



**HAL**  
open science

## Trust-Based Scenarios – Predicting Future Agent Behavior in Open Self-organizing Systems

Gerrit Anders, Florian Siefert, Jan-Philipp Steghöfer, Wolfgang Reif

► **To cite this version:**

Gerrit Anders, Florian Siefert, Jan-Philipp Steghöfer, Wolfgang Reif. Trust-Based Scenarios – Predicting Future Agent Behavior in Open Self-organizing Systems. 7th International Workshop on Self-Organizing Systems (IWSOS), May 2013, Palma de Mallorca, Spain. pp.90-102, 10.1007/978-3-642-54140-7\_8 . hal-01291502

**HAL Id: hal-01291502**

**<https://inria.hal.science/hal-01291502v1>**

Submitted on 21 Mar 2016

**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.



Distributed under a Creative Commons Attribution 4.0 International License

# Trust-Based Scenarios – Predicting Future Agent Behavior in Open Self-Organizing Systems

Gerrit Anders, Florian Siefert, Jan-Philipp Steghöfer, and Wolfgang Reif

Institute for Software & Systems Engineering, Augsburg University, Germany  
{anders, siefert, steghoefer, reif}@informatik.uni-augsburg.de

**Abstract.** Agents in open self-organizing systems have to cope with a variety of uncertainties. In order to increase their utility and to ensure stable operation of the overall system, they have to capture and adapt to these uncertainties at runtime. This can be achieved by formulating an expectancy of the behavior of others and the environment. Trust has been proposed as a concept for this purpose.

In this paper, we present trust-based scenarios as an enhancement of current trust models. Trust-based scenarios represent stochastic models that allow agents to take different possible developments of the environment’s or other agents’ behavior into account. We demonstrate that trust-based scenarios significantly improve the agents’ capability to predict future behavior with a distributed power management application.

**Keywords:** Scenarios, Trust, Robustness, Resilience, Uncertainty, Self-Organizing Systems, Adaptive Systems, Open Multi-Agent Systems

## 1 Uncertainties in Open Self-Organizing Systems

In open self-organizing systems, agents interact with and are embedded in a heterogeneous and dynamic environment in which potential interaction partners might not cooperate [14]. These characteristics introduce uncertainty into the decision making process of each individual agent. But even in a cooperative environment, uncertainties are relevant since agents usually have imperfect and limited knowledge about their environment or the behavior of other agents. Agents that intentionally cheat to increase their own utility as well as agents that unintentionally do not behave as expected can not only compromise the agents’ efficiency but also their ability to effectively pursue and reach their objectives. The problem is exacerbated if the regarded systems are mission-critical. In such cases, the resilience and dependability of the systems hinge on the ability to deal with uncertainties introduced by other agents and the environment.

Decentralized power management systems are an example of mission-critical systems. Because their failure can have massive consequences for people, industries, and public services, it is of utmost importance that they are stable and available at all times. However, they have to deal with a variety of uncertainties introduced by the systems’ participants and their environment. A dependable and resilient control scheme is thus necessary that, at the same time, must be

able to adapt to changing conditions quickly and autonomously. In [1], we introduced the concept of Autonomous Virtual Power Plants (AVPPs) that embodies such a control scheme in the context of a self-organizing system structure. Each AVPP controls a group of power plants. The structure changes in response to new information and changing conditions to enable each AVPP to balance its power demand and production. For this purpose, each AVPP calculates schedules that stipulate the output of controllable power plants for future points in time. The scheduled output of controllable power plants has to satisfy the “residual load” which arises as the difference between the power demand and the production of the non-controllable power plants like weather-dependent generators. To approximate the residual load, AVPPs have to rely on predictions about the future demand as well as the future output of non-controllable power plants. However, these predictions are subject to uncertainties: consumers and power plants can behave arbitrarily since they pursue individual goals and are exposed to their environment, resulting in (un)intentionally inaccurate predictions; the behavior is variable since it depends on external conditions (e.g., if snow covers the solar panels); and the behavior is time-dependent since prior good predictions are usually an indicator for future good ones. If these uncertainties are known and thus a model of the consumers’ and power plants’ underlying stochastic processes is available, an AVPP can use demand and power predictions to predict the corresponding predictor’s expected behavior. Incorporating uncertainties into the schedule creation then allows AVPPs to prevent or mitigate imbalances between energy production and demand that are caused by inaccurate predictions.

As a way to deal with uncertainties, trust models have been proposed (see Sect. 2). They allow agents to measure and quantify other agents’ behavior by means of a trust value. The value captures the experiences with that particular agent and allows to derive whether or not the interactions were beneficial in the past. If a trust value is regarded as a measure of uncertainty, it can in turn be used to make evidence-based assumptions about an agent’s future behavior. However, in general, trust values do not capture the stochastic process underlying the observed behavior. This information is lost when the trust metric derives the value. Instead, an average over the last experiences is formed that, at best, allows to predict with which probability a future interaction will again be beneficial.

In the domain of operations research, *scenarios* are a well-known concept to mirror a system’s underlying stochastic process [5]. It is expressed by different possible developments of the system. As each scenario has a certain probability of occurrence, an agent can, e.g., choose the most likely scenario and optimize for the future the scenario predicts. However, current approaches usually use predetermined scenarios and probabilities [15]. In an open self-organizing system, it is not possible to determine possible scenarios beforehand. As self-interested agents come and go and interaction patterns change, no assumptions can be made about the individual agent’s behavior. It is thus essential that scenarios and their probabilities are determined at runtime with up-to-date data.

