Trust in control: a trust model for power system network assessment

Michael Brand[1,*], *Davood* Babazadeh[1], *Sebastian* Lehnhoff[1], and *Dominik* Engel[2]

[1]OFFIS – Institute for Information Technology, Energy Division, Escherweg 2, 26121 Oldenburg, Germany
[2]Salzburg University of Applied Sciences, Center for Secure Energy Informatics, Urstein Süd 1, 5412 Puch bei Hallein, Austria

Abstract. The question of whether a process variable transmitted from a device in the field to a power system control center is trustworthy is of high importance nowadays. Traditional bad data detection schemes have their limits in cases of elaborated cyberattacks and cascading failures in a system of systems such as a digitalized power system. This paper proposes a trust model designed for power system network assessment (PSNA). Different to other domains, where trust models already exist (e.g., OC-Trust for organic computing systems), the environment for PSNA is more centralized, and the focus lies on other facets than in organic computing due to the nature of the environment. Therefore, OC-Trust is tailored by categorizing its facets regarding their relevance for PSNA on the one hand. On the other hand, the trust model is extended to realize context-sensitive intersections of trust values. Furthermore, an example of an instantiation of the resulting PSNA-Trust model is given. Two security metrics and one credibility metric based on literature are presented as well as an equation for a context-sensitive intersection.

1 Introduction

The main task of a power system operator is to maintain the system in a normal, secure state as the operating conditions vary during the daily operation. That is indeed an important and responsible job. The control actions, an operator performs, depend on network assessment, while a state estimation is a basis for this network assessment. State estimation in power systems is to estimate the physical values with which the power system can be fully described [1].

Comparing today's and especially future power systems, i.e., smart grids, with conventional power systems, power system management nowadays is more dependent on information and communication technology (ICT), and there is an increasing threat of software failures and cyberattacks [2]. For the case of software failures, a common approach of systems in other domains is to shut down, reconfigure and restart the system. However, the fact that a power system is a critical infrastructure and the power supply must always be guaranteed makes such a procedure infeasible [2].

Coordinated (cyber) attacks pose another threat to modern power systems. The authors of the NISTIR 7628 guidelines state that “it is clear that cyber attacks or combined cyber/physical attacks pose a significant threat to the power grid” [2]. Liu et al. [3, 4] started in 2009 to investigate false data injection attacks (FDIAs) on a state estimator. They showed that, with control of several meters and knowledge of the hypothesized system model, i.e., the grid topology and line impedances, it is possible to affect the estimated system state and to remain undetected. Furthermore, there exist other works that discuss attack scenarios without complete knowledge of the system model [5]. On the other side, it is not so hard to generate comparatively accurate models of existing power systems with today's technology and communities [6].

There exist solutions in the literature for FDIAs, but they do not work out necessarily for all threat scenarios (cf. for example [7, 8, 9]). Most of them require a measurement redundancy that is not given in all systems (e.g., distribution grids). However, with the increase of complexity of power systems, the need for state estimation in all subsystems increases, too [10]. In such subsystems, data identified as compromised would need to be replaced (e.g., by simulated data).

The conclusion is that traditional state estimation with traditional bad data detection is not sufficient in case of coordinated attacks. Moreover, that may also hold for cascading failures if they behave like a coordinated attack.

Therefore, threats, when dealing with process variables received in a supervisory, control, and data acquisition (SCADA) system, are:

- The risk that a third party might have violated their integrity is higher.
- The chance that failures in upstream systems might have affected their correctness or accuracy is higher.
- The risk that either a third party or failures in upstream systems might have decreased their availability is more elevated.

But what if the compromise of a measurement or variable cannot be assessed definitely? What if there is only a given probability that the variable is compromised? For what probability values is a replacement of the variable reasonable?

Regarding those questions, a more general model is needed. It must describe an assumed compromise of

[*] Corresponding author: michael.brand@offis.de

process variables to make it possible to take action accordingly and depending on the network application and situation. With that, the following question gets into the focus: How to map different potential losses of trust, as given in examples above, in process variables?

