Confidence as a means to assess the accuracy of trust values

Rolf Kiefhaber, Gerrit Anders, Florian Siefert, Theo Ungerer, Wolfgang Reif

Angaben zur Veröffentlichung / Publication details:

Kiefhaber, Rolf, Gerrit Anders, Florian Siefert, Theo Ungerer, and Wolfgang Reif. 2012. "Confidence as a means to assess the accuracy of trust values." In *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, 25-27 June 2012, Liverpool, UK*, edited by Geyong Min, Yulei Wu, Lei (Chris) Liu, Xiaolong Jin, Stephen Jarvis, and Ahmed Y. Al-Dubai, 690–97. Piscataway, NJ: IEEE. https://doi.org/10.1109/trustcom.2012.111.

Confidence as a Means to Assess the Accuracy of Trust Values

Rolf Kiefhaber, Gerrit Anders, Florian Siefert, Theo Ungerer, and Wolfgang Reif
Institute of Computer Science
Augsburg University, Germany
E-Mail: {kiefhaber, anders, siefert, ungerer, reif}@informatik.uni-augsburg.de

***Abstract*—Open, heterogeneous multi-agent systems (MAS) have to cope with a variety of uncertainties introduced by the systems' participants and their environment. In such systems, agents can use trust values that characterize the behavior of their interaction partners as a measure of uncertainty, allowing agents to make more appropriate decisions. However, because of the systems' dynamics and the agents' limited knowledge, the accuracy of these trust values is often limited as well, introducing another form of uncertainty. In this paper, we present confidence as a general concept to indicate the degree of certainty that a trust value describes the actual observable behavior of an agent. On the basis of three open MAS, we identify different criteria the confidence depends on by revealing situations in which trust values can be inaccurate and therefore impair an agent's decision. By means of scenarios, we show that a trust-aware agent can increase its own utility when its decisions are based on the confidence in trust values.**

***Keywords*-Confidence, Trust, Reputation, Credibility, Reliability, Risk, Multi-Agent Systems, Self-Organizing Systems**

I. Introduction

Participants in open, heterogeneous multi-agent systems (MAS) have to deal with numerous uncertainties. These arise from the following properties: In such systems, agents (1) are embedded in a dynamic, potentially hostile environment, (2) only have very limited knowledge about the behavior of other agents, and (3) only have very limited control over these agents and their environment. Consequently, agents might be confronted with interaction partners that do not behave as expected or desired, and there even might be agents that try to cheat or damage the system. It is therefore crucial that agents are able to identify interaction partners they can rely on to fulfill their own or the system's goals.

Trust is a multi-faceted concept that allows agents to gauge and cope with these uncertainties. It includes credibility and reliability [1]. An agent's credibility specifies its willingness to participate in an interaction in a desirable manner, and corresponds to the original notion of trust in MAS [2]. An agent's reliability indicates its quality with regard to its availability under disturbances or partial failure.

Basically, an agent's opinion of another agent's trustworthiness, which is expressed in a trust value, originates from experiences made in repeated interactions with this agent [3]. An experience compares the expected behavior of the agent to the behavior that was observed in the course of the interaction. Trust is therefore of use when an agent's prior behavior is indicative of its future behavior. Because agents make autonomous decisions and thus can behave differently towards different interaction partners, trust is subjective. However, there may be situations in which an agent cannot appraise the trustworthiness of a potential interaction partner because it has not made any personal experiences with this agent. In such a situation, the agent can inquire about the potential interaction partner's reputation, i.e., its trustworthiness derived from the experiences of other agents in the system. Trust further depends on the context in which an interaction takes place because agents have different capabilities and goals. Importantly, an agent's trust in an interaction partner may change over time since the interaction partner may adapt its behavior according to its goals and changes in its environment.

An agent that makes decisions based on the trustworthiness of potential interaction partners (e.g., when searching for a suitable contractor at an electronic market) can increase its benefit because it is likely that an interaction partner with a high trust value behaves more beneficial than an agent with a low trust value. Therefore, the risk to interact with less trustworthy agents might be higher.

However, as agents gauge the trustworthiness of an agent on the basis of prior experiences, the trust value can be inaccurate if only a small number of experiences is available so that it does not reflect the agent's actual observable behavior. Moreover, as agents can change their behavior, the trust value might be inaccurate if it is derived from outdated experiences. Experiences that are too old are therefore possibly not used to evaluate an agent's trust value. However, it might not be possible or applicable to derive an agent's trust value from exclusively recent experiences. Furthermore, if an agent's behavior varies from one interaction to another and is thus hard to predict, it might be difficult to map this actual observable behavior onto a trust value.

These situations have in common that they yield a new form of uncertainty in the heart of the concept that is actually used to overcome the uncertainties in the system. In this paper, we introduce the *confidence* in trust values – a similar but more general concept Huynh et al. refer to as the "trust value's reliability" [4] (see Section VI for details). Confidence values allow agents to assess the accuracy of trust values, i.e., the degree of certainty that a trust value mirrors another agent's actual observable behavior.

Our evaluations show that agents can considerably augment

their own utility if they enhance their trust-based decisions by confidence values (see Section V).