These characteristics imply several challenges that an approach to measure uncertainties and predict future behavior has to deal with:

1. **Arbitrary behavior:** No assumptions about agent behavior can be made. The stochastic model must depict behavior that can be described by continuous or discrete random variables as well as different probability distributions.
2. **Subjective behavior:** Because agents can behave differently towards different interaction partners, each agent has to build its individual stochastic model of others.
3. **Variable behavior:** An agent’s behavior can vary and completely change over time as it can adapt to changes in its environment and, e.g., adjusts its objectives accordingly. The stochastic model must be flexible enough to reflect these changes, e.g., by a mechanism that allows for forgiveness.
4. **Time-dependent behavior:** An agent’s current behavior might depend on its behavior in the past. A stochastic model should thus be able to mirror time-dependent behavior.

In this paper, we propose to meet these challenges by dynamically calculating *trust-based scenarios* and their probabilities at runtime. They provide a stochastic model of sources of uncertainty, allowing agents to predict and deal with them by making informed and robust decisions. With regard to AVPPs, i.e., our running example, this increases the system’s resilience against mispredictions. Sect. 2 gives an overview of existing trust models, while Sect. 3 introduces trust-based scenarios. Our running example is then used to evaluate our approach in Sect. 4. In Sect. 5, we conclude the paper and give an outlook on future work.

## 2 Advantages and Limitations of Current Trust Models

*Trust* is a multi-faceted concept that allows agents to appraise their interaction partners’ behavior. Among others, trust includes the facets reliability and credibility [14]. An agent’s reliability specifies its availability, while an agent’s credibility indicates its willingness to participate in an interaction in a desirable manner and corresponds to the original notion of trust in MAS [12]. We focus on credibility in this paper. In principle, an agent  $a$ ’s trust in an interaction partner  $b$  results from *experiences* with  $b$ . Each gathered experience stems from a *contract* in which the desired result of  $a$ ’s interaction with  $b$  was stipulated. The experience is created after the interaction is completed. It contains the stipulated and the actual result of the interaction. Trust is highly subjective as an agent can show different behavior towards its interaction partners. Because of an agent’s capabilities and objectives, trust further depends on the *context*, e.g., the environmental circumstances, in which an interaction takes place. For the sake of simplicity, we deliberately abstain from trust context considerations here.

If it is valid to assume that an agent’s prior behavior is indicative for its future behavior, trust can be used to predict this future behavior and is thus a measure to cope with uncertainties as shown in Sect. 2.1 and Sect. 2.2.

### 2.1 Predicting Agent Behavior by Means of Trust Values

Because an interaction can last multiple time steps, we define a contract  $C_t^i = (c_{t+0}^i, \dots, c_{t+n}^i)$  as an  $(n+1)$ -tuple that comprises multiple stipulated results  $c_{t+j}^i$ ,

where  $j, n, t \in \mathbb{N}_0^+$ ,  $i \in \mathbb{N}^+$  is a unique identifier, and  $t + j$  identifies the time step in which the interaction partner should behave as stated in  $c_{t+j}^i$ .  $t$  and  $t + n$  thus specify the time frame in which  $C_t^i$  is valid. With respect to power, i.e., residual load, predictions in AVPPs (see Sect. 1),  $n + 1$  is the length of the prediction. For instance, if the power prediction covers a time frame of 8 hours in which the residual load is predicted in 15 minute intervals, we have  $n + 1 = 32$ . In the following, let  $[X]_j$  denote the  $j$ -th element of a tuple  $X$ . An atomic experience  $e_{t+j}^i = (c_{t+j}^i, r_{t+j}^i)$  is a 2-tuple, consisting of the stipulated result  $c_{t+j}^i$  and the actual result  $r_{t+j}^i$ . An atomic experience  $e_{t+j}^i = (7 \text{ MW}, 8 \text{ MW})$  with an AVPP's residual load states that a residual load of 7 MW was stipulated for time step  $t + j$ , but 8 MW were measured. Consequently, an experience  $E_t^i = (e_{t+0}^i, \dots, e_{t+n}^i)$  is an  $(n+1)$ -tuple of atomic experiences  $e_{t+j}^i = [E_t^i]_j$ , and  $t + j$  is the time step in which  $[E_t^i]_j$  was gained. Contracts  $C_t^i$  and experiences  $E_t^i$  comprise  $n + 1$  so-called *time slots*, e.g.,  $[E_t^i]_j$  was gained in the  $j$ -th time slot.