There exists an elaborated trust model in the field of organic computing (OC) named OC-Trust [11]. The authors of [11] define (OC-) trust as "a multi-faceted concept that incorporates all constituting entities and users of a system and thus enables cooperation in systems of distributed entities."

The multi-faceted concept fits very well the different kinds of threats to trust in SCADA process variables described above. A SCADA system in the context of power system network assessment (PSNA) is not an OC system; it is more centralized. The master of a SCADA system must assess the trust of data received from systems (or agents) in the "field". Besides having a centralized system, another difference is the interaction with users, which is a key component in OC systems but limited to operators at SCADA masters.

Therefore, the first contribution of this work is the application of OC-Trust for PSNA, resulting in a trust model, which will be referenced as PSNA-Trust throughout the rest of the paper. The second contribution is an extension of the model to realize a context-sensitive intersection of trust values. That is needed because many applications expect a single scalar as a trust value and not a multi-faceted one. The third contribution is a conceptual instantiation of PSNA-Trust based on patterns and trust metrics proposed by literature.

The rest of the paper has the following structure: Section 2 gives an overview of related work, and Section 3 presents the application of OC-Trust for PSNA. The extension of the model to realize a context-sensitive intersection is explained in Section 4, followed by an example of the instantiation of PSNA-Trust in Section 5. Section 6 and 7 give an outlook about future work and conclude the paper respectively.

2 Related work

This section presents three kinds of related work. The first kind is work in the context of OC-Trust that is relevant for PSNA-Trust. Trust, with a focus on security, in energy data management as a multi-agent system was part of the research project Smart Nord. That is the second kind of related work. The third kind is specific metrics that can be used to assess certain facets of trust.

2.1 OC-Trust

Trust is defined in the research project OC-Trust [11] for organic computing systems as a multi-facet concept with the following facets:

- Functional correctness is “the quality of a system to adhere to its functional specification under the condition that no unexpected disturbances occur in the system’s environment” [11].
- Safety is “the quality of a system to be free of the possibility to enter a state or to create an output that may impose harm to its users, the system itself or parts of it, or to its environment” [11].
- Security is “the absence of possibilities to defect the system in ways that disclose private information, change or delete data without authorization, or to unlawfully assume the authority to act on behalf of others in the system” [11].
- Reliability is “the quality of a system to remain available even under disturbances or partial failure for a specified period of time as measured quantitatively by means of guaranteed availability, mean-time between failures, or stochastically defined performance guarantees” [11].
- Credibility is “the belief in the ability and willingness of a cooperation partner to participate in an interaction in a desirable manner. Also, the ability of a system to communicate with a user consistently and transparently” [11].