The remainder of this paper is structured as follows: Section II presents three open, heterogeneous MAS, shows how trust values are used to deal with the uncertainties in these systems, and unveils situations in which the use of these trust values is limited. In Section III and Section IV, we define the concept confidence and corresponding metrics that allow to quantify the confidence in a trust value. Subsequently, we evaluate how agents can benefit from confidence on the basis of different scenarios in Section V. Section VI discusses related concepts from literature. Finally, we conclude the paper and give an outlook on future work in Section VII.

II. Trust and its Accuracy in Three Open Heterogeneous Multi-Agent Systems

Based on our definition of trust (see Section I), we show its utilization in three different examples from the domain of open, heterogeneous MAS in the following. However, we do not only explain when and how agents use trust values to deal with the uncertainties in these systems but we also expose pitfalls, i.e., situations in which agents rely on inaccurate trust values that impair their decisions. The first example is a trustworthy infrastructure for MAS, the second example is from the domain of decentralized energy management, and the third example stems from the field of distributed desktop grid systems.

The *Trust-Enabling Middleware (TEM)* [5] is based on the message-based middleware OCμ [6] that incorporates self-x properties like the ability to heal, (re-)configure, or optimize itself. In OCμ, services are run on nodes, each typically representing a single PC. To increase the system's stability and to enable load-balancing, OCμ can move services from one node to another, or restart services elsewhere in case of a node failure. To decide on the relocation of services, OCμ gathers information about services and nodes by using a piggy-back mechanism, which enriches existing messages by this information. The TEM enhances OCμ to a middleware for open, heterogeneous MAS. To allow agents to cope with uncertainties, the TEM provides trust mechanisms. All agents are able to store their experiences into the TEM and to derive trust values by using customizable metrics. The TEM itself assesses the reliability of nodes and agents by monitoring the message flow and identifying lost messages by using the Delayed Ack algorithm [5]. Its evaluation has shown that about 25 experiences are needed on average to be able to derive an accurate trust value. In addition, the TEM implements a reputation mechanism that calculates reputation values on the basis of provided trust values. By using these trust and reputation values, the TEM can improve the self-x properties of OCμ because the incorporation of trust values allows informed decisions for the relocation of agents. An agent can be moved to a more reliable node to prevent further self-healing actions. Moreover, critical agents with high credibility requirements can be moved to more credible nodes, increasing the system's trustworthiness. While these trust values can improve the system, it is essential to know how precise these trust values actually are to make accurate decisions, demonstrating the importance to introduce confidence values.

Autonomous Virtual Power Plants (AVPPs) [7] are an approach to deal with the growing complexity of and the dynamics in decentralized energy management systems. Each AVPP is a self-organizing group of power plants able to adjust its composition if needed. It is responsible for providing a specific portion of the demanded energy in the system by scheduling the energy resources under its control on the basis of predicted future output and energy demand. Furthermore, AVPPs can participate in an energy market where they can trade energy by concluding contracts. In such a system, uncertainty is introduced by non-credible and unreliable power plants, market participants, and power consumers. To cope with these uncertainties, AVPPs make extensive use of trust values. Power plants, for example, self-organize into AVPPs with respect to their trustworthiness so that each AVPP has a similar mix of trustworthy and untrustworthy power plants. Each AVPP is therefore equally capable of dealing with the uncertainties in the system. Moreover, AVPPs utilize trust values when creating power plant schedules based on predicted future output and demand to identify most likely system states, and schedule suitable power plants as well as sufficient standby energy accordingly. In this context, an agent's credibility value reflects the accuracy of its predictions. But there might be power plants or consumers whose predictions are sometimes very accurate and sometimes very inaccurate, e.g., the predictions of a solar power plant might be very accurate at night but rather inacccurate at noon. Consequently, the variance of experiences indicates the predictability of an agent's behavior. AVPPs that buy or sell energy on the market use trust values to determine trustworthy contractors, impose trade restrictions, and adjust prices to increase their competitiveness. In the context of the energy market, an agent's credibility value assesses to what extent it complies with market contracts. However, when concluding a contract on the market, the number of experiences made with a potential contractor might not be sufficient to correctly describe its behavior by a trust value. As power plants and other AVPPs pursue economic goals, they adapt their strategy from time to time. The age of experiences is therefore very important to be able to describe an agent's behavior by a trust value.

The *Trusted Computing Grid (TCG)* [8] is an open, distributed desktop grid system that allows agents (each agent represents a user) to efficiently process computationally complex tasks by splitting these tasks into work units and distributing them to other agents in the grid. Agents that receive a request for processing a work unit can decide on their own whether to accept or reject it. In general, this decision depends on the agent's strategy and the current situation. If an agent accepts the request, it commits to return a useful result in reasonable time. For example, in a soft real-time system, credible interaction partners complete the processing of a work unit and send its result before the deadline occurs. However, there may also be agents that notoriously

reject work units, those that accept work units but do not complete their computation, or others that return incorrect results. Because of these uncertainties, it is difficult to predict how long the processing of a work unit will take or whether an agent will return a useful result. Thus, an agent has to consider carefully which other agents should be responsible for processing which work unit. The participants of the TCG cope with these uncertainties by choosing their interaction partners on the basis of their credibility and reliability. As each agent preferably chooses interaction partners it trusts or those that have a good reputation, each agent forms a so-called *Implicit Trusted Community* [8] that consists of trustworthy potential interaction partners. However, trust and reputation values might be inaccurate, resulting in an improper selection of interaction partners. An agent might not have made enough experiences to be able to perceive and understand another agent's behavior. Additionally, because an agent's behavior and its strategy is a consequence of dynamic user constraints, its trustworthiness might be rated on the basis of outdated experiences. Strategic agents further can exhibit a complex varying behavior that cannot be properly described by a trust value.

