






An Extension of Reciprocal Logic for Trust Reasoning: A Case Study in PKI

Sameera Basit  and Yuichi Goto  

Department of Information and Computer Sciences, Saitama University,
Saitama 338-8570, Japan
{sameera,gotoh}@aise.ics.saitama-u.ac.jp

Abstract. Trust relationship is one of the kinds of reciprocal relationship and basis of communications among agents, especially in open and decentralized systems, e.g., public key infrastructure (PKI). In such systems, it is difficult to know whether an agent that is required to communicate with us can be trusted or not. Thus, it is indispensable to calculate the degree of trust of the target agent by using already known facts, hypotheses, and observed data. Trust reasoning is a process to calculate the degree of trust of the target agents. Although the current extension of reciprocal logic is an expectable candidate for a logic system underlying trust reasoning, it has a limitation when we deal with trust messages from other agents as a proposition. From the viewpoint of predicate logic, the current extension of reciprocal logic deals with messages from other agents as countable objects and are represented as individual constants. However, Demolombe represents messages from other agents as a proposition. From the viewpoint of expressive power, Demolombe's approach is better and the current extension of reciprocal logic is not enough. Following the Demolombe's approach, we introduced modal operators *Bel* and *Inf* and add several axioms to the current extension of reciprocal logic and a case study of trust reasoning based on the proposed extension in PKI is also presented.

Keywords: Trust reasoning · Strong relevant logic · Reciprocal logic

1 Introduction

A trust relationship is one of the important reciprocal relationships in our society and cyberspace. There are many reciprocal relationships that must concern two parties, e.g., parent-child relationship, relative relationship, friendship, cooperative relationship, complementary relationship, trade relationship, buying and selling relationship, and so on [6]. Especially, the trust relationship is the basis of communications among agents (human to human, human to system, and system to system), and the basis of the decision-making of the agents.

Trust reasoning is an indispensable process for establishing trustworthy and secure communication under open and decentralized systems that include multi-agents. In open and decentralized systems, although it is difficult to know

whether an agent that is required to communicate with us can be trusted or not before communication with it, we want to know whether the agent is trusted or not to establish trustworthy and secure communication, e.g., public key infrastructure (PKI). Thus, we should calculate the degree of trust of the target agent by using already known facts, hypotheses, and observed data. Trust reasoning is a process to calculate the degree of trust of the target agents and messages that come from other agents.

Although reciprocal logic [6] is an expectable candidate for a logic system underlying trust reasoning, current reciprocal logic cannot deal with several trust properties. Cheng proposed reciprocal logic [6] as a logic system underlying reasoning for such reciprocal relationships and formalized trust relationships between agents and agents, and agents and organizations in the logic. On the other hand, there are various trust properties for trust relationships, i.e., “an agent α trusts another agent β about a message from β in *property*” where *property* are sincerity, validity, vigilance, credibility, cooperativity, completeness, and so on [10, 13–16]. The trust properties focus on not only trust relationships between agents and agents but also trust relationships between agents and messages that are informed by other agents. We cannot describe the trust relationship between agents and messages in current reciprocal logic.

An extension of reciprocal logic is demanded to deal with trust relationships between agents and messages which includes trust properties. We proposed an extension of reciprocal logic [4]. In the extension, messages that come from other agents are regarded as countable objects and are represented as individual constants from the viewpoint of predicate logic. On the other hand, Demolombe [8] defined several trust properties and formalized them. He regarded messages from agents as the beliefs of the agents and represented them as propositions (logical formulas) from the viewpoint of predicate logic. From the viewpoint of expressive power, Demolombe’s approach is better than the approach of our last extension, so the last extension is not enough.

This paper presents a new extension of reciprocal logic that can deal with trust properties and shows a case study of trust reasoning based on the proposed extension in PKI. We introduced two modal operators *Bel* and *Inf* into reciprocal logic to represent trust relationships between agents and messages from other agents according to Demolombe’s approach. We also add several axioms into reciprocal logic. One of the reasons why we want to know trustworthy agents is to reduce the process of whether the messages from the target agents are correct or not. In other words, the reason is to filter messages. Under this consideration, we defined axioms representing how agent α deals with a message from agent β if α trusts β in some trust property. Finally, we conducted a case study of trust reasoning based on the proposed extension in PKI. The case study shows that the proposed extension can deal with several trust properties. Thus, we can conclude that the proposed extension is an expectable candidate for a logic system underlying trust reasoning. The rest of the paper is organized as follows: Sect. 2 presents a summary of survey results concerning trust relationships and trust properties, the limitations of reciprocal logic, and the last extension. Section 3 presents the new extension of reciprocal logic. In Sect. 4, we showed a case study

of trust reasoning based on the new extension in PKI. Some concluding remarks are given in Sect. 5.