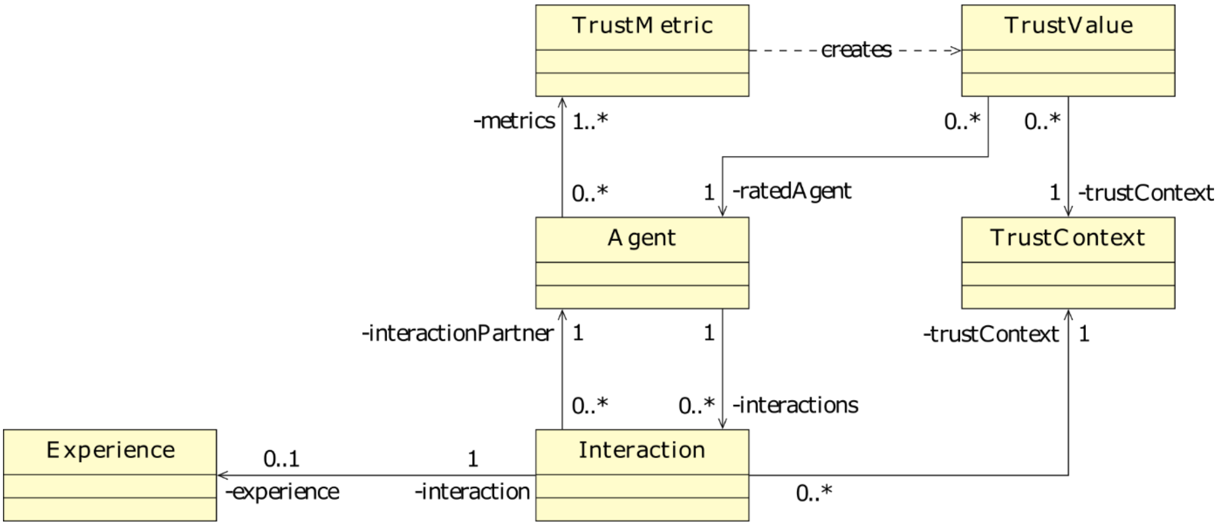

Fig. 1. Class diagram for trust in a multi-agent system [12].

• Usability is “the quality of a system to provide an interface to the user that can be used efficiently, effectively and satisfactorily that in particular incorporates consideration of user control, transparency and privacy” [11].

Steghöfer et al. [11] categorize the facets into two categories. The first category contains facets that allow building trust a priori: functional correctness, safety, and security. The second category contains facets that build trust at runtime: reliability, credibility, and usability.

Anders et al. [12] provide patterns to implement OC-Trust for single metrics. Fig. 1 shows the corresponding class diagram. Several trust metrics can be assigned to agents, who are interacting with each other. Every interaction happens in a specific trust context and can be used as experience (e.g., for a credibility analysis based on previous behavior). A trust metric creates trust values, which rate the corresponding agent. Trust values have, like interactions, a trust context. Anders et al. propose three different patterns based on their model: a direct trust, a reputation, and a trusted communities pattern. Direct trust is the trust of one agent in another agent based on interactions with this agent in the past (experience). If the trust is also based on the experience of other agents with the agent, it is called reputation. Another possibility is to form groups of trusted agents (trusted communities) [12].

2.2 Smart Nord

Another trust model, which is based on OC-Trust, is proposed in the research project Smart Nord [13-15]. Rosinger et al. [14] define a trust value as *tv* = [A_i, A_j, *c*, *tw*, *t*]. In this case, a trust value *tv* represents a unidirectional trust of an agent A_i in another agent A_j in a context *c* with a trustworthiness *tw* of A_j and a timeframe *t*. *tw* is a tuple containing values for the facets defined in OC-Trust.

Comparing this model with the patterns for OC-Trust, it is a model for direct trust, which also contains the interaction between two agents and respects different contexts (cf. Fig. 1). The timeframe can be seen as a specific context. Besides, the model by Rosinger et al. brings trust values of different facets (or in their terms trustworthiness values) together.

2.3 Metrics

The third part of this section deals with specific metrics that can be applied to measure certain facets of trust. Anders et al. [12] give an example for the instantiation of the direct trust pattern. The example is a deal between autonomous virtual power plants to have a contract about production and consumption. The context of such a contract consists of the product (generation or consumption) and the duration of the contract. Experience is the difference between the promised and the actual power during a contract. The so-called contract compliance metric then calculates a credibility value [12]. Unfortunately, the authors give no equation of how the value is calculated.

Another possibility is to take the credibility metric from Rosinger and Beer [15]. The context of their metric is the creation of dynamic, active power composites.

$$z = f(x,y) = \frac{1}{2}(1-x)(1-y) \qquad (1)$$

Equation 1 shows the calculation of an experience based on the promised and delivered power as well as on the promised reliability. x is the difference between promised and delivered power, y is the promised reliability, and the result is an experience, which Rosinger and Beer call singular credibility [15]. But like Anders et al., the authors give no equation of how single experiences can be aggregated to a credibility value.