In the following section, we introduce confidence as a common concept to gauge the accuracy of trust values.

III. Confidence in Trust Values

As the trust value should describe the actual observable behavior of an interaction partner based on its prior behavior, we define the *confidence in a trust value* as follows:

> *The degree of certainty that the trust value describes the actual observable behavior of an interaction partner.*

As identified in Section II, inaccurate trust values can arise from too few, outdated, or varying experiences. Accordingly, our definition of confidence depends on these criteria:

1) *Number of experiences:* The more experiences with an interaction partner were made, the more confidence in the trust value. This is the case since – if the interaction partner does not change its behavior in a serious way – more experiences mean that the trust value tends to describe the expected observable behavior more precisely. Outliers therefore do not carry that much weight, similar to the law of large numbers.
2) *Age of experiences:* Because agents may change their behavior between two interactions, the older the experiences with an interaction partner, the less confidence in the trust value. Thus, recent experiences should be considered more than older experiences as it is likely that they mirror more accurately the interaction partner's current behavior.
3) *Variance of experiences:* The more variance in the interaction partner's behavior and thus in the experiences with it, the less confidence in the trust value, because the actual observable behavior is rather likely to differ from the expected behavior described by the trust value. This is especially the case when an interaction partner often changes its behavior.

Based on this definition, we define a metric for assessing the confidence in a trust value in the next section.

IV. A Metric for Determining the Confidence in a Trust Value

In the following, we give a precise, mathematical definition of the confidence in a trust value, i.e., a metric that allows to quantify the confidence. The number of experiences (function c_n, see Section IV-C), the age of experiences (function c_a, see Section IV-D), and the variance of experiences (function c_v, see Section IV-E) are the basis for determining the confidence as a whole (function c, see Section IV-F). First, we state assumptions about the experiences and the trust metric (see Section IV-A) and introduce basics of the confidence metric (see Section IV-B).

A. Assumptions about Experiences and the Trust Metric

Let X^* be the set of all experiences made with an interaction partner. Each experience $x \in X^*$ has a time stamp t_x when it was made and therefore has an age $a_x \geq 0$. As stated in Section I, experiences that are too old are possibly not used for assessing the corresponding trust value, depending on the concrete application-specific trust metric. Thus, let $X \subseteq X^*$ be the set of experiences that are used for determining the trust value. We assume that $|X| > 0$ as otherwise no trust value exists and, therefore, it would not make sense to determine the confidence in the trust value. Further, we assume that the experiences can be mapped onto the interval $[0, 1]$. The higher this value, the more positive the experience with the interaction partner (i.e., the more beneficial the interaction). Furthermore, we assume the trust value to be the mean of the experiences X, resulting in a value $\in [0, 1]$, too. However, the trust value can also be a weighted mean value if the trust metric weights the experiences. This can be useful, e.g., if it is assumed that some experiences describe the behavior of the agent more accurate than other experiences (e.g., the newest experience could be weighted higher than the experience before, etc.).

B. Basics of the Confidence Metric

Since the confidence gauges the accuracy of the trust value, the same set X of experiences used for determining the trust value is also used for determining the confidence in the trust value. Each of the functions c_n, c_a, c_v as well as the function c has X as input and returns a value $\in [0, 1]$. Thus, the confidence value is a value $\in [0, 1]$. The functions c_n, c_a, c_v are based on a common function f whose characteristics were derived from the evaluation results of the Delayed Ack algorithm [9]. f takes some input value $z \in \mathbb{R}$ from a fixed interval $[z_0, z_1]$, $z_0 \neq z_1$:

$$f(z) = \begin{cases} 4\left(\frac{z-z_0}{z_1-z_0}\right)^3 & \text{if } z_0 \leq z \leq z_0 + \frac{1}{2}(z_1 - z_0) \\ 4\left(\frac{z-z_1}{z_1-z_0}\right)^3 + 1 & \text{if } z_0 + \frac{1}{2}(z_1 - z_0) < z \leq z_1 \end{cases}$$

f (see also Figure 1) has the property to be monotonically increasing between $f(z_0) = 0$ and $f(z_1) = 1$. Moreover, with the value in the middle of the interval $[z_0, z_1]$ as input, i.e.,

$z_0 + \frac{1}{2}(z_1 - z_0)$, f returns $\frac{1}{2}$. A combination of two cubic functions gives us low slopes for z near z_0 and z_1. Thus, the coefficient 4 maintains the condition $f(z_0 + \frac{1}{2}(z_1 - z_0)) = \frac{1}{2}$.

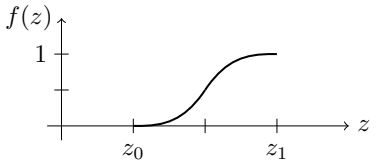

Fig. 1. Illustration of $f(z)$.

C. Number of Experiences

We expect the function $c_n(X)$ to be monotonically increasing with the number of experiences $|X| \in \mathbb{N}^+$, as, in general, more experiences with an interaction partner mean more confidence in the trust value. However, we assume that a certain number of experiences $\tau_n \in \mathbb{N}^+$ (which [4] refers to as the "rating intimacy threshold") is sufficient to derive a trust value that accurately describes the interaction partner's behavior. More than τ_n experiences do not increase $c_n(X)$. Thus, this constant τ_n defines how many experiences at least have to be made to be able to be maximum confident in a trust value. For small $|X|$ and for $|X|$ near τ_n, $c_n(X)$ should have a rather low slope, because an agent does not really have an idea how the interaction partner behaves if it made just a few experiences, and, having made already many experiences, one more experience does not improve the assessment of the interaction partner that much more.