If an agent  $a$  evaluates the trustworthiness of an agent  $b$ , it uses a *trust metric*  $\mathcal{M} : \mathcal{E} \times \dots \times \mathcal{E} \rightarrow \mathcal{T}$  to evaluate a number of experiences with  $b$  ( $\mathcal{E}$  is the domain of experiences). The metric returns a *trust value*  $\tau \in \mathcal{T}$  and relies on a *rating function*  $\mathcal{R} : \epsilon \rightarrow \mathcal{T}$  that appraises atomic experiences ( $\epsilon$  is the domain of atomic experiences). The result of  $\mathcal{R}$  is a rating  $\pi \in \mathcal{T}$ .  $\mathcal{T}$  usually is an interval  $[0, 1]$  or  $[-1, 1]$ . Regarding  $\mathcal{T} = [0, 1]$ , a trust value  $\tau = 0$  or  $\tau = 1$  states that agent  $b$  either never or always behaves beneficially [10]. However,  $b$  behaves predictably in both cases. If the trust value is around the interval's midpoint,  $b$ 's behavior is highly unpredictable and thus induces a high level of uncertainty.

Because the residual load can be over- or underestimated, we use  $\mathcal{T} = [-1, 1]$  so that positive and negative deviations from predictions can be captured. A rating  $\pi = 0$  states that the residual load is predicted exactly, whereas  $\pi = -1$  or  $\pi = 1$  state that the residual load is greatly under- or overestimated (i.e., the actual residual load is far higher or lower than predicted).

A trust value has to be semantically sound to allow valid predictions of an agent's future behavior. This property depends on the metric  $\mathcal{M}$ .  $\mathcal{M}$  can, e.g., calculate the mean deviation between the stipulated  $c_{t+j}^i \in \mathbb{R}$  and actual result  $r_{t+j}^i \in \mathbb{R}$  of atomic experiences  $[E_{t_h}^i]_j$  contained in a list of  $m$  experiences  $E_{t_1}^{i_1}, \dots, E_{t_m}^{i_m}$  ( $k \in \mathbb{R}$  equals the maximum possible or, if not available, observed deviation from a contract and thus normalizes the result to a value in  $[-1, 1]$ ):

$$\mathcal{M}(E_{t_1}^{i_1}, \dots, E_{t_m}^{i_m}) = \frac{\sum_{h=1}^m \sum_{j=0}^n \mathcal{R}([E_{t_h}^i]_j)}{m \cdot (n + 1)}; \quad \mathcal{R}([E_t^i]_j) = \frac{c_{t+j}^i - r_{t+j}^i}{k} \quad (1)$$

Based on  $\mathcal{M}$ , a trust value  $\tau$ , and a contract  $C_t^i$ , an agent can predict the *expected behaviors*  $B_t^i = (b_{t+0}^i, \dots, b_{t+n}^i)$  of its interaction partner during  $C_t^i$ 's validity. With respect to Eq. 1, the agent's *expected behavior*  $[B_t^i]_j$  in time step  $t + j$  is defined as the difference between  $[C_t^i]_j$  and the expected deviation  $\tau \cdot k$ :

$$[B_t^i]_j = [C_t^i]_j - \tau \cdot k \quad (2)$$

For example, if  $k = 10 \text{ MW}$ ,  $\tau = 0.1$ , and the power prediction's stipulated results are  $C_t^i = (5 \text{ MW}, 6 \text{ MW})$ , the expected residual load can be predicted as

$B_t^i = (4 \text{ MW}, 5 \text{ MW})$ . If the AVPP schedules its subordinate controllable power plants on the basis of  $B_t^i$  instead of  $C_t^i$ , it is expected that the deviation between the power plants' output and the actual residual load can be decreased. However, the prediction of the residual load's future behavior with Eq. 2 can be imprecise because we disregard that an agent's behavior can be arbitrary and that it might be time-dependent (see Sect. 1). Since agents can behave arbitrarily, one and the same trust value can stem from very different experiences, e.g.,  $\tau = 0.1$  could be based on experiences in which the residual load was always 1 MW lower than stipulated or a situation in which 25% of the predictions were overestimated by 2 MW and 75% of the predictions were underestimated by  $-2$  MW. With regard to time-dependent behavior, the prediction of the residual load for a time step  $t$  could, e.g., tend to be rather precise if the prediction for the previous time step  $t - 1$  is accurate. A similar dependence could exist for inaccurate predictions.

As a trust value is usually just a mean value, it is not applicable to predict an agent's behavior in this case. Regarding an agent's underlying stochastic process, which can be represented by a probability distribution, the mean value simply does not capture enough information to approximate and describe it sufficiently.

## 2.2 Other Trust Models to Predict Agent Behavior

In the body of literature, various trust models have been presented that can be used to predict the environment's or other agents' behavior in specific situations.

In [8], confidence is proposed as a concept that indicates the degree of certainty that a trust value mirrors an agent's actual observable behavior. Among others, it depends on the standard deviation of experiences used to assess the trust value. The combined information given by the trust value (i.e., the mean behavior) and the standard deviation is still insufficient to approximate an arbitrary stochastic process underlying an agent's behavior though. The concrete probability distribution an agent's behavior follows must still be assumed.

In [6], an agent's trust value in the next time step is predicted on the basis of a Markov model. States mirror an agent's trustworthiness. The probability associated with a state change is determined at runtime, dependent on how often this transition was observed. While this model includes a basic mechanism to reflect time-dependent behavior (the trust value in the next time step depends on the current trust value), it is not applicable in situations that need predictions for more than a single time step or in situations in which too few data is available (e.g., there might be no transitions for the current trust value).

