for dealing with uncertainty in IoT are [23], [22] and [18]. Wasserkrug et al. [23] present various sources of uncertainty. [22] is a working group report which brings forward the issues of uncertainty, its causes and the challenges to counter it. [18] is whitepaper which is the outcome of ongoing future internet projects under FI PPP [1]. The whitepaper [18] lists some of the practical problems posed by uncertainty in IoT and the various sources of uncertainty. As we would like to find a solution for these types of uncertainty, we list them below with further explanation.

Uncertainty due to lack of trust on data source A data source may have different levels of trust at different times and different contexts. Uncertainty on a data source which has provided accurate measurements fairly regularly is less than compared to a new data source.

Data uncertainty vs. Model uncertainty [22] Data uncertainty is caused mainly by imprecise sensor readings while model uncertainty is mainly because of error in the specification and parameter estimation. For uncertainty management, generally one of them needs to be fixed. If the model is considered true, the data must comply to the model or if the data uncertainty is fixed the model must comply to the data. If they do not comply, we would need to repair the data or the model accordingly.

Uncertainty due to inaccurate sensors This is basically the measurement error that are related to the devices. e.g. cheap sensors yielding data of very low precision.

Uncertainty due to absence of data Sometimes, due to some faulty sensors or due to lack of sensors in the area of interest, there may be no data at all. In some other cases, we may have data not exactly at the points of interest that is wanted. So, interpolation/extrapolation of data needs to be done. This results in uncertainty to reason or to infer on a particular statement.

Uncertainty due to abductive reasoning When we try to reason to our best explanations on top of available data, we are doing an abductive reasoning. For e.g., the external luminosity sensor reported a very low value. Based on this to infer that it is evening/morning time of the day can be wrong.

Uncertainty due to propagation In various data fusion components, uncertainties are taken into account to obtain a value or to take help take a decision or to pass the value to another component. This obviously propagates the uncertainty value from the first step to a later step.

Uncertainty is primarily related to data. Of all the different types of uncertainty presented in above list, the first one is of particular interest as it is related to the data source rather than data itself. We consider managing this type of uncertainty in our work.

3 Smart Home Use case

IoT has made it possible for us to connect a large numbers of electronic devices to a network and manage/control/monitor them virtually. One of the most promising areas of application of such kind of a connectivity is the concept of “Smart Home" or “Home automa-

1. http://www.fi-ppp.eu

tion". Smart home generally deals with controlling of heating, ventilation, thermostat, air-conditioning, lighting, multimedia, shading (control of curtains), security etc. by using an intelligent system that can take decisions like humans both in the presence/absence of the humans. As shown in figure 1, we can have multiple sensors in a smart home. Sensors to capture interior/exterior temperatures, luminosity, presence of the persons in the room etc. In fact, most of the time there are multiple sensors of the same or different type to detect the data more precisely. In the figure, we have two humidity and temperature sensors (HT1 and HT2) and two presence and luminosity sensors (PL1 and PL2). The goal for the controller application which controls the devices is to effectively control the devices smartly based upon approximation of available sensor data.

The problem with such a setup is more related to effective sensor data fusion based on levels of trust and uncertainty associated with the respective data sensors. For example, let us consider that temperature sensors (HT1 and HT2) provide us values with uncertainty in the range (i) $\pm\,0.5\,^{\circ}\mathrm{C}$ with probability 0.15, and (ii) in the range $\pm\,2\,^{\circ}\mathrm{C}$ with probability 0.65 (assuming that we have a probability density function for the sensors). Let us assume that the sensors have a default level of trust value associated with them when the system begins functioning. Also, let's assume a possible scenario where over a time '*t*', sensor HT1 provides us measurements within the more precise range ($\pm\,0.5\,^{\circ}\mathrm{C}$) more often than the other sensor HT2. Based on this sequence of events, we argue that trust levels associated with the sensors need to be updated. Precisely, the trust value for HT1 should increase and for HT2 decrease (by what amounts is a point of discussion and there is a existing literature which suggests that trust deteriorates faster than it ameliorates [16]). Thus, we put forward the intuition that uncertainty in the sensor data affects the trust level associated with the sensor. Conversely, we also argue that the uncertainty values to be calculated for the future sensor readings should incorporate the newly available trust values. We assume this kind of a mechanism of dynamic evolution of trust and uncertainty values to exist for all available sensors.