In detail, the function $c_n(X)$ is defined as follows:

$$c_n(X) = \begin{cases} 4\left(\frac{|X|}{\tau_n}\right)^3 & \text{if } 0 \le |X| \le \frac{1}{2}\tau_n \\ 4\left(\frac{|X|-\tau_n}{\tau_n}\right)^3 + 1 & \text{if } \frac{1}{2}\tau_n < |X| \le \tau_n \\ 1 & \text{if } \tau_n < |X| \end{cases}$$

The function $c_n(X)$ (see also Figure 2) is a variant of the cubic function f and therefore is strictly increasing with the number of experiences $|X|$ for $0 \le |X| \le \tau_n$, starting with $c_n(0) = 0$. If $|X|$ exceeds τ_n, $c_n(X)$ stays 1. Moreover, we have $c_n(X) = 0.5$ with $|X| = \frac{1}{2}\tau_n$. By using a variant of the cubic function f, we make sure that the slope for small $|X|$ and for $|X|$ near τ_n is rather low.

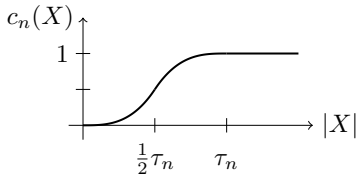

Fig. 2. Illustration of $c_n(X)$.

D. Age of Experiences

We expect the function $c_a(X)$ to be monotonically decreasing with the age of experiences. For that purpose, the age of each experience $x \in X$ has to be rated with a value $\in [0, 1]$. A low rating refers to a quite outdated experience, whereas quite recent experiences are rated high. The average rated age of all experiences contained in X gives us $c_a(X)$.

To rate the age of an experience with age a, we need a rating function $r(a)$. A simple approach would suggest to classify experiences that are below or above a certain defined age as outdated or recent experiences. However, it might be critical to distinguish only between outdated and recent experiences, especially for experiences whose age is very close to the defined age. We therefore define another classification for experiences whose age is between τ_r and τ_o ($\tau_r < \tau_o$). Thus, outdated experiences older than τ_o get a low rating, recent experiences below τ_r a high rating, and experiences with an age between τ_r and τ_o a medium rating.

In detail, the function $c_a(X)$ is defined as follows:

$$c_a(X) = \frac{\sum_{x \in X} r(a_x)}{|X|} \quad \text{with}$$

$$r(a_x) = \begin{cases} 1 & \text{if } 0 \le a_x < \tau_r \\ -4\left(\frac{a_x-\tau_r}{\tau_o-\tau_r}\right)^3 + 1 & \text{if } \tau_r \le a_x \le \tau_r + \frac{1}{2}(\tau_o - \tau_r) \\ -4\left(\frac{a_x-\tau_o}{\tau_o-\tau_r}\right)^3 & \text{if } \tau_r + \frac{1}{2}(\tau_o - \tau_r) < a_x \le \tau_o \\ 0 & \text{if } \tau_o < a_x \end{cases}$$

As stated above, the function $c_a(X)$ calculates the average rated age of experiences in X by calling the rating function $r(a)$ for each experience $x \in X$. The rating function (see also Figure 3) rates each experience x with 1 if $0 \le a_x < \tau_r$ and with 0 if $\tau_o < a_x$. For all experiences with $\tau_r \le a_x \le \tau_o$, a decreasing rating from 1 to 0 is chosen, again based on the cubic function f. Thus, those experiences near τ_r get a rather high rating, those experiences near τ_o get a rather low rating, and those in the middle get a medium rating. Consequently, we have $c_a(X) = 0$ if $\forall x \in X : r(a_x) = 0$, $c_a(X) = 1$ if $\forall x \in X : r(a_x) = 1$, and $c_a(X) = 0.5$, e.g., if half the experiences in X are rated with 0 and half the experiences in X are rated with 1.

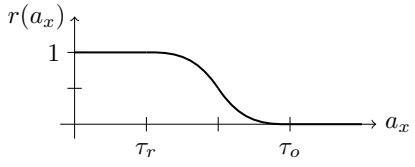

Fig. 3. Illustration of $r(a_x)$.

E. Variance of Experiences

We expect the function $c_v(X)$ to be monotonically decreasing with the variance of experiences, i.e., the higher the variance, the lower $c_v(X)$. As the experiences can be mapped

onto the interval $[0, 1]$, the variance of experiences can be determined mathematically. $c_v(X)$ should be 1 for a variance of 0, and should be 0 for the maximum possible variance. The maximum possible variance ν for each probability distribution with random values $\in [0, 1]$ is 0.25 [10], if the biased sample variance is used. As a very low or very high variance is in general hard to achieve, $c_v(X)$ should have a low slope for variances near 0 or ν.

Please note that the maximum possible variance ν can only be achieved with a sample of binary values $\in \{0, 1\}$ and a mean of 0.5, because the maximum variance of a sample depends on its mean. In all other cases, $c_v(X)$ cannot be 0. Using the maximum possible variance ν instead of a sample-specific maximum variance allows us to treat samples with the same actual variance but different means the same.