2 Related Works

2.1 Trust Relationship and Trust Properties

Trust is a common phenomenon, and it is an essential element in a relationship that concerns two parties. These two parties are usually regarded as a trustor and a trustee when we consider a trust relationship, i.e., a trustee provides trustworthy messages to make the trustor trust the trustee. Trust relationships between parties are more tractable with the aid of trust properties.

Many previous works focus on trust properties. Several authors in [9, 11] target a certain property and focus on one dimension only whereas other authors in [10, 16] deal with trust in the reliability, credibility, and collectively. The authors in [14, 15] provide a classification of trust properties from the viewpoint of the trustor and trustee and regard them as essential in the establishment of a trust relationship.

In the context of trust not all the information from the other agent can be taken as a true message, i.e., an agent α trusts another agent β with respect to some property means that α believes that β satisfies this property. Demolombe [8] defined several trust properties. His definitions are as follows.

- *Sincerity*: An agent α trusts in the sincerity of an agent β if β informs α about a proposition p then β believes p .
- *Validity*: An agent α trusts in the validity of an agent β if β informs α about a proposition p then p is the case.
- *Completeness*: An agent α trusts in the completeness of an agent β if p is the case then β informs α about p .
- *Cooperativity*: An agent α trusts in the cooperativity of an agent β if β believes p then β informs α about p .
- *Credibility*: An agent α trust in the credibility of an agent β if β believes p then p is the case.
- *Vigilance*: An agent α trust in the vigilance of an agent β if p is the case then β believes p .

Demolombe also provided a formal definition of the above properties. His formalization is based on classical mathematical logic.

2.2 Reciprocal Logic and Its Extension

Reciprocal logic was proposed by Cheng [6] as a logic system underlying reasoning for a reciprocal relationship. Classical mathematical logic and its various conservative extensions are not suitable for logic systems underlying reasoning because they have paradoxes of implication [2, 3]. Strong relevant logic has rejected those paradoxes of implication and is considered the universal basis of

various applied logic for knowledge representation and reasoning [5, 7]. Thus, strong relevant logic and its conservative extensions are candidates for logic systems underlying reasoning. Reciprocal logic is one of the conservative extensions of strong relevant logic to deal with various reciprocal relationships. Reciprocal logic provides primitive predicates representing trust relationships between an agent and another agent, and between an agent and an organization, defined predicates based on the primitive predicates, and several axioms that include the predicates [6]. Let pe_1 , pe_2 , and pe_3 be individual variables representing agents, and let o_1 and o_2 be individual variables representing organizations. The primitive predicates are as follows

- $TR(pe_1, pe_2)$: pe_1 trusts pe_2 .
- $B(pe_1, o_1)$: agent pe_1 belongs to organization o_1

Defined predicates based on the above primitive predicate are as follows.