There are also cases where data from two different types of sensor may be dependent. For example, a high humidity value may imply that we could have a high temperature value. This introduces additional complexity to model the system. The general problem is to deal with the issue of incoherent data from different sensors.

4 Proposed Solution Approach

As explained in the earlier use case, there is a need to be able to rank the different inputs based upon the source's trustworthiness and uncertainty in the measurement. To solve these issues, we propose to use of a distributed version of Assumption-based Truth Maintenance System (ATMS) [4], explained further in the following subsection. Accomodating uncertainty into ATMS needs a special type

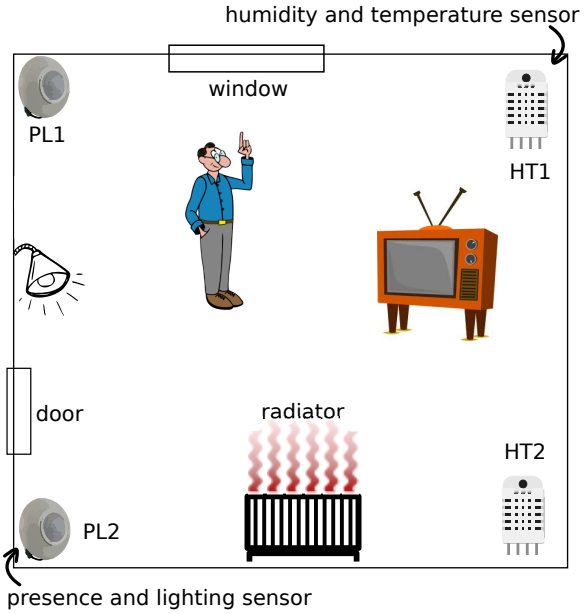

FIGURE 1 – Smart home with devices (TV, lamp, radiator) controlled by an intelligent system

of ATMS - the Π-ATMS, which is explained in the subsection 4.2.

4.1 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. 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 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 conjunction of which derives the node, i.e., $a_1 \wedge a_2 \wedge ...a_i \rightarrow n$. The task of ATMS is to maintain beliefs, or in other words check whether the assumptions hold good on the arrival of new nodes and justifications and then inform this to the problem solver. The nodes which lead to falsity are called *nogood* nodes.

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 closely resembles to our problem statement which consists incoherent data from different sources, giving rise to competing theories. Hence, we consider ATMS as a solution.

4.2 Π-ATMS

Π-ATMS (also called Possibilistic ATMS), as introduced by Dubois and Prade [5], is an extension of ATMS. It

takes into consideration the uncertainty values that may be related to the environments, clauses and assumptions. In other words, it integrates possibilistic logic with ATMS. Π-ATMS provides a mechanism to rank different environments based on the associated uncertatinty values, with the help of which the least certain ones can be safely ignored. According to possibilistic logic, uncertainty 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 necessity measure $N(p)$ satisfies the following axioms :

1. $N(\bot) = 0,\ N(\top) = 1,$
2. $\forall p,\ \forall q,\ N(p \wedge q) = min(N(p), N(q)).$

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 $(f\ \alpha)$ which 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 $(Resolvent(c_1, c_2)\ min(\alpha, \beta))$.