Attention is to be paid if the trust value is a weighted mean of the experiences (see Section IV-A) as the variance v_X of the sample X then should be a weighted variance, too.

In detail, the function $c_v(X)$ is defined as follows:

$$c_v(X) = \begin{cases} -4\left(\frac{v_X}{\nu}\right)^3 + 1 & \text{if } 0 \leq v_X \leq \frac{1}{2}\nu \\ -4\left(\frac{v_X - \nu}{\nu}\right)^3 & \text{if } \frac{1}{2}\nu < v_X \leq \nu \end{cases}$$

$c_v(X)$ (see also Figure 4) compares the actual variance v_X in the sample X to the maximum possible variance $\nu = 0.25$, using a monotonically decreasing variant of the cubic function f. Therefore, $c_v(X) = 1$ with $v_X = 0$, $c_v(X) = 0$ with $v_X = \nu$, and $c_v(X) = 0.5$ with $v_X = \frac{1}{2}\nu$.

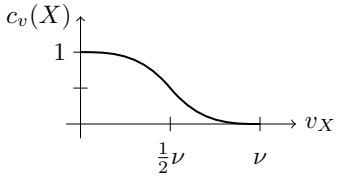

Fig. 4. Illustration of $c_v(X)$.

F. Confidence Value

For determining the confidence in a trust value, each of the three functions c_n, c_a, c_v is used, which are weighted with $\kappa_n, \kappa_a, \kappa_v \in [0, 1]$ such that $\kappa_n + \kappa_a + \kappa_v = 1$:

$$c(X) = \kappa_n \cdot c_n(X) + \kappa_a \cdot c_a(X) + \kappa_v \cdot c_v(X)$$

These weights depend on the application and whether certain aspects are particularly useful or necessary to consider in calculating the confidence. Moreover, in contrast to c_v which is defined independently of specific applications, c_n and c_a have to be parameterized for a concrete use in an application. While $\tau_n = 25$ seems to be a good value for many applications according to [9], τ_o and τ_r might depend on the frequency of interactions in the application.

V. Scenarios and Discussion

In this section, we (1) demonstrate our confidence metric by gauging the accuracy of a trust value for a specific interaction partner, and (2) show that an agent can exploit confidence values to assess the risk that trust values do not mirror the actual behavior of interaction partners, thereby increasing the agent's overall benefit in repeated interactions.

For evaluation, we modeled a system consisting of a designated agent that initiates exactly one interaction with one of 100 potential interaction partners in each time step. The outcome of each interaction could be mapped onto an interval between 0 and 1. The higher the outcome, the higher the designated agent's gain in benefit. The designated agent's goal was to maximize its overall benefit by selecting appropriate interaction partners. There were four different types of potential interaction partners. Each type defined an interval for the mean benefit b_i achieved when interacting with agent i. The b_i were generated at random, using a uniform distribution:

- Type 1: $b_i \in [0.85, 0.95]$ (30 agents)
- Type 2: $b_i \in [0.55, 0.65]$ (40 agents)
- Type 3: $b_i \in [0.4, 0.5]$ (20 agents)
- Type 4: $b_i \in [0.2, 0.3]$ (10 agents)

Within each type, the behavior of i had a variance of either $v_i = 0.1$ (40% of the agents), $v_i = 0.05$ (30% of the agents), or $v_i = 0.025$ (30% of the agents) so that the behavior of i could be modeled as a 2-tuple $\langle b_i, v_i \rangle$. The actual benefit of a single interaction with i was a random value generated by a beta distribution with mean b_i and variance v_i.

In our evaluation, the designated agent determined the trust value for an interaction partner i by calculating the average benefit of the last 30 experiences with i, thereby enabled to perceive b_i and adapt if i changed its behavior.

First, we show to what extent the designated agent can perceive the behavior of one specific interaction partner i by assessing its trustworthiness and a confidence value with respect to the number, age, and variance of experiences with this agent (c_n, c_a, and c_v). This evaluation consisted of three phases:

1) The agent i was initialized with $b_i \approx 0.6$ and $v_i = 0.1$ (Type 2). Afterwards, 30 interactions were performed.
2) i changed its behavior to Type 4 with $b_i \approx 0.2$ and $v_i = 0.1$ at time step 30, but no interactions were performed until another 30 time steps have passed.
3) At time step 60, the designated agent resumed interactions with i until the end of the evaluation was reached at time step 100.

The function c_a was parameterized with $\tau_r = 30$ and $\tau_o = 40$ (the age was measured in time steps) to analyze the effect of prohibiting any interaction with i for 30 time steps. Since the trust value was derived from the last 30 experiences, the 20 oldest experiences (with an age between 41 and 60) had a rating of 0 directly after resuming interactions with i, while the 10 newest (with an age between 31 and 40) had a rating between 0 and 1 (see Section IV-C).