Rosinger et al. [14] provide an elaborate a priori metric for security. They model the assessment of security measures and split the model into four parts. In the first part, threat scenarios, security attacks, and attacker types are modeled. They threaten security requirements, which are modeled in the second part. In the third part, security measures and the security assessment model are modeled. The security measures shall prevent security attacks. The security assessment model calculates the security value (trust value for the security facet). Security standards that support security measures are modeled in the fourth part.

$$Sec(Agent) = \frac{\sum_{i=1}^{\#secreq} A(i) \cdot St(i) \cdot Prio(i)}{\sum_{j=1}^{\#secreq} Prio(j)} \qquad (2)$$

$$Prio(i) \in \{1,2,3,4\};\ St(i) = \begin{cases} 1.2, \text{std. used} \\ 1, \text{otherwise} \end{cases}$$

Based on that assessment model, a security metric is calculated as shown in Equation 2 [14]. For all security requirements *i*, their assessment *A*(*i*) is the main input. It is weighted with 120% if security standards are used for the realization and with a priority between 1 (low) and 4 (high).

A metric presented by Liu et al. in their paper about an "abnormal traffic-indexed state estimation" [16] can be used to assess security at runtime. The authors aim at considering alerts from an intrusion detection system (IDS) for the weight of process variables in a state estimation. Therefore, they propose a so-called network impact factor matrix **Ω** that contains for each device (agent) the aggregation of all alerts.

$$\mathbf{\Omega} = DiagonalMatrix(\Omega_1, \Omega_2, \ldots, \Omega_n)$$

$$\Omega_\mathrm{i} = \sqrt{1 + \sum_{k\, \epsilon\, alert(device_i)} m^{priority(k)}} \qquad (3)$$

Equation 3 shows the calculation of **Ω**. *m* (*m* > 1) is a coefficient to weigh threats, *alerts*(*device*$_i$) is the set of alerts for the device *i*, and *priority(k)* is the priority of the alert *k*.

3 PSNA-Trust: applying OC-Trust

Recapitulating the difference in environments between OC-Trust and PSNA-Trust, the latter environment is typically more centralized and has fewer user interactions than OC systems. This difference opens a discussion about the role of each facet, defined for OC-Trust, in PSNA-Trust and how it could be assessed:

- Functional correctness: The question of whether an agent or device in the "field" adheres to its functional specifications is of high relevance for PSNA. For example, a metering device that does not measure as accurate as it should, based on its specifications, does not have a high functional correctness. This is particularly relevant if the current state's operational margin of error needs high accuracy.
- Safety: If an agent or device in the “field” can enter a state that may impose harm, it could result in an outage. For example, a transmission line could be overloaded or a server executing a critical function could fail. Therefore, safety is also of high relevance for PSNA. It can be assessed a priori by safety standards and methods (e.g., model checking) or during run-time, for example in the case of ICT devices, with ICT health monitoring tools.
- Security: All process variables in the context of PSNA are to be secured. With the knowledge about the data, an unauthorized individual, i.e., attacker, could learn about the system and plan further harmful attacks. Additionally, FDIAs are an example of what information modification (lack of integrity) can cause. They can lead to a pretense of a wrong system state and, accordingly, to harmful control actions. Security can be assessed a priori with an information security management system or during run-time with an IDS.
- Reliability: The question of whether an agent or device in the “field” remains available for a specified period is also of relevance for PSNA. If not, it may not be considered for PSNA, and its functionality must be taken over by other agents. Availability (as well as other Quality of Service/QoS metrics) can be assessed, for example in the case of ICT devices, with ICT health or network monitoring tools.
- Credibility: In the context of PSNA-Trust, one can distinguish between internal and external agents. Internal agents are under control of the network provider and connected by an ICT system also under control of the network provider. External agents are under third party control and may be connected by the internet to the network provider (e.g., a wind farm). The expectancy of accuracy as well as goodwill of those external agents needs to be part of credibility also considering past experiences with (e.g. data from) an agent. In summary, credibility is important and can be assessed with the help of contextual knowledge about devices or agents.
- Usability: Human operators play an important role within the power domain, in SCADA control rooms as well as in more decentralized decision making. Therefore, usability is also of relevance for PSNA, wherever data will be interpreted by a human.