- $NTR(pe_1, pe_2) =_{df} \neg(TR(pe_1, pe_2))$ ($NTR(pe_1, pe_2)$ means pe_1 does not trust pe_2).
- $TREO(pe_1, pe_2) =_{df} TR(pe_1, pe_2) \wedge (TR(pe_2, pe_1))$ ($TREO(pe_1, pe_2)$ means pe_1 and pe_2 trust each other.)
- $ITR(pe_1, pe_2, pe_3) =_{df} \neg(TR(pe_1, pe_2) \wedge TR(pe_1, pe_3))$ ($ITR(pe_1, pe_2, pe_3)$ means pe_1 does not trust both pe_2 and pe_3 (Incompatibility))
- $XTR(pe_1, pe_2, pe_3) =_{df} (TR(pe_1, pe_2) \vee TR(pe_1, pe_3)) \wedge (NTR(pe_1, pe_2) \vee NTR(pe_1, pe_3))$ ($XTR(pe_1, pe_2, pe_3)$ means pe_1 trusts either pe_2 or pe_3 but not both (exclusive disjunction)).
- $JTR(pe_1, pe_2, pe_3) =_{df} \neg(TR(pe_1, pe_2) \vee TR(pe_1, pe_3))$ ($JTR(pe_1, pe_2, pe_3)$ means pe_1 trusts neither pe_2 nor pe_3 (joint denial)).
- $TTR(pe_1, pe_2, pe_3) =_{df} (TR(pe_1, pe_2) \wedge TR(pe_1, pe_3)) \Rightarrow TR(pe_1, pe_3)$. ($TTR(pe_1, pe_2, pe_3)$ means pe_1 trusts pe_3 if pe_1 trusts pe_2 and pe_2 trusts pe_3).
- $CTR(pe_1, pe_2, pe_3) =_{df} (TR(pe_1, pe_3) \Rightarrow (TR(pe_2, pe_3)))$ ($CTR(pe_1, pe_2, pe_3)$ means pe_2 trusts pe_3 if pe_1 trusts pe_3 .)
- $NCTR(pe_1, pe_2, pe_3) =_{df} (\neg TR(pe_1, pe_3) \Rightarrow (TR(pe_2, pe_3)))$ ($NCTR(pe_1, pe_2, pe_3)$ means pe_2 trusts pe_3 if pe_1 does not trusts pe_3)
- $CNTR(pe_1, pe_2, pe_3) =_{df} \neg(TR(pe_1, pe_3) \Rightarrow \neg(TR(pe_2, pe_3)))$ ($CNTR(pe_1, pe_2, pe_3)$ means pe_2 does not trusts pe_3 if pe_1 does not trusts pe_3)
- $TRpo(pe_1, o_1) =_{df} \forall pe_2(B(pe_2, o_1) \wedge (TR(pe_1, pe_2)))$ ($TRpo(pe_1, o_1)$ means pe_1 trusts o_1).
- $NTRpo(pe_1, o_1) =_{df} \forall pe_2(B(pe_2, o_1) \wedge (NTR(pe_1, pe_2)))$ ($NTRpo(pe_1, o_1)$ means pe_1 does not trusts o_1).
- $TRop(o_1, pe_1) =_{df} \forall pe_2(B(pe_2, o_1) \wedge (TR(pe_2, pe_1)))$ ($TRop(o_1, pe_1)$ means o_1 trusts pe_1).
- $NTRop(o_1, pe_1) =_{df} \forall pe_2(B(pe_2, o_1) \wedge (NTR(pe_2, pe_1)))$ ($NTRop(o_1, pe_1)$ means o_1 does not trusts pe_1).
- $TRoo(o_1, o_2) =_{df} \forall pe_1 \forall pe_2(B(pe_1, o_1) \wedge (B(pe_2, o_2)) \wedge (TR(pe_1, pe_2)))$ ($TRoo(o_1, o_2)$ means o_1 trusts o_2).

- $NTRoo(o_1, o_2) =_{df} \forall pe_1 \forall pe_2 (B(pe_1, o_1) \wedge (B(pe_2, o_2)) \wedge (NTR(pe_1, pe_2) \wedge (NTRoo(o_1, o_2) \text{ means } o_1 \text{ does not trusts } o_2)).$

Through the above definitions of predicates, we can consider that reciprocal logic focuses on only trust relationship between an agent and other agent, and between an agent and an organization.

Axioms of the reciprocal logic are as follows:

- TR1: $\neg(\forall pe_1 \forall pe_2 (TR(pe_1, pe_2) \Rightarrow TR(pe_2, pe_1)))$
- TR2: $\neg(\forall pe_1 \forall o_1 (TRpo(pe_1, o_1) \Rightarrow TRop(o_1, pe_1)))$
- TR3: $\neg(\forall o_1 \forall pe_1 (TRop(o_1, pe_1) \Rightarrow TRpo(pe_1, o_1)))$
- TR4: $\neg(\forall o_1 \forall o_2 (TRoo(o_1, o_2) \Rightarrow TRoo(o_2, o_1)))$
- TR5: $\neg(\forall pe_1 \forall pe_2 \forall pe_3 (TR(pe_1, pe_2) \wedge TR(pe_2, pe_3) \Rightarrow TR(pe_1, pe_3)))$
- TR6: $\neg(\forall pe_1 \forall pe_2 \forall o_1 (TRpo(pe_1, o_1) \wedge TRop(o_1, pe_2) \Rightarrow TR(pe_1, pe_2)))$
- TR7: $\neg(\forall pe_1 \forall pe_2 \forall o_1 (TRop(o_1, pe_1) \wedge TR(pe_1, pe_2) \Rightarrow TRop(o_1, pe_2)))$
- TR8: $\neg(\forall o_1 \forall o_2 \forall o_3 (TRoo(o_1, o_2) \wedge TRoo(o_2, pe_3) \Rightarrow TR(o_1, o_3)))$

$TrTcQ =_{df} TcQ + \{TR1, \dots, TR8\}$, $TrEcQ =_{df} EcQ + \{TR1, \dots, TR8\}$, and $TrRcQ =_{df} RcQ + \{TR1, \dots, TR8\}$ are the minimal logic systems of reciprocal logic where TcQ , EcQ , and RcQ are logic systems of the first order predicate strong relevant logics [6].

We proposed an extension of reciprocal logic to deal with trust properties [4]. Current reciprocal logic cannot deal with the trust properties explained in Sect. 2.1 because it does not provide a representation method of the trust relationship between an agent and a message that came from other agents. We introduced several predicates to represent the trust relationship between an agent and a message into reciprocal logic. In the extension, messages that come from other agents are regarded as countable objects, and are represented as individual constants from the viewpoint of predicate logic. However, the extension is not enough to represent the trust properties explained in Sect. 2.1.

3 A New Extension of Reciprocal Logic

Although we proposed an extension of reciprocal logic for trust reasoning [4], we regarded messages that come from agents as countable objects (individual constants). From the viewpoint of applications of trust reasoning, we should regard the messages from other agents as propositions like Demolombe's logic system [8]. Thus, we replaced the trust properties in the first extension with Demolombe's logic system like logical formulas.

At first, we add a predicate " $TR(pe_1, pe_2, PROP)$ " where pe_1 and pe_2 are agents, and $PROP$ is individual constant that represents trust properties: sincerity, validity, completeness, cooperativity, credibility, and vigilance into reciprocal logic. For example, " $TR(pe_1, pe_2, sincerity)$ " means " pe_1 trusts pe_2 in sincerity". Note that " $TR(pe_1, pe_2, all)$ " means " pe_1 trusts pe_2 in all trust properties", i.e., " $TR(pe_1, pe_2)$ " in reciprocal logic is as same as " $TR(pe_1, pe_2, all)$ " in our new extension.

Secondly, we introduced two modal operators $Bel_i(A)$ and $Inf_{i,j}(A)$ used in Demolombe's logic system into reciprocal logic to represent the trust relationship between agents and information that comes from other agents. The two modal operators follow the KD systems of modal logic [8].

$Bel_i(A)$: an agent i believes that a proposition A is true.
 $Inf_{i,j}(A)$: an agent i has informed an agent j about A .

Finally, we add new axioms into reciprocal logic.

ERcL1: $\forall i \forall j (TR(i, j, sincerity) \Rightarrow (Inf_{j,i}(A) \Rightarrow Bel_j(A)))$
 ERcL2: $\forall i \forall j (TR(i, j, validity) \Rightarrow (Inf_{j,i}(A) \Rightarrow A))$
 ERcL3: $\forall i \forall j (TR(i, j, vigilance) \Rightarrow (A \Rightarrow Bel_j(A)))$
 ERcL4: $\forall i \forall j (TR(i, j, credibility) \Rightarrow (Bel_j(A) \Rightarrow A))$
 ERcL5: $\forall i \forall j (TR(i, j, cooperativity) \Rightarrow (Bel_j(A) \Rightarrow Inf_{j,i}(A)))$
 ERcL6: $\forall i \forall j (TR(i, j, completeness) \Rightarrow (A \Rightarrow Inf_{j,i}(A)))$
 BEL: $\forall i (Bel_i(A \Rightarrow B) \Rightarrow (Bel_i(A) \Rightarrow Bel_i(B)))$.