For e.g., let us suppose that we know two facts. First, temperature below 10°C denotes coldweather and second, we have a sensor which shows temperature below 10°C with a certainty of 65%. These facts can be represented by two clauses as (i). $(\neg TemperatureLessThan10 \vee ColdWeather\ 1)$, (ii). $(TemperatureLessThan10\ 0.65)$, then applying resolution rule we can infer $(ColdWeather\ min(1, 0.65))$ i.e., $(ColdWeather\ 0.65)$. Thus, Π-ATMS integrates uncertainty into the reasoning process. Given necessity or possibility values for the clauses, it allows a problem solver to rank different alternatives.

4.3 Distributed Π-ATMS

Distributed reason maintenance systems have been studied in the past for resolving a distributed problem [13]. We use a variant of such reason maintenance system called - Distributed Π-ATMS (DPi-ATMS). It is a Π-ATMS in a distributed setting. The main motivation for us to look into DPi-ATMS is because of the distributed knowledge in our use case. An agent can not possess the entire knowledge of the world itself. It has to communicate with a number of sensors and other information sources in order to augment its knowledge base.

A simple example of such a setting with two agents is shown in figure 2. As shown, DPi-ATMS is a component of an agent. The other components being Problem Solver (PS) and Trust Module (TM). The problem solver is the core of an agent. It communicates with other agents, sensors, actuators and the external world and constructs its belief base. The trust module stores an agent's trust on other agents. The PS updates the trust values corresponding to an agent after each interaction with it. Though the TM may be a part of the belief base itself, we have considered it separately for clarity. Assuming that trust on an agent (or data source) is proportional to the uncertainty in the data, a numerical trust value is converted to corresponding possibility or necessity measure. A distributed Π-ATMS serves two distinct purposes. Firstly, it can resolve uncertain clauses and classify them from the least possible to the most likely. Secondly, it acts as a cache for all the facts entered into it by the PS. In our use case, the agents could be the sensors, the central sensor control device, gateway devices connecting to the smart home to external services.

5 Discussion/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 decisions by their own, they must be able to reason for the decisions that they make. An ATMS is one such tool which helps compare different beliefs simultaneously. But, it still lacks ability to handle uncertainty and trust values associated with the input clauses. A Π-ATMS is a modified ATMS 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 or possibility measures of the clauses, the relative values of certainty measures of the sources can be helpful. Converting trust on data sources to absolute uncertainty measures may be domain-specific and subjective. As a future work, we look to extend our work to more complicated use cases and to a distributed environment.

Références

[1] Internet of Things in 2020 : A Roadmap for the future. Technical report, 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] C. Castelfranchi and R. Falcone. Social trust : A cognitive approach. *Trust and deception in virtual societies*, 2001.

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

[5] D. Dubois, J. Lang, and H. Prade. A possibilistic assumption-based truth maintenance system with uncertain justifications, and its application to belief revision. In *Truth Maintenance Systems (ECAI Workshop)*, pages 87–106, 1990.

[6] V. Gligor and J. M. Wing. Towards a Theory of Trust in Networks of Humans and Computers Humans and Computers. *19th International Workshop on Security Protocols*, 2011.

[7] X. Gong, T. Yu, and A. J. Lee. Bounding trust in reputation systems with incomplete information. *Proceedings of the second ACM conference on Data and Application Security and Privacy - CODASKY '12*, page 125, 2012.