In summar, all facets are of relevance for PSNA with domain-specific mappings and correlation. For practicality reasons, the rest of the paper focuses on security and credibility because that are the facets indicating whether a process variable may be compromised or not. All other facets are outside the scope of this paper but intended for future work.

4 Context-senstive intersection of trust values

As described in Section 2.1, Anders et al. [12] propose patterns for the use of trust metrics. The result of their approach is, under the assumption that the pattern is implemented for a couple of metrics, a facet-independent set of trust values. But it may be difficult to work with such a set of trust values. Network applications that shall use the trust values will rather expect a single trust value (e.g., a value in [0; 1]). An example of such a network application will be given in the next section. Therefore, this work extends the concept proposed in [12] with an intersection of trust values.

Because depending on the context the one or the other facet or metric may be of higher relevance, the

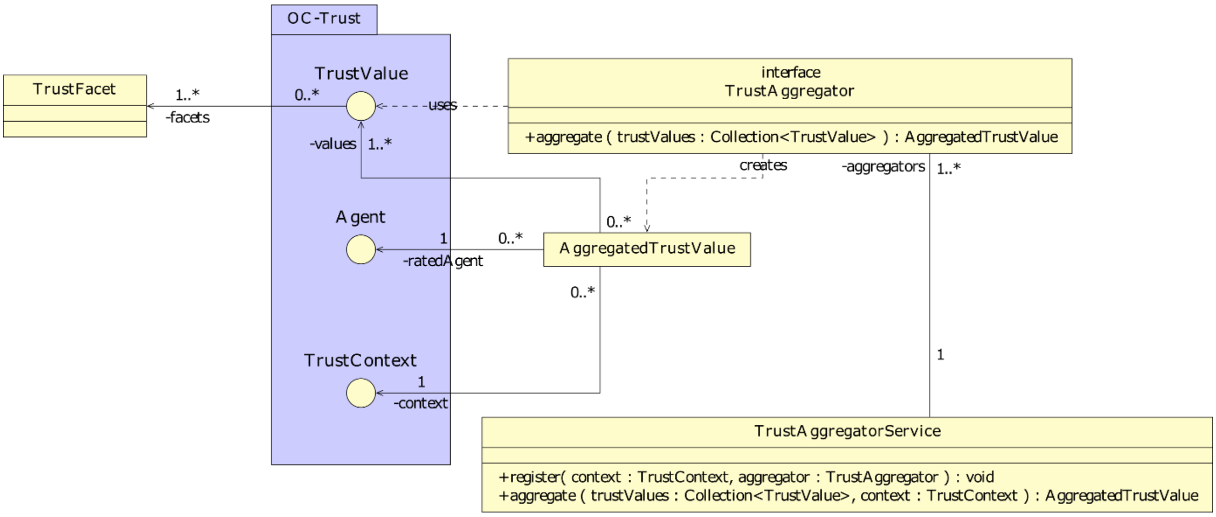
OC-Trust
TrustFacet
1..*
-facets
0..*
TrustValue
uses
interface
TrustAggregator
+aggregate (trustValues : Collection<TrustValue>) : AggregatedTrustValue
-values
1..*
creates
-aggregators
1..*
Agent
1
0..*
-ratedAgent
0..*
AggregatedTrustValue
0..*
1
TrustContext
1
-context
TrustAggregatorService
+register(context : TrustContext, aggregator : TrustAggregator) : void
+aggregate (trustValues : Collection<TrustValue>, context : TrustContext) : AggregatedTrustValue

Fig. 2. Class diagram for PSNA-Trust. The package OC-Trust is a simplified representation of the class diagram in Fig. 1.

intersection is context-sensitive. Another argument for a context-sensitivity is the possibility that the choice of the aggregation function may depend on the context. If, for example, it is known that in the current context some trust values are always outliers, it might be useful to work with the median as aggregation function. On the other side, in a very accurately calibrated context, the minimum might be a better aggregation function.