We summarize our new extension of reciprocal logic. Let RcL be all axioms of reciprocal logic. Our new extension is $RcL \cup \{ERcL1, \dots, ERcL6, BEL\}$.

4 A Case Study of Trust Reasoning Based on New Extension in PKI

4.1 Scenario

We present a simple scenario in PKI inspired from [12]. We have formalized the scenario and applied the trust reasoning process based on new extension in PKI.

Suppose that a certificate c_2 is signed by the subject of a certificate c_1 with the private key corresponding to the public key of c_1 . Agent e_1 trusts the certificate c_1 because c_1 is informed by its parent agent. In PKI, we consider that every agent trusts its parent agent in its validity, i.e., $\forall e (TR(e, parent(e), validity))$. Moreover, agent e_2 informs agent e_1 about certificate c_2 . We assume that agent e_1 trusts agent e_2 in its completeness, i.e., $TR(e_1, e_2, completeness)$. Agent e_1 does not trust the certificate but wishes to use certificate c_2 . We need to know that whether certificate c_2 informed by agent e_2 is valid or not. From these two trust relationship: $TR(e_1, parent(e_1), validity)$ and $TR(e_1, e_2, completeness)$, we can conclude that certificate c_2 informed by agent e_2 is valid, i.e., $Inf_{e_2,e_1}(is Valid(c_2))$.

4.2 Formalization

To formalize the above scenario, we defined following constants, functions, and predicates.

- Individual variables:
 - e : an agent

- c, c' : certifications
- Individual constants:
 - e_1, e_2 : agents
 - c_1, c_2 : certifications
 - $today$: date of today
- Functions:
 - $I(c)$: Issuer of certification c .
 - $S(c)$: Subject of certification c .
 - $PK(c)$: Public key of c .
 - $SK(c)$: Share key of c .
 - $DS(c)$: Start date of c .
 - $DE(c)$: End date of c .
 - $Sig(c)$: Signature of c .
 - $parent(e)$: The parent of agent e .
- Predicates:
 - $inCRL(c)$: c is in certification revocation list.
 - $isValid(x)$: x is valid.
 - $isSigned(x, k)$: x is message signed by key k .
 - $x = y$: x is equal to y .
 - $x \leq y$: x is equal to or less than y .
 - $x < y$: x is less than y .

In PKI, we can assume following empirical theories.

PKI1: $\forall e(TR(e, parent(e), validity))$

(Any agent trusts its parent agent in validity).

PKI2: $\forall c(\exists c'((isValid(c')) \wedge (I(c) = S(c')) \wedge (isSigned(c, PK(c')))) \Rightarrow isValid(Sig(c)))$

PKI3: $\forall c((isValid(Sig(c)) \wedge (DS(c) \leq today) \wedge (today < DE(c)) \wedge \neg inCRL(c)) \Rightarrow isValid(c))$

(PKI2 and PKI3 allows to verify the signature and certificate itself on the basis of another certificate whose validity has been proven).

From scenario, we can assume following logical formulas.

P1: $I(c_2) = S(c_1)$

(This observed facts are used as a premises in our reasoning process and it is true in this scenario only).

P2: $isSigned(c_2, PK(c_1))$

(A certificate c_2 is signed by the subject of certificate c_1 with the private key corresponding to the public key of c_1).

P3: $Inf_{parent(e_1), e_1}(isValid(c_1))$

(The parent agent of e_1 has informed e_1 about “certificate c_1 is valid”).

P4: $TR(e_1, e_2, completeness)$

(our assumption)

P5: $DS(c_2) \leq today$ (our assumption)

P6: $today < DS(c_2)$ (our assumption)

P7: $\neg inCRL(c_2)$ (our assumption).

In the next section, we used inference rules of reciprocal logic and our new extension for trust reasoning. The inference rules are as follows.

$\Rightarrow E$: “from A and $A \Rightarrow B$ to infer B ” (Modus Ponens)

$\wedge I$: “from A and B infer $A \wedge B$ ” (Adjunction).

4.3 Trust Reasoning Process

According to the above formalization, we can reason out the expected conclusion “ $Inf_{e_2, e_1}(is\ Valid(c_2))$ ”. The reasoning process is as follows.