To be able to consider (in)regularities when predicting future trust values (such a behavior is shown by strategic agents), [7] presents several metrics that can determine the trust value for the next time step dependent on such properties. However, the applicability of the metrics depends on the characteristic of the agent's behavior and it is not possible to predict different possible developments of the trust value. Apart from this, these metrics basically exhibit similar drawbacks as the Markov chain method presented in [6].

Another trust model that is based on a Markov chain is introduced in [3]. In a mobile ad-hoc network, events, i.e., new experiences, trigger state changes

in the Markov chain, i.e., changes in the trust value. While the Markov chain is used to analyze the trust model, it remains unclear how agents use information like transition probabilities to make decisions or predict future behavior.

Finally, [9] presents a fuzzy regression-based approach to predict uncertainties in the field of service-oriented applications. Predictions are made on the basis of a trust value, a trust trend, and the performance consistency that reflects the trust value's stability. While this model can predict behavioral changes and trends in an e-service environment, the stochastic model an agent can derive from it can not be applied in the domain of open self-organizing systems as the challenges mentioned in Sect. 1 are not completely met.

### 3 Trust-Based Scenarios

Scenarios are a proven concept to approximate a stochastic process by taking several possible developments of a system into account (see, e.g., [11]). These developments are often represented by a *scenario tree* which can be annotated with probabilities that the system develops in a specific direction. Scenario trees then serve as input for solving optimization problems under uncertainty such as stochastic programming [13]. These techniques are applied, e.g., in the domain of power management systems (see, e.g., [4, 2, 15]). In literature, scenarios are generated, e.g., by solving an optimization problem on the basis of a system model, gathered historical data, and expert knowledge [5]. The scenario tree's structure, the number of scenarios, or probability distributions are often predefined (see, e.g., [4, 2]). Due to the computational complexity, scenarios are often determined off-line [15]. In [15], a Markov process is introduced as a simple mechanism to generate scenarios as input to solve a power plant scheduling problem. However, the authors act on the assumption that the scenarios and transition probabilities are predetermined at design time and not adapted at runtime. Further, simple scenario generation mechanisms, like the one presented in [15], often lack the ability to mirror time-dependent behavior. Scenario generation is thus often based on assumptions that do not meet the challenges outlined in Sect. 1.

In the following, we present an approach to generate trust-based scenarios (TBSs) at runtime which can in turn be used to predict the environment's or an agent's behavior on the basis of experiences gained in prior interactions.

#### 3.1 Generating Trust-Based Scenarios

Our approach to generate TBSs is similar to the creation of a histogram or a Markov chain with variable transition probabilities. We assume that an agent's behavior can be assessed by a rating function like the one given in Eq. 1 and that an agent's prior behavior is indicative of its future behavior. Basically, we categorize atomic experiences according to their rating into classes of equivalent behavior and thus approximate an agent's underlying stochastic process. The TBSs originate from experiences that have been identically classified and serve as input to generate a trust-based scenario tree (TBST). If we have contracts and

experiences of length  $n + 1$ , an agent can predict the behavior of an interaction partner or its environment in up to  $n + 1$  future time steps. The probability that an agent changes its behavior from one class to another depends on the number of experiences in which this transition was actually observed. In the following, we explain in more detail how TBSs are generated.

**Classifying Experiences:** As mentioned above, we categorize rated atomic experiences into classes. For each regarded time slot  $j \in \{0, \dots, n\}$ , these classes are consecutive and non-overlapping intervals  $\Delta\pi_j \in \Delta\mathcal{T}_j$  that only contain elements in  $\mathcal{T}$  so that each possible rating  $\pi$  can be assigned to an interval, i.e.,  $\forall \pi \in \mathcal{T} : \exists \Delta\pi_j \in \Delta\mathcal{T}_j : \pi \in \Delta\pi_j$ . Suitable sizes of these intervals depend on the application. To mirror time-dependent behavior, there is a set  $\Delta\mathcal{T}_j$  for each time slot  $j$  so that an atomic experience  $[E_t^i]_j$  is classified into an interval  $\Delta\pi_j \in \Delta\mathcal{T}_j$ .

By assigning each rated atomic experience  $\mathcal{R}([E_t^i]_j)$  contained in an experience  $E_t^i \in \mathcal{E}$  to such an interval  $\Delta\pi_j \in \Delta\mathcal{T}_j$ ,  $E_t^i$  is converted into a *classified experience*  $\Xi^i \in \mathcal{E}^*$  ( $\mathcal{E}^*$  denotes the set of classified experiences). A classified experience  $\Xi^i = (\Delta\pi_0, \dots, \Delta\pi_n)$  is thus an  $(n+1)$ -tuple, i.e., a sequence, of intervals  $\Delta\pi_j \in \Delta\mathcal{T}_j$  that contain the corresponding rated atomic experience  $\mathcal{R}([E_t^i]_j)$  of  $E_t^i$  (see Eq. 3). In Fig. 1, steps 1 and 2 illustrate how experiences are classified.

$$\Xi^i \Leftrightarrow \forall j \in \{0, \dots, n\} : \mathcal{R}([E_t^i]_j) \in [\Xi^i]_j \wedge [\Xi^i]_j \in \Delta\mathcal{T}_j \quad (3)$$