Fig. 2 shows the model of PSNA-Trust as an extension of the concept from Anders et al. The concept from Anders et al. [12] is represented in a simplified way to support readability (cf. Fig. 1 for a detailed figure of the concept). The data model for aggregated trust values is located to the right of the package "OC-Trust". They are the aggregated representation of single trust values, are associated with an agent, and have a context.

Trust aggregators create such aggregated trust values. The trust aggregator interface allows different implementations (e.g., median and minimum). In general, a trust aggregator implements for trust values in [0; 1] a function

$$f: [0,1]^n \to [0;1]. \tag{4}$$

A service for trust aggregators decides which aggregator to use in which context. Registering trust aggregators for particular contexts is possible. The last concept shown in Fig. 2 is the mapping of each trust value to at least one trust facet. It is also considered that single metrics and, therefore, the trust values created by them may be useful in several facets.

5 Instantiation of PSNA-Trust

This section gives an example of an instantiation of PSNA-Trust with a focus on the security and credibility facets. The network application is a special state estimation, called anomaly-sensitive state estimation [17]. The idea is to use anomaly detection techniques for some metrics to build trust at runtime and to perform a state estimation that considers those anomalies. PSNA-Trust can be used to model the anomaly scores. In the following, the trust metrics for the security and credibility facets are presented.

5.1 Metrics for the security facet

The security metric proposed by Rosinger et al. [14] (cf. Sec. 2.3, Eqn. 2) will be used as a “vulnerability metric”. It is a metric to build trust a priori that can be measured with an information security management system. “An Information Security Management System (ISMS) consists of the policies, procedures, guidelines, and associated resources and activities, collectively managed by an organization, in the pursuit of protecting its information assets” [18]. The ISMS provides the security requirements, the assessments, and the priorities for the environment under investigation. An example is given in [14].

An IDS can be used to measure a metric called “network anomaly metric”. It is a metric to build trust at runtime and based on a metric that Liu et al. proposed in [16] (cf. Sec 2.3, Eqn 3).

$$\begin{aligned} ano\big(agent(pv)\big) &= \frac{1}{\sqrt{\frac{1}{n}\sum_{k=1}^{n} w_k}} \\ &= \sqrt{\frac{n}{\sum_{k=1}^{n} w_k}} \end{aligned} \tag{5}$$

Equation 5 shows the calculation of the network anomaly metric. It is not directly taken from [16] because the metric in [16] converges towards infinity and a metric for PSNA-Trust must be in [0; 1]. The metric in Eqn. 5 is also weighted with the number of alerts to get a higher difference between scenarios with high priority alerts and scenarios with low priority alerts as an example will demonstrate. w_k ($w_k \geq 1$) is a coefficient to weigh an alert k including the threat priority ($m^{priority(k)}$ in [16]). n is the number of alerts for the agent.

For example, an attacker called Mallory may perform his attack on agent A_1 in two steps executed remotely [16]. First, he changes the password. Mallory could have got access to a password by guessing a not changed default password. After Mallory changed the password, he changes parameters of the system. A proper configured IDS fires alerts for both events. Let w be 9 for the password modification and 16 for the parameter modification. Let there further be two unimportant alerts with w = 1. As a comparison, an agent A_2, who is not attacked but who causes several unimportant alerts, is considered. Then, the network anomaly values for A_1 and A_2 would be

$$ano(A_1) = \sqrt{\frac{4}{9+16+1+1}} \approx 0.38; \tag{6}$$

$$ano(A_2) = \sqrt{\frac{4}{2+2+1+1}} \approx 0.89. \tag{7}$$

Correspondingly, process variables from A_1 would be rated as less trustworthy.