1. $Inf_{j,i}(A) \Rightarrow A$ [Deduced from PKI1 and ERcL2 with $\Rightarrow E$]
2. $is\ Valid(c_1)$ [Deduced from P3 and 1 with $\Rightarrow E$]
3. $is\ Valid(c_1) \wedge (I(c_2) = S(c_1)) \wedge is\ Signed(c_2, PK(c_1))$ [Deduced from 2, P1 and P2 with $\wedge I$]
4. $\exists c'((is\ Valid(c')) \wedge (I(c_2) = S(c')) \wedge (is\ Signed(c_2, PK(c')))) \Rightarrow is\ Valid(Sig(c_2))$ [Substitute c_2 for c in PKI2]
5. $is\ Valid(Sig(c_2))$ [Deduced from 3 and 4 with $\Rightarrow E$]
6. $is\ Valid(Sig(c_2)) \wedge (DS(c_2) \leq today) \wedge (today < DE(c_2)) \wedge \neg inCRL(c_2)$ [Deduced from 5 and P5 to P7 with $\wedge I$]
7. $(is\ Valid(Sig(c_2)) \wedge (DS(c_2) \leq today) \wedge (today < DE(c_2)) \wedge \neg inCRL(c_2)) \Rightarrow is\ Valid(c_2)$ [Substitute c_2 for c in PKI3]
8. $is\ Valid(c_2)$ [Deduced from 6 and 7 with $\Rightarrow E$]
9. $A \Rightarrow Inf_{j,i}(A)$ [Deduced from P4 and ERcL6 with $\Rightarrow E$]
10. $Inf_{e_2, e_1}(is\ Valid(c_2))$ [Deduced from 8 and 9 with $\Rightarrow E$].

Having completed the trust reasoning process, we can therefore have $Inf_{e_2, e_1}(is\ Valid(c_2))$ derived from the fact $Inf_{parent(e_1), e_1}(is\ Valid(c_1))$.

Instinctively, it represents a trust transfer. Agent e_1 trust in certificate c_2 informed by an agent e_2 is transferred from its trust in the validity of its parent entity. In PKI, agents can transfer their trust from where it exists to where it is needed, e.g., if you initially trust the authenticity of a public key and you verify a message signed by the corresponding private key, then you will also trust the authenticity of the message [18]. Our trust reasoning process enables agents to achieve trust transfer correctly because it includes trust relationships with trust properties. Various forms of trust transfer occur in PKI. Since these certificates and PKI’s do not create trust, they just propagate it [18]. Therefore, first agents must trust something. We can call it initial trust.

One of the advantages of our trust reasoning process based on reciprocal logic is that it provides us with trust relationships and their properties and these trust relationships can be regarded as initial trust. In our PKI Scenario, trust relationship between agent e_1 and its parent entity $TR(e_1, parent(e_1), validity)$ is considered as an initial trust. Therefore, based on the initial trust agent e_1 believes that certificate c_1 informed by its parent entity is valid. Moreover, agent e_1 trust in completeness of agent e_2 $TR(e_1, e_2, completeness)$ but at this point only agent e_2 believes that certificate c_2 is valid and agent e_1 need to know whether the informed certificate c_2 is valid or not. Therefore, through initial trust and other known trust relationships, agents can reason out the desired beliefs by correctly achieving trust transfer through our trust reasoning process.

4.4 Discussion

In PKI (Public key infrastructure), trust relationships play an important role, especially in cases when an agent wants to know whether the certificate informed by another agent is valid or not. Usually, authors have focused on certification relationships [12,19] instead of trust relationships. Our trust reasoning process focuses on two trust relationships, i.e., validity, and completeness. Why are these two trust relationships essential? Because if an agent trusts another agent in its validity, it means that the agent believes that the other agent is a valid information source about both p and $\neg p$. In PKI, every agent has trust in the validity of its parent entity and an agent believes that the information provided by its parent entity, whether p and $\neg p$, is valid. Also, if an agent in PKI has complete information, e.g., about a certificate the agent should inform another agent about that certificate.

Such trust properties are essential in a trust relationship, especially when an agent deals with a message from another agent. Traditional reciprocal logic only deals with the trust relationships between agents. Trust reasoning without these trust properties does not provide us with room to deal with messages from other agents. For such a purpose, a new extension of reciprocal logic is introduced with two new modal operators *Bel* and *Inf* and axioms. Thus, our new extension is $RcL \cup \{ERcL1, \dots, ERcL6, BEL\}$.