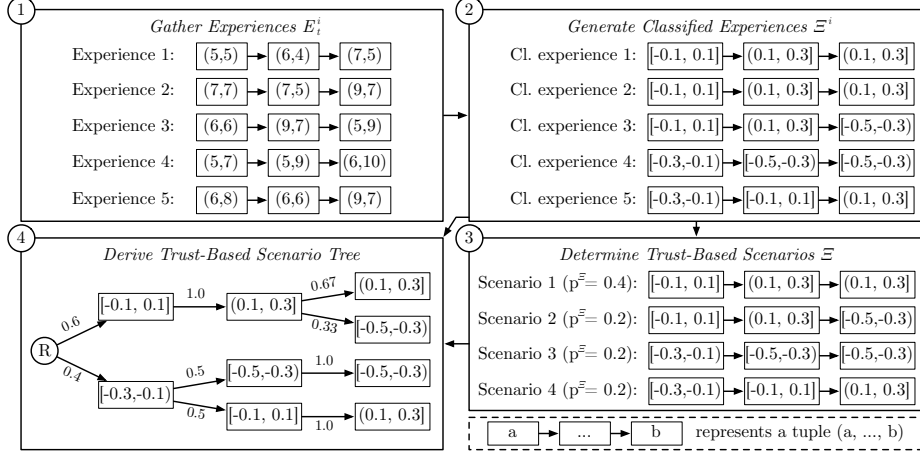
**Determining Trust-Based Scenarios:** A *trust-based scenario* (TBS)  $\Xi = (\Delta\pi_0, \dots, \Delta\pi_n)$  arises from classified experiences  $\Xi^i = (\Delta\pi_0, \dots, \Delta\pi_n)$ . More precisely, each group of experiences that were classified into the same sequence of intervals forms a TBS (the set  $\mathcal{S} = \Delta\mathcal{T}_0 \times \dots \times \Delta\mathcal{T}_n$  contains all possible TBSs). Note that in contrast to  $\Xi^i$ ,  $\Xi$  has no identifier  $i$ . For each  $\Xi$ , a probability of occurrence  $p^\Xi$  can be calculated. It is the ratio of the number of classified experiences  $\Xi^i$  that feature the same sequence of intervals as defined by  $\Xi$  ( $\forall j \in \{0, \dots, n\} : [\Xi^i]_j = [\Xi]_j$ ) to the number of all classified experiences  $|\mathcal{E}^*|$ :

$$p^\Xi := \frac{1}{|\mathcal{E}^*|} \cdot |\{\Xi^i \mid \forall \Xi^i \in \mathcal{E}^* : \forall j \in \{0, \dots, n\} : [\Xi^i]_j = [\Xi]_j\}| \quad (4)$$

Hence, each generated TBS  $\Xi$  has a probability  $p^\Xi > 0$ . These TBSs form the set  $\mathcal{S}^+ \subseteq \mathcal{S}$ . If  $\Xi$  is used to predict an agent's behavior, it thus represents a corridor of expected behavior. In Fig. 1, steps 2 and 3 show how TBSs are created.

**Deriving a Trust-Based Scenario Tree:** The TBSs can now be used to derive an agent's stochastic model, the *trust-based scenario tree*  $TBST = (\Theta, \Phi)$ , where  $\Theta$  is the set of nodes and  $\Phi$  the set of edges. The TBST is based on all generated TBSs (i.e., all  $\Xi \in \mathcal{S}^+$ ). A node  $\theta^\rho \in \Theta$  is defined by an interval  $\theta$  and a sequence of intervals  $\rho$ . Let  $[\Xi]^j = ([\Xi]_0, \dots, [\Xi]_{j-1})$  specify the first  $j$  elements  $[\Xi]_j$  of the interval sequence defined by a TBS  $\Xi$ , i.e., a prefix of length  $j$  (if  $j < 1$ ,  $[\Xi]^j$  is the empty sequence " $\emptyset$ "). Then, for each  $j \in \{0, \dots, n\}$  and each TBS  $\Xi \in \mathcal{S}^+$ , there is a node  $\theta^\rho \in \Theta$  with  $\theta = [\Xi]_j$  and  $\rho = [\Xi]^j$ . A node  $\theta^\rho \in \Theta$  thus states that an agent showed a behavior classified as  $\theta$  after having shown behaviors classified as  $\rho$  in the given order. Additionally,  $\Theta$  contains the tree's root  $R$ . If there are two nodes  $\theta_1^{\rho_1}$  and  $\theta_2^{\rho_2}$  with  $\rho_1 = [\Xi]^{j-1}$  and  $\rho_2 = [\Xi]^j$ , there





**Fig. 1.** Trust-based scenarios approximate an agent's underlying stochastic process by classifying gathered experiences, determining trust-based scenarios (TBSs), and thereupon deriving a trust-based scenario tree (TBST). The TBST is then used to predict future behavior on the basis of a stipulated contract. In this example,  $\mathcal{T} = [-1, 1]$  and  $\forall j \in \{0, \dots, n\} : \Delta\mathcal{T}_j = \{[-1.0, -0.9], [-0.9, -0.7], \dots, [-0.3, -0.1], [-0.1, 0.1], (0.1, 0.3], \dots, (0.7, 0.9], (0.9, 1.0]\}$ . Atomic experiences are rated using the rating function given in Eq. 1 with  $k = 10$ . The length of experiences is  $n + 1 = 3$ . Tuples  $(a, b)$  represent atomic experiences  $[E_t^i]_j$ . In step 4, the values at the TBST's edges indicate the conditional probabilities to change from one behavior to another.