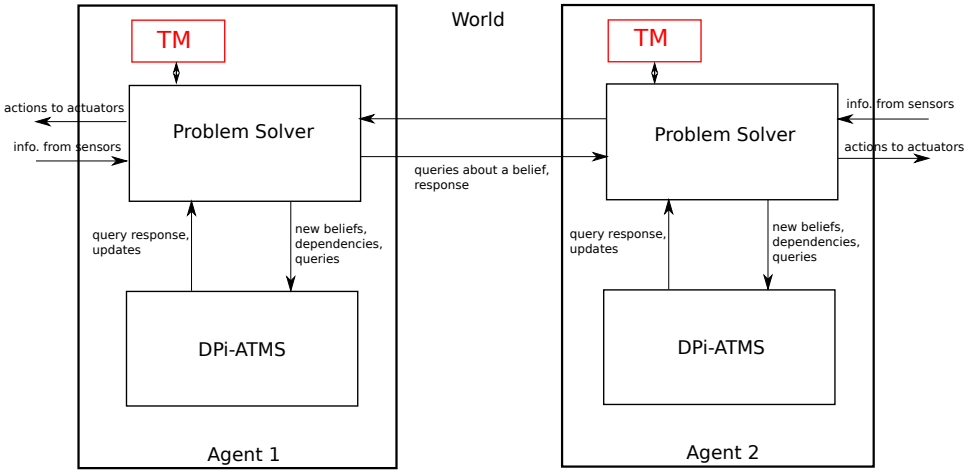

FIGURE 2 – Distributed Π-ATMSes in action

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

[9] C. Hochleitner, C. Graf, D. Unger, M. Tscheligi, and I. Center. Making Devices Trustworthy : Security and Trust Feedback in the Internet of Things. *Fourth International Workshop on Security and Privacy in Spontaneous Interaction and Mobile Phone Use (IWSSI/SPMU)*, 2012.

[10] A. Jøsang. The right type of trust for distributed systems. *Proceedings of the 1996 workshop on New security paradigms*, pages 119–131, 1996.

[11] A. Jøsang, S. Marsh, and S. Pope. Exploring different types of trust propagation. *Trust management*, (May), 2006.

[12] A. Jøsang and S. Pope. Semantic constraints for trust transitivity. In *Proceedings of the 2Nd Asia-Pacific Conference on Conceptual Modelling - Volume 43*, APCCM '05, pages 59–68, Darlinghurst, Australia, Australia, 2005. Australian Computer Society, Inc.

[13] G. K. Kraetzschmar. *Distributed Reason Maintenance for Multiagent Systems*, volume 1229 of *Lecture Notes in Computer Science*. Springer, 1997.

[14] 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 : Report on the urw3-xg incubator group. *Fourth International Workshop on Uncertainty Reasoning for the Semantic Web*, 2008.

[15] A. Lee and T. Yu. Towards a dynamic and composite model of trust. *Proceedings of the 14th ACM symposium on Access control models and technologies*, pages 217–226, 2009.

[16] P. Madhavan and D. A. Wiegmann. A new look at the dynamics of human-automation trust : is trust in humans comparable to trust in machines ? In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Santa Monica, CA, USA, 2004.

[17] S. P. Marsh, U. of Stirling. Dept. of Computing Science, and Mathematics. *Formalising trust as a computational concept*. University of Stirling, 1994.

[18] A. Metzger, A. Beulens, F. Facca, and F. Fournier. Data and Information Uncertainty. *s-cube-network.eu*, (2), 2012.

[19] S. Parsons and E. Mamdani. On reasoning in networks with qualitative uncertainty. *Proceedings of the Ninth international conference on Uncertainty in artificial intelligence*, pages 435–442, 1993.

[20] S. Parsons and A. Saffiotti. Integrating uncertainty handling formalisms in distributed artificial intelligence. In *Proceedings of the 2nd European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 304–309, 1993.

[21] S. Ries and S. Habib. Certainlogic : A logic for modeling trust and uncertainty. *Proceedings of the 4th international conference on Trust and trustworthy computing*, (2), 2011.

[22] M. Spiliopoulou, M. V. Keulen, and H. Lenz. 08421 Working Group : Imprecision, Diversity and Uncertainty : Disentangling Threads in Uncertainty Management. pages 1–3, 2009.

[23] S. Wasserkrug, A. Gal, and O. Etzion. A taxonomy and representation of sources of uncertainty in active systems. *Proceedings of the 6th international conference on Next Generation Information Technologies and Systems*, 2006.

[24] L. A. Zadeh. A simple view of the dempster-shafer theory of evidence and its implication for the rule of combination. *AI Mag.*, 7(2) :85–90, July 1986.