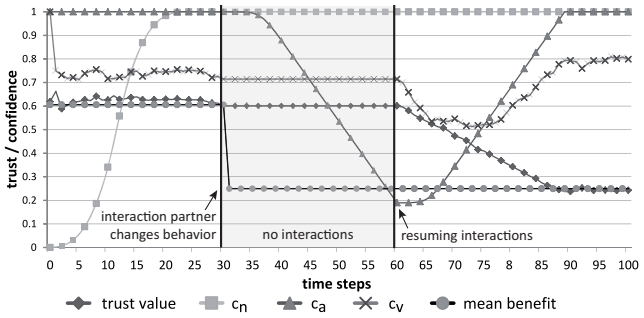

Fig. 5. Confidence metrics on a single agent

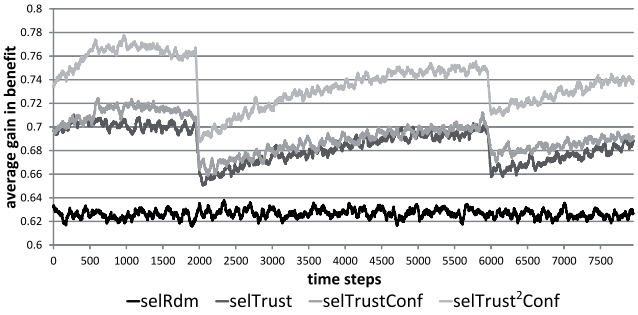

Fig. 6. Average gain in benefit for all 4 selection methods

Initially, the trust value needed some time until it represented the interaction partner's real behavior, i.e., the mean benefit of interactions with it (see Figure 5). In this part, c_n slowly raises until enough interactions are achieved. c_a is 1 since no outdated experiences exist yet. c_v stabilizes and accurately reflects i's varying behavior.

At time step 30, i changed its behavior. c_n and c_v remain constant since no new experiences were made until time step 60. However, c_a decreases as the existing experiences age.

At time step 60, the designated agent resumed interaction with i, resulting in a decreasing confidence with regard to c_v because the designated agent assessed i's behavior on the basis of experiences stemming from two different behavioral types. Within the next 30 time steps, the designated agent more and more perceived i's new behavior. Finally, the trust value mirrored the new mean benefit of an interaction. c_n, c_a, and c_v stabilize on a high value, stating that the trust value now represents i's actual observable behavior.

In the second part of our evaluation, we show that the designated agent can exploit confidence values to increase its overall benefit. This time, instead of permanently interacting with the same predefined agent, the designated agent could choose its interaction partners from the whole set of 100 agents in the system. Therefore, we compared the overall benefit the designated agent achieved when using one of four different selection methods. For the trust-based selection methods, we decided to use a roulette-wheel selection method, where each agent was associated with a given probability to be chosen. Depending on the concrete selection method, these probabilities corresponded to the trust values or to the product of trust and confidence values. This allowed to explore the behavior of different agents on the one hand, and to exploit the gathered experiences on the other hand. The selection methods were defined as follows:

- selRdm: An agent was chosen randomly using a uniform distribution, setting a baseline for comparison.
- selTrust: The designated agent calculated the trust values for all agents and selected an interaction partner by a roulette-wheel selection method.
- selTrustConf: The designated agent additionally assessed its confidence in these trust values and used the product of trust and confidence in combination with a roulette-wheel selection method.
- selTrust^{2}Conf: This method determined an interaction partner in two stages. First, a set of 10 agents was selected by using the selTrust method. Subsequently, the designated agent selected its interaction partner by using the selTrustConf selection method.

Again, we changed the behavior of some agents. This time at time steps 2000 and 6000:

- Type 1: 6/4/2 agents changed to Type 2/3/4.
- Type 2: 6/6/2 agents changed to Type 1/3/4.
- Type 3: 4/6/1 agents changed to Type 1/2/4.
- Type 4: 2/2/1 agents changed to Type 1/2/3.

After initializing the trust value for each agent by performing an initial interaction with it, exactly one interaction was performed in each time step by the designated agent. In this configuration, there might be a long time span between two interactions with the same agent. We therefore used $\tau_r = 2000$ and $\tau_o = 3000$ for c_a. The designated agent used the weights $\kappa_n = \kappa_a = \kappa_v = \frac{1}{3}$ in the function $c(X)$ to determine the confidence values, i.e., all three criteria of the confidence were considered equally. To increase the risk of choosing inappropriate interaction partners, the number of agents with a high mean benefit was lower than the number of agents with a low mean benefit (30% of the agents were of Type 1). For each selection method, we ran 200 evaluations over 8000 time steps and calculated the average gain in benefit as well as the average deviation between the expected and actual gain in benefit per time step. For each run, a new set of agents was created with respect to the different types.

Figure 6 depicts the designated agent's average gain in benefit of the last 50 interactions per time step. The results using selRdm are worse than using a trust- or confidence-based selection method. Using selTrust already significantly increases the average gain in benefit compared to selRdm, e.g., by approximately 9.64% at time step 8000. Interestingly, the results of selTrustConf are slightly higher than by selTrust, but not significantly. However, the more complex method selTrust^{2}Conf features a significant increase even over selTrustConf. At time step 8000, selTrust^{2}Conf's gain in benefit is approximately 17.84% higher than selRdm's gain in benefit. This emphasizes the importance to define a sophisticated selection method to be able to exploit the potential of confidence

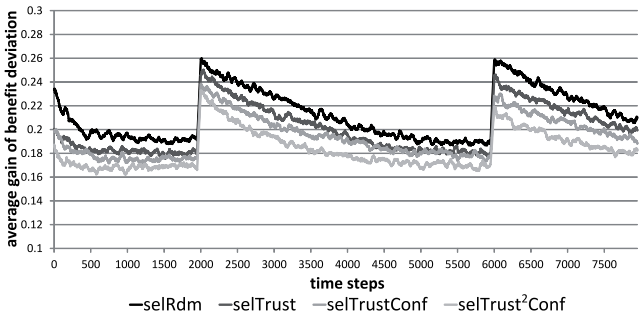

Fig. 7. Average benefit deviation for all 4 selection methods

values.