Moreover, traditional reciprocal logic does not help us deal with complex situations, for example, when agents have trust relationships based on trust properties. $TR(pe_1, pe_3, validity)$ cannot be concluded from $TR(pe_1, pe_2, validity)$ and $TR(pe_2, pe_3, validity)$ because trust relationships with trust properties are not transitive. Also, some studies have discussed reasons why trust is not transitive [17]. However if we consider the trust transitive as pseudo-transitivity [11], i.e., if all agents in a trust relationship have similar trust properties for the same proposition p then we can say that agent pe_1 trust agent pe_3 in its validity, i.e., $\neg(\forall pe_1 \forall pe_2 \forall pe_3 (TR(pe_1, pe_2, validity) \wedge TR(pe_2, pe_3, validity) \Rightarrow TR(pe_1, pe_3, validity)))$. Therefore, we can conclude that in a case where there are three agents pe_1 , pe_2 , and pe_3 , and two of them, pe_1 and pe_3 , do not have a trust relationship. Thus, based on pseudo transitivity, a trust relationship may be derived if agents pe_1 and pe_2 , as well as pe_2 and pe_3 , have trust relationships with the same trust property and hold the same belief.

Alongside, there are cases when agents can have different trust relationships with different trust properties and agents believe in different propositions. This refers to trust transfer. We have already discussed in Sect. 4.3 how agents can reason out desired beliefs by correctly achieving trust transfer through our trust reasoning process. In the scope of the current paper, trust relationships in the current PKI scenario focus on validity and completeness trust property only. Because in the domain of PKI, validity and completeness are one of the essential properties when dealing with messages from other agents. Also, not all the axioms have been used in the current scope of this paper, but future studies include the application of a trust reasoning process based on these axioms in scenarios complex PKI scenarios and other areas.

We know that trust relationships change themselves over space and time, e.g., at time t if agent α believe p coming from β and at time $t + 1$ agent α believe $\neg p$ coming from β . This could cause complexity for agent α when trusting agent β . This problem provides us a new insight into maintaining and updating trust relationships an agent has with other agents as an agent view. Through a trust relationship, one can capture the beliefs of the agent about the message from other agents at a specific time or in a particular space. These captured beliefs can be added to agents view whom the agent would trust. These agent views containing trust relationships need to be maintained because two different agents may unequally trust any received message and may act differently. Also, these view needs to be updated when an agent makes new trust relationships with other agents. Maintaining and updating views will not only help deal with the future threats but also aid in making a decision. Further research is needed to establish such agents view in new extensions of reciprocal logic.

5 Concluding Remarks

We have proposed a new extension of reciprocal logic that can deal with trust properties. Two modal operators *Bel* and *Inf* have been introduced to represent trust relationships between agents and messages from other agents. A case study has been shown in PKI. Modal operators and new axioms aid in reasoning out new trust relationships in PKI. This we believe is an improvement our new extension. One of the advantages of our approach is generality. Trust reasoning based on a new extension of reciprocal logic is general in terms that not only trust relationships in PKI could be described as an empirical theory but also trust relationships in various complex scenarios can be described.

In the future, the aim is to provide a trust reasoning framework based on a new extension of reciprocal logic and its implementation in various areas and the ideas contained in this paper. Moreover, dealing with trust relationships with time-related constraints is also part of future works.

References

1. Amgoud, L., Demolombe, R.: An argumentation-based approach for reasoning about trust in information sources. *Argument Comput.* **5**(2–3), 191–215 (2014). <https://doi.org/10.1080/19462166.2014.881417>
2. Anderson, A.R., Belnap Jr., N.D.: *Entailment: The Logic of Relevance and Necessity*, vol. 1. Princeton University Press, Princeton (1975)
3. Anderson, A.R., Belnap Jr., N.D., Dunn, J.M.: *Entailment: The Logic of Relevance and Necessity*, vol. 2. Princeton University Press, Princeton (1992)
4. Basit, S., Goto, Y.: An extension of reciprocal logics for trust reasoning. In: Nguyen, N.T., Jearanaitanakij, K., Selamat, A., Trawiński, B., Chittayasothorn, S. (eds.) *ACIHDS 2020. LNCS (LNAI)*, vol. 12034, pp. 65–75. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-42058-1_6