is an edge  $(\theta_1^{\rho_1}, \theta_2^{\rho_2}) \in \Phi$ . Note that  $\rho_2$  is equal to the concatenation  $\rho_1 \oplus \theta_1$ . Further, for each TBS  $\Xi$ ,  $\Phi$  contains an edge  $(R, \theta^0)$  from  $R$  to the interval  $\theta = [\Xi]_0$  at time slot 0. More precisely, nodes and edges are defined as follows:

$$\Theta := R \cup \{ \theta^\rho \mid \exists \Xi \in \mathcal{S}^+ : \exists j \in \{0, \dots, n\} : \theta = [\Xi]_j \wedge \rho = [\Xi]^j \} \quad (5)$$

$$\Phi := \{ (R, \theta^\rho) \mid \exists \Xi \in \mathcal{S}^+ : \theta = [\Xi]_0 \wedge \rho = \emptyset \} \cup \{ (\theta_1^{\rho_1}, \theta_2^{\rho_2}) \mid \exists \theta_1^{\rho_1}, \theta_2^{\rho_2} \in \Theta : \exists \Xi \in \mathcal{S}^+ : \exists j \in \{1, \dots, n\} : \rho_1 = [\Xi]^{j-1} \wedge \rho_2 = [\Xi]^j \} \quad (6)$$

As all leaves are at depth  $n + 1$ , each path from the TBST's root to a leaf represents a TBS, and each TBS with non-zero probability is represented by such a path.

In a TBST, each edge  $(\theta_1^{\rho_1}, \theta_2^{\rho_2})$  has a probability  $p(\theta_1^{\rho_1}, \theta_2^{\rho_2})$  that indicates the conditional probability that the agent will show a behavior classified as  $\theta_2$  if it showed behaviors specified by  $\rho_2$  in the previous time steps. This probability is the ratio of the number of classified experiences that start with the sequence  $\rho_1 \oplus \theta_1 \oplus \theta_2$  ( $\rho_2 = \rho_1 \oplus \theta_1$ ) to the number that start with  $\rho_2$ . Similarly, the probability of edges  $(R, \theta^0)$  is the probability that the agent shows a behavior as described in  $\theta$  in the first time step of the contract's validity. The probability  $p(\theta_1^{\rho_1}, \theta_2^{\rho_2})$  with  $\theta_1^{\rho_1} \neq R$  is calculated as follows ( $\mathcal{J} = \{1, \dots, n\}$ ):

$$p(\theta_1^{\rho_1}, \theta_2^{\rho_2}) := \frac{|\{ \Xi^i \mid \exists \Xi^i \in \mathcal{E}^* : \exists j \in \mathcal{J} : \rho_2 = [\Xi^i]^j \wedge \theta_2 = [\Xi^i]_j \}|}{|\{ \Xi^i \mid \exists \Xi^i \in \mathcal{E}^* : \exists j \in \mathcal{J} : \rho_2 = [\Xi^i]^j \}|} \quad (7)$$

As each TBS with non-zero probability is represented by a path in the TBST, the TBS's probability is the product over all conditional probabilities assigned to the edges of its path. In Fig. 1, steps 3 and 4 depict how a TBST is generated.

### 3.2 Predicting Agent Behavior by Means of Trust-Based Scenarios

A TBST can now be used to predict an agent's behavior. Given a TBST, a rating function  $\mathcal{R}$  (see Sect. 2.1), and a contract  $C_t^i$ , several scenarios for an agent's expected behavior can be determined. For this purpose, each interval  $\theta = [\bar{\theta}, \underline{\theta}]$  that is represented by a node  $\theta^\rho$  in the TBST with lower and upper endpoints  $\bar{\theta}$  and  $\underline{\theta}$  is transformed into a *corridor of expected behavior*  $[\bar{b}_j, b_j]$ , similar to Eq. 2. If  $\mathcal{R}$  and  $k$  are defined as in Eq. 1, this is done as follows:

$$[\bar{b}_j, b_j] = [([C_t^i]_j - \bar{\theta} \cdot k), ([C_t^i]_j - \underline{\theta} \cdot k)] \quad (8)$$

Consequently, the TBST reflects different possible future developments of an agent's or the environment's behavior. Obviously, the smaller the size of the intervals  $\Delta\pi_j$  used to classify the atomic experiences, the narrower these corridors and the higher the accuracy of predicted behavior. However, this accuracy heavily depends on the number of gathered experiences the TBST is based on.

To make decisions, an agent can take the whole TBST, a subtree, or a single TBS into account. In case a single TBS is preferred, the agent could choose the TBS represented by a path that is obtained by, starting at the root, choosing the edge with the highest conditional probability. This TBS is not necessarily the most likely TBS, but, under the assumption that it is more important to predict an agent's behavior in the near future than in the distant future, this procedure yields very precise predictions (see Sect. 4).