After 2000 and 6000 time steps, we changed the behavior of some agents as stated above to be able to observe how well (1) the confidence metric reflects these dynamics and (2) the designated agent adapts its decisions to the new situation. This change can clearly be seen at time step 2000 and 6000. All selection methods except selRdm experience lower gains in benefit until the designated agent adapted to the new situation. But even in the recovering phase, selTrust^{2}Conf achieves better results than the other selection methods: 50 steps after the first / second behavioral change, selTrust^{2}Conf achieves a gain in benefit that is approximately 8.24% / 13.23% higher than selRdm's gain in benefit, while selTrust only achieves a gain in benefit that is approximately 3.90% / 4.85% higher than selRdm's gain in benefit.

Figure 7 displays the average difference between the expected and the actual gain in benefit of the last 50 interactions for each interaction. The lower the value, the closer the interaction's outcome to what was expected according to the trust value. Regarding selRdm, the trust values were calculated for illustration but not used for selecting interaction partners. The results in this graph are similar to the gain in benefit of Figure 6. Using confidence, an interaction's outcome can be predicted more accurately. The changes in the agents' behavior in time steps 2000 and 6000 are also mirrored in the graphs. This illustrates that confidence is an excellent means to assess the accuracy of trust values.

VI. Related Work

Confidence is used in a variety of fields of computer science that span from statistics to branch prediction [11], neuronal networks [12], and information extraction [13].

Bloomfield et al. [14] examined confidence for the trust facet safety. More precisely, experts use specific methods such as tests or static analysis to define a *safety integrity level (SIL)* as well as the confidence that this SIL is judging the system correctly at design time. The SIL for such a system is fixed. In contrast, as we consider systems that can change their behavior, our method calculates the confidence in an agent's credibility or reliability at runtime.

In the field of multi-agent systems and trust management systems, confidence was also researched as can be seen by the following works:

FIRE [4] is a trust system that combines four types of trust and reputation: interaction trust, role-based trust, witness reputation, and certified reputation. Similar to our confidence values that describe the accuracy of trust values, FIRE introduces reliability values for interaction trust. Further, FIRE also measures the number of experiences as an important part of the final reliability value. Compared to the variance of experiences in our approach, FIRE calculates the absolute deviation of rated experiences from the trust value. We chose variance instead of the absolute deviation because we noticed in our applications (see Section II) that the variance gives valuable information to assess the uncertainty introduced by inaccurate trust values. In future work, we will show that this allows our agents to make informed decisions under uncertainty. As opposed to FIRE, we also incorporate the age of experiences into our confidence metric, which gives us another important information about the confidence in a trust value. Moreover, we based all of our confidence metrics on the same function, whose characteristic has shown to be suitable by the Delayed-Ack evaluation [9].

Hermoso et al. [15] use the same confidence metrics as proposed in FIRE. The confidence values serve as a decision tool to determine whether a decision should be made on the basis of either trust or reputation. If the confidence in the own trust value is above a given threshold, only the own trust values are used to select an appropriate interaction partner. Otherwise, reputation retrieved from other agents is taken into account. In contrast to [15], we showed that our agents can make informed decisions based on a combination of trust and confidence values, e.g., in the form of a sophisticated method for selecting interaction partners.

Ramchurn et al. [16] determine trust values on the basis of the deviation between the expected and actual result of interactions. By using a 95% confidence interval, the expected result of future interactions is predicted to select appropriate interaction partners. Further, they rate positive deviations (the actual result is better than the expected result) as a highly desirable outcome. In contrast, we do not distinguish between positive or negative deviations, e.g., an underestimated prediction of a power plant's output still negatively affects the utility frequency of the power grid.

Ba and Pavlou [17] investigated the effects of trust in ebay auctions. They have observed that a higher number of transactions is required before transactions with higher risk are considered by buyers. We acknowledge this fact by using the number of experiences in our confidence metric. They noticed that a set of negative experiences has a strong effect on trust, especially for auctions with expensive goods. Such variations can be uncovered by our variance confidence metric. This is even the case if negative experiences are rated higher than positive experiences (which thus results in a weighted mean) as our confidence metric supports the use of a weighted variance.

VII. Conclusion and Future Work

In this paper, we introduced the concept of confidence as a means to estimate the accuracy of trust values. More

precisely, we define confidence as the degree of certainty that the trust value describes the actual observable behavior of an interaction partner. Without such additional information, trust and reputation values can be potentially misleading, e.g., if an agent changed its behavior during a period with no interaction. We determined (1) the number of experiences, (2) the age of experiences, and (3) the variance of experiences as criteria that influence the confidence in a trust value. To quantify the confidence in a trust value, we specified corresponding metrics. We evaluated different scenarios in which agents select their interaction partners at random, exclusively based on the trustworthiness of their potential interaction partners, and on the basis of a combination of trust and the confidence values. The scenarios showed that agents can significantly increase their benefit in various situations when making confidence-based decisions.