5.2 Metrics for the credibility facet

A “residual metric” is considered for the credibility facet. Let the expected value of a process variable be the value that is calculated from the system model and the estimated system state. The residual is then the absolute difference between the expected and received value of a process variable. This residual for a single interaction with an agent can be stored as an experience and used to measure the credibility of that agent.

$$\exp_{res}(pv) = 1 - \sqrt{|r(pv) - std(pv)|} \tag{8}$$

Equation 8 shows the calculation of the experience. It is inspired by [15] but different because of the context of PSNA-Trust. $r(pv)$ is the residual of a process variable normalized to [0;1]. $std(pv)$ is the known standard deviation of the process variable in p.u. and in [0; 1] (the reliability promised by the agent would be 1-$std(pv)$). A higher standard deviation leads to higher residuals considered to be credible. The square root function is applied to increase the influence of the difference between residual and promised reliability.

$$res(pv) = \frac{\sum_{e \,\epsilon\, \exp_{res}(pv)} t_e \cdot e}{\sum_{e \,\epsilon\, \exp_{res}(pv)} e} \qquad (9)$$

The resulting experience can be used to measure the credibility as shown in Equation 9. t_e is a time weigh factor for an experience to allow different weights for newer and older experiences.

For example, Mallory, who got control over an agent with the process variable pv_1, could inject his data that is different from the expected data. Let 1.12 p.u. be the received value and 0.001 p.u. be the standard deviation for pv_1. The state estimation results in an expected value of 1.0 p.u. As a comparison, a process variable pv_2, which is not attacked but is not as reliable ($std(pv_2)$ = 0.1), is considered. Then, the experiences for pv_1 and pv_2 would be

$$\exp_{res}(pv_1) = 1 - \sqrt{|0.12 - 0.001|} \approx 0.66; \qquad (10)$$

$$\exp_{res}(pv_2) = 1 - \sqrt{|0.12 - 0.1|} \approx 0.86. \qquad (11)$$

Correspondingly, process variables from A_1 would be rated as less trustworthy.

5.3 Context-sensitive intersection

An example context to intersect the three metrics is an anomaly-sensitve state estimation in a high voltage power grid [17].

$$trust(pv) = \frac{1}{7} \cdot vul[agent(pv)] + 3 \cdot ano[agent(pv)] + 3 \cdot res(pv)] \qquad (12)$$

In that context, the proposed aggregation function is a weighted arithmetic average shown in Equation 12. The vulnerability metric (*vul*[*agent*(*pv*)]) is in fact of high relevance, but usually, all relevant agents (i.e., remote terminal units (RTUs)) are protected the same a priori. However, that does not mean necessarily that no cyber-attacker has control over an RTU or a communication channel. Therefore, the network anomaly metric is weighted more. The residual metric also has a high weight, because failures may also occur at the RTUs or during data transmission.

6 Future work

This paper focuses on the security and credibility because that are the facets relevant for detecting compromised process variables. In future work, metrics for all facets shall be discussed in detail and evaluated.

The evaluation of PSNA-Trust is planned in the research project Smart Grid Cyber-Resilience Laboratory[1] (CybResLab). The infrastructure of the CybResLab consists of state-of-the-art devices and software that can also be found in the infrastructures of service providers. That infrastructure is planned to be modeled in an ISMS in terms of its security requirements and vulnerabilities. The anomaly-sensitive state estimation, which is mentioned in Sec. 5, will be implemented and evaluated in the CybResLab. All three metrics described in Sec. 5 and further metrics for the other facets mentioned in Sec. 3 are intended to be implemented in the CybResLab, too.

6 Future work

This paper proposed a model for trust in power system network operations, called PSNA-Trust, that is based on the elaborated model of OC-Trust.