5. Cheng, J.: A strong relevant logic model of epistemic processes in scientific discovery. In Kawaguchi, E., Kangassalo, H., Jaakkola, H., Hamid, I.A. (eds.) *Information Modeling and Knowledge Bases XI. Frontiers in Artificial Intelligence and Applications*, vol. 61, pp. 136–159. Amsterdam, IOS Press (2000)
6. Cheng, J.: Reciprocal logic: logics for specifying, verifying, and reasoning about reciprocal relationships. In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) *KES 2005. LNCS (LNAI)*, vol. 3682, pp. 437–445. Springer, Heidelberg (2005). https://doi.org/10.1007/11552451_58
7. Cheng, J.: Strong relevant logic as the universal basis of various applied logics for knowledge representation and reasoning. In Kiyoki, Y., Henno, J., Jaakkola, H., Kangassalo, H. (eds.) *Information Modeling and Knowledge Bases XVII. Frontiers in Artificial Intelligence and Applications*, vol. 136, pp. 310–320. IOS Press (2006)
8. Demolombe, R.: Reasoning about trust: a formal logical framework. In: Jensen, C., Poslad, S., Dimitrakos, T. (eds.) *iTrust 2004. LNCS*, vol. 2995, pp. 291–303. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24747-0_22
9. Josang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. *Decision Support Syst.* **43**(2), 618–644 (2007). <https://doi.org/10.1016/j.dss.2005.05.019>
10. Koutrouli, E., Tsalgatidou, A.: Credibility enhanced reputation mechanism for distributed e-communities, In: *Proceedings of the 2011 19th International Euromicro Conference on Parallel, Distributed and Network-Based Processing*, pp. 627–634. IEEE Computer Society, Washington (2011). <https://doi.org/10.1109/PDP.2011.68>
11. Leturc, C., Bonnet, G.: A normal modal logic for trust in the sincerity. In: *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, Stockholm, Sweden, pp. 175–183. International Foundation for Autonomous Agents and Multiagent Systems, Richland (2018)
12. Liu, C., Ozols, M., Cant, T.: An axiomatic basis for reasoning about trust in PKIs. In: Varadharajan, V., Mu, Y. (eds.) *ACISP 2001. LNCS*, vol. 2119, pp. 274–291. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-47719-5_23
13. Liao, C.: Belief, information acquisition, and trust in multi-agent systems: a modal logic formulation. *Artif. Intell.* **149**(1), 31–60 (2003). [https://doi.org/10.1016/S0004-3702\(03\)00063-8](https://doi.org/10.1016/S0004-3702(03)00063-8)
14. Namiluko, C.: An architectural approach for reasoning about trust properties. Ph.D. thesis, University of Oxford (2016)
15. Yan, Z., Zhang, P., Vasilakos, A.: A survey on trust management for Internet of Things. *J. Netw. Comput. Appl.* **42**, 120–134 (2014)
16. Zhao, H., Li, X.: A group trust management system for peer-to-peer desktop grid. *J. Comput. Sci. Technol.* **24**(5), 833–843 (2009). <https://doi.org/10.1007/s11390-009-9275-7>
17. Christianson, B., Harbison, W.S.: Why isn't trust transitive? In: Lomas, M. (ed.) *Security Protocols 1996. LNCS*, vol. 1189, pp. 171–176. Springer, Heidelberg (1997). https://doi.org/10.1007/3-540-62494-5_16
18. Jøsang, A., Pedersen, I.G., Povey, D.: PKI seeks a trusting relationship. In: Dawson, E.P., Clark, A., Boyd, C. (eds.) *ACISP 2000. LNCS*, vol. 1841, pp. 191–205. Springer, Heidelberg (2000). https://doi.org/10.1007/10718964_16
19. El Bakkali, H., Kaitouni, B.I. A logic-based reasoning about PKI trust model. In: *Proceedings of Sixth IEEE Symposium on Computers and Communications*, pp. 42–48 (2001). <https://doi.org/10.1109/ISCC.2001.935353>