Future work includes to integrate the presented confidence metrics in the applications described in Section II so that their participants can make informed decisions under uncertainty. We also want to add reputation to the methods for selecting appropriate interaction partners and use confidence as a means to decide between the information gathered from trust and reputation. Using the age of experiences as additional information and a combination of trust and reputation, we will be able to tackle the exploration vs. exploitation problem.

ACKNOWLEDGMENT

This research is partly sponsored by the research unit *OC-Trust* (FOR 1085) of the German Research Foundation.

REFERENCES

[1] J.-P. Steghöfer, R. Kiefhaber, K. Leichtenstern, Y. Bernard, L. Klejnowski, W. Reif, T. Ungerer, E. André, J. Hähner, and C. Müller-Schloer, "Trustworthy Organic Computing Systems: Challenges and Perspectives," in *Proc. of the 7th Int. Conf. on Autonomic and Trusted Computing (ATC 2010)*. Springer, October 2010.

[2] S. Ramchurn, D. Huynh, and N. Jennings, "Trust in multi-agent systems," *The Knowledge Engineering Review*, vol. 19, no. 01, pp. 1–25, 2005.

[3] G. Anders, J.-P. Steghöfer, F. Siefert, and W. Reif, "Patterns to Measure and Utilize Trust in Multi-Agent Systems," in *Proc. of the 2011 Fifth IEEE Int. Conf. on Self-Adaptive and Self-Organizing Systems Workshop (SASOW)*, October 2011, pp. 35 –40.

[4] T. Huynh, N. R. Jennings, and N. Shadbolt, "An integrated trust and reputation model for open multi-agent systems," *Journal of Autonomous Agents and Multi-Agent Systems*, vol. 13, no. 2, pp. 119–154, 2006.

[5] R. Kiefhaber, F. Siefert, G. Anders, T. Ungerer, and W. Reif, "The Trust-Enabling Middleware: Introduction and Application," Universitätsbibliothek der Universität Augsburg, Augsburg, Tech. Rep. 2011-10, 2011.

[6] M. Roth, J. Schmitt, R. Kiefhaber, F. Kluge, and T. Ungerer, "Organic Computing Middleware for Ubiquitous Environments," in *Organic Computing – A Paradigm Shift for Complex Systems*. Springer Basel, 2011, pp. 339–351.

[7] G. Anders, F. Siefert, J.-P. Steghöfer, H. Seebach, F. Nafz, and W. Reif, "Structuring and Controlling Distributed Power Sources by Autonomous Virtual Power Plants," in *Proc. of the Power & Energy Student Summit*, October 2010, pp. 40–42.

[8] Y. Bernard, L. Klejnowski, J. Hähner, and C. Müller-Schloer, "Towards Trust in Desktop Grid Systems," in *IEEE Int. Symp. on Cluster Computing and the Grid*. Los Alamitos, CA: IEEE Computer Society, 2010, pp. 637–642.

[9] R. Kiefhaber, B. Satzger, J. Schmitt, M. Roth, and T. Ungerer, "Trust Measurement Methods in Organic Computing Systems by Direct Observation," in *Proc. of the 8th Int. Conf. on Embedded and Ubiquitous Computing (EUC 2010)*. IEEE, 2010, pp. 105–111.

[10] J. Croucher, "An Upper Bound on the Value of the Standard Deviation," *Teaching Statistics*, vol. 26, no. 2, pp. 54–55, 2004.

[11] E. Jacobsen, E. Rotenberg, and J. E. Smith, "Assigning confidence to conditional branch predictions," in *Proc. of the 29th annual ACM/IEEE Int. Symp. on Microarchitecture*, ser. MICRO 29. Washington, DC, USA: IEEE Computer Society, 1996, pp. 142–152.

[12] G. Papadopoulos, P. Edwards, and A. Murray, "Confidence estimation methods for neural networks: a practical comparison," *IEEE Trans. on Neural Networks*, vol. 12, no. 6, pp. 1278 –1287, Nov. 2001.

[13] A. Culotta and A. McCallum, "Confidence estimation for information extraction," in *Proc. of HLT-NAACL 2004: Short Papers*, ser. HLT-NAACL-Short '04. Stroudsburg, PA, USA: Association for Computational Linguistics, 2004, pp. 109–112.

[14] R. E. Bloomfield, B. Littlewood, and D. Wright, "Confidence: Its role in dependability cases for risk assessment," in *Proceedings of the 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, ser. DSN '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 338–346. [Online]. Available: http://dx.doi.org/10.1109/DSN.2007.29

[15] R. Hermoso, H. Billhardt, and S. Ossowski, "Coordination, Organizations, Institutions, and Norms in Agent Systems II," P. Noriega, J. Vázquez-Salceda, G. Boella, O. Boissier, V. Dignum, N. Fornara, and E. Matson, Eds. Berlin, Heidelberg: Springer-Verlag, 2007, ch. Integrating Trust in Virtual Organisations, pp. 19–31.

[16] S. Ramchurn, C. Sierra, L. Godo, and N. R. Jennings, "Devising a trust model for multi-agent interactions using confidence and reputation," *Int. Journal of Applied Artificial Intelligence*, vol. 18, no. 9-10, pp. 833–852, 2004.

[17] Z. Lee, I. Im, and S. J. Lee, "The effect of negative buyer feedback on prices in internet auction markets," in *Proc. of the 21st Int. Conf. on Information systems*, ser. ICIS '00. Atlanta, GA, USA: Association for Information Systems, 2000, pp. 286–287.