As the first part of the concept of PSNA-Trust, the facets of OC-Trust in the context of PSNA and tools to assess them were discussed. The second part was the extension of the model for patterns to implement OC-Trust with trust aggregators and a service to choose them context-sensitively. A trust aggregator intersects trust values from different metrics (e.g., metrics for different facets). The result is a single trust value that can be used in applications. The intersection becomes context-sensitive by the trust aggregator service that chooses the trust aggregator according to a context.

An instantiation of the model for an anomaly-sensitive state estimation was the third part. Appropriate metrics for the security and the credibility facet were presented. For the security facet, a vulnerability metric, which can build trust a priori, and a network anomaly metric, which can build trust at runtime, were proposed. A residual metric was suggested for the credibility facet. Furthermore, a weighted arithmetic average function was presented to aggregate trust values. The evaluation shall be part of the research project Smart Grid Cyber-Resilience Laboratory.

References

1. A. Abur, A.G. Exposito, *Power System State Estimation: Theory and Implementation*, **1st edn.** (2004)
2. V.Y. Pillitteri, T.L. Brewer, *Guidelines for Smart Grid Cybersecurity,* Technical report (NIST) (2014)
3. Y. Liu, P. Ning., M.K. Reiter, *Proceedings of the ACM CCS'09*, pp. 21 – 32 (2009)
4. Y. Liu, P. Ning, M.K. Reiter, ACM TISSEC **14(1)**, 13 – 29 (2011)
5. A. Teixeira, S. Amin, H. Sandberg, K.H. Johansson, S.S. Sastry, *Proceedings of the 49th IEEE CDC*, pp. 5991 – 5998 (2010)
6. J. Rivera, J. Leimhofer, H.A. Jacobsen, Computer Science - Research and Development, **32(1-2)**, pp. 13 – 23 (2017)

[1] https://www.offis.de/en/offis/project/cybreslab.html

7. S. Cui, Z. Han, S. Kar, T.T. Kim, H.V. Poor, A. Tajer, IEEE Signal Processing Magazine, **29(5)**, pp. 106 – 115 (2012)
8. Y. Mo, B. Sinopoli, IEEE Transactions on Automatic Control, **60(4)**, 1145 – 1151 (2015)
9. K. Xiong, P. Ning, *Proceedings of the 2017 ACM SAC*, pp. 2192 – 2197 (2015)
10. Y.-F. Huang, S. Werner, J. Huang, N. Kashyap, V. Gupta, IEEE Signal Processing Magazine, **29(5)**, 33—43 (2012)
11. J.-P. Steghöfer, R. Kiefhaber, K. Leichtenstern, Y. Bernard, L. Klejnowski, W. Reif, T. Ungerer, E. André, J. Hähner, C. Müller-Schloer, *Proceedings of the ATC*, pp. 62 – 76 (2010)
12. G. Anders, J.-P. Steghöfer, F. Siefert, W. Reif, *Proceedings of the IEEE SASOW 2011*, pp. 35 – 40 (2011)
13. C. Rosinger, M. Uslar, J. Sauer, *Proceedings of the EnviroInfo 2013*, pp. 258 – 264 (2013)
14. C. Rosinger, M. Uslar, J. Sauer, *Proceedings of the EnviroInfo 2014*, pp. 373 – 380 (2014)
15. C. Rosinger, S. Beer, Informatik-Spektrum, **38(2)**, pp. 103 – 110 (2015)
16. T. Liu, Y. Sun, Y. Liu, Y. Gui, Y. Zhao, D. Wang, C. Shen, Future Generation Computer Systems, **49**, pp. 94 – 103 (2015)
17. M. Brand, S. Ansari, F. Castro, R. Chakra, B. Hage Hassan, C. Krüger, D. Babazadeh, S. Lehnhoff, *Proceedings of the 2019 IEEE Milan PowerTech*, pp. 1 – 6 (2019)
18. ISO/IEC 27001:2016 (2016)