As agents might show time-dependent behavior, predictions can be improved by selecting relevant TBSs on the basis of an agent's behavior in recent time steps. The TBST is then derived from these TBSs. For example, if the accuracy of the prediction of the residual load depends on the accuracy of predictions of the last  $x$  time steps, then it would be beneficial to take the prediction quality of the last  $x$  time steps into account when predicting the expected residual load's behavior by means of TBSs. This is achieved as follows: Whenever a contract  $C_t^i$  is concluded, the last  $x$  atomic experiences gained before the current time step are recorded. Regarding the experience  $E_t^i$  gained for  $C_t^i$  as a list, these recorded atomic experiences are then added to the front of  $E_t^i$ . When a TBST is generated to predict an agent's expected behavior during the validity of a contract, its behavior in the last  $x$  time steps is then used to select a subset of relevant TBSs from the set of TBSs with non-zero probability. Relevant TBSs are those whose intervals' midpoint in the first  $x$  time slots is closest to the rated atomic experiences in the last  $x$  time steps. In the course of deriving the TBST from the selected TBSs, these additional atomic experiences are ignored. We show in Sect. 4 that this selection process yields very good results.

## 4 Evaluation

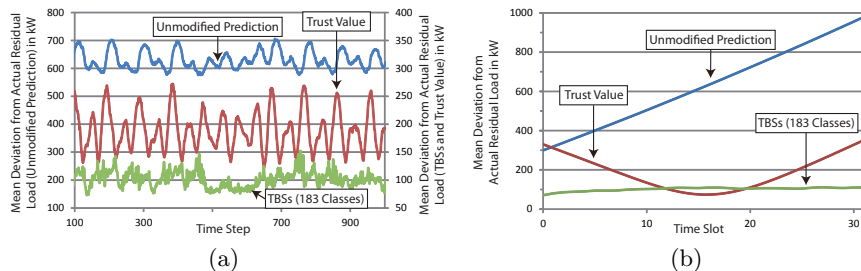
For evaluation, we regarded a setting of a single AVPP and an agent, called *predictor*, that represented all consumers as well as the AVPP’s subordinate stochastic power plants. Power plant models and power demand are based on real-world data. In each time step, the predictor predicted the residual load the AVPP had to satisfy in the next  $n + 1 = 32$  time steps (a single time step represented 15 minutes). For each time slot, the residual load prediction was created by adding a generated prediction error to the actual residual load. Therefore, in each time step, the predictor generated 32 prediction errors at random, using a gaussian distribution. To reflect time-dependent behavior, prediction errors depended on previous prediction errors. The AVPP’s objective was to minimize the deviation between the actual residual load and the residual load it expected. To predict the expected residual load, the AVPP used TBSs and a trust value for comparison. The TBSs and the trust value were determined on the basis of the last  $m = 50$  experiences (see Eq. 1) because we identified in preliminary tests that  $m = 50$  allows the trust value to achieve good results. The AVPP rated experiences and determined the trust value by means of Eq. 1 with  $\mathcal{T} = [-1, 1]$  and  $k = 9125$  kW (the AVPP’s maximum output/residual load). The AVPP pre-selected relevant TBSs on the basis of the prediction error of the last two time steps and selected a single TBS to predict the expected behavior as explained in Sect. 3.2. For each parametrization, we performed 100 simulation runs over 1000 time steps. Fig. 2(a) depicts the predictor’s mean behavior over time. Fig. 2(b) shows the mean prediction error for each of the 32 time slots of a prediction.

The predictions of the expected residual load are significantly improved when using TBSs instead of a trust value (see Tab. 1 and Fig. 2(a)). Compared to the situation in which the AVPP relied on the residual load prediction, the mean deviation  $\mu_{\Delta}$  between the expected and the actual residual load can be reduced by approximately 70% when using trust values compared to approximately 84% when using TBSs with  $\forall j \in \{0, \dots, n\} : |\Delta\mathcal{T}_j| = 183$ . TBSs thus reduce the trust value’s  $\mu_{\Delta}$  by 46%. In the power grid, it is of utmost importance to reduce the maximum deviation  $max_{\Delta}$  between the expected and actual residual load. Trust values reduce  $max_{\Delta}$  by 61%. TBSs obtain 78% and thus  $max_{\Delta}$  is 44% lower than the trust value’s  $max_{\Delta}$ . While we expected that TBSs benefit from a high number of classes  $|\Delta\mathcal{T}_j|$ , one can see in Tab. 1 that 183 classes were sufficient

Predicted Behavior	Number of Classes	Size of Classes	$\mu_{\Delta}$	$\sigma_{\Delta}$	$min_{\Delta}$	$max_{\Delta}$
1) Unmodified Prediction	-	-	629.1	32.8	573.6	707.3
2) Trust Value	-	-	188.9	34.9	121.2	273.2
3) Trust-Based Scenarios	$ \Delta\mathcal{T}_j  = 365$	$\approx 50$ kW	100.4	14.6	69.0	160.6
	$ \Delta\mathcal{T}_j  = 183$	$\approx 100$ kW	101.7	14.0	73.0	152.5
	$ \Delta\mathcal{T}_j  = 91$	$\approx 201$ kW	109.0	14.7	76.5	165.6
	$ \Delta\mathcal{T}_j  = 45$	$\approx 405$ kW	130.2	17.5	91.6	185.0

**Table 1.** Deviations between the actual and the expected residual load: the expected residual load is 1) equivalent to the predictor’s residual load prediction (“unmodified prediction”), 2) based on a trust value (see Sect. 2.1), 3) based on TBSs (see Sect. 3).

to perceive the underlying stochastic process. Fig. 2(b) shows that TBSs could estimate the behavior in the next 32 time steps much better than a trust value.



**Fig. 2.** Mean deviation of the expected residual load (unmodified residual load prediction, trust value, TBSs) from the actual residual load. Fig. 2(a) depicts this deviation over time. Fig. 2(b) shows the accuracy of the expected behavior for the 32 time slots.

Summarizing, compared to trust values, TBSs significantly increased the AVPP’s ability to predict the residual load’s behavior. Moreover, the risk the AVPP is exposed to, i.e., the maximum deviations, decreases considerably and the variation in prediction quality, i.e., the standard deviation  $\sigma_{\Delta}$  (see Tab. 1), declines. The latter increases the confidence in the predicted expected behavior.

## 5 Conclusion and Future Work

In this paper, we present trust-based scenarios (TBSs) as an instrument to precisely measure, predict, and deal with uncertainties introduced by agents and their environment. Agents generate TBSs by categorizing and grouping experiences gained in the past. Each TBS gives information about an agent’s or the environment’s possible future behavior and the probability of occurrence. As agents can take several TBSs into account, they are well-equipped to make informed and robust decisions in and to adapt to an uncertain and possibly malevolent environment. In contrast to the majority of current trust models, TBSs make less assumptions about the agent’s behavior. TBSs further mirror time-dependent behavior – a prerequisite for coping with strategic agents.

We illustrated our investigations on the basis of a self-organizing power management system that consists of AVPPs. This mission-critical system relies on predictions to ensure stable operation and availability. The evaluation showed that AVPPs benefit from TBSs since TBSs approximate the stochastic process underlying other agents’ or the environment’s behavior. Compared to the results achieved using trust values, TBSs reduce uncertainties by 46% on average.

In future work, we will devise an algorithm AVPPs use to calculate power plant schedules on the basis of TBSs, apply the concept of TBSs to the trust facet reliability, and investigate in which way TBSs can be used to proactively trigger reorganization in a self-organizing system in order to improve its resilience.

**Acknowledgments** This research is partly sponsored by the German Research Foundation (DFG) in the project “OC-Trust” (FOR 1085). The authors would like to thank F. Nafz, H. Seebach, and O. Kosak for the valuable discussions.

## References

1. 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 2010*, pages 40–42, October 2010.
2. F. Bouffard and F. Galiana. Stochastic security for operations planning with significant wind power generation. In *Power and Energy Society General Meeting—Conversion and Delivery of Electrical Energy in the 21st Century*, pages 1–11. IEEE, 2008.
3. B. Chang, S. Kuo, Y. Liang, and D. Wang. Markov chain-based trust model for analyzing trust value in distributed multicasting mobile ad hoc networks. In *Asia-Pacific Services Computing Conference, 2008*, pages 156–161. IEEE, 2008.
4. M. Densing. Hydro-electric power plant dispatch-planning—multi-stage stochastic programming with time-consistent constraints on risk. *Dissertation Abstracts International*, 68(04), 2007.
5. R. Hochreiter and G. Pflug. Financial scenario generation for stochastic multi-stage decision processes as facility location problems. *Annals of Operations Research*, 152(1):257–272, 2007.
6. F. Hussain, E. Chang, and T. Dillon. Markov model for modelling and managing dynamic trust. In *3rd IEEE International Conference on Industrial Informatics*, pages 725–733. IEEE, 2005.
7. F. Hussain, E. Chang, and O. Hussain. A robust methodology for prediction of trust and reputation values. In *Proc. of the 2008 ACM workshop on Secure web services*, pages 97–108. ACM, 2008.
8. R. Kiefhaber, G. Anders, F. Siefert, T. Ungerer, and W. Reif. Confidence as a Means to Assess the Accuracy of Trust Values. In *Proc. of the 11th IEEE Int. Conf. on Trust, Security and Privacy in Computing and Communications (TrustCom-2012)*. IEEE, 2012.
9. L. Li, Y. Wang, and V. Varadharajan. Fuzzy regression based trust prediction in service-oriented applications. *Autonomic and Trusted Computing*, pages 221–235, 2009.
10. R. C. Mayer, J. H. Davis, and F. D. Schoorman. An integrative model of organizational trust. *The Academy of Management Review*, 20(3):pp. 709–734, 1995.
11. V. Pappala and I. Erlich. Power System Optimization under Uncertainties: A PSO Approach. In *Swarm Intelligence Symposium (SIS 2008)*, pages 1–8. IEEE, 2008.
12. S. Ramchurn, D. Huynh, and N. Jennings. Trust in multi-agent systems. *The Knowledge Engineering Review*, 19(01):1–25, 2004.
13. N. V. Sahinidis. Optimization under uncertainty: state-of-the-art and opportunities. *Computers & Chemical Engineering*, 28(6-7):971–983, 2004.
14. 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 International Conference on Autonomic and Trusted Computing*. Springer, Oct. 2010.
15. B. Zhang, P. Luh, E. Litvinov, T. Zheng, F. Zhao, J. Zhao, and C. Wang. Electricity auctions with intermittent wind generation. In *Power and Energy Society General Meeting*, pages 1–8. IEEE, 2011.