Fairness in Non-Repudiation Protocols

Wojciech Jamroga, Sjouke Mauw, and Matthijs Melissen

Computer Science and Communication
University of Luxembourg

Abstract. We indicate two problems with the specifications of fairness that are currently used for the verification of non-repudiation and other fair-exchange protocols. The first of these problems is the implicit assumption of perfect information. The second problem is the possible lack of effectiveness. We solve both problems in isolation by giving new definitions of fairness, but leave the combined solution for further work. Moreover, we establish a hierarchy of various definitions of fairness, and indicate the consequences for existing work.

Keywords: Security protocols, verification, non-repudiation and fair exchange protocols, alternating-time temporal logic, imperfect information.

1 Introduction

The correctness of a security protocol depends in general on the precise formulation of its security requirements. Consequently, the development of appropriate security requirements is at least as important as the proper design of security protocols. Classical requirements, such as *confidentiality* and *authentication*, are well understood and have been exhaustively investigated [1–3]. Research on more recent requirements, such as *receipt-freeness* in electronic voting protocols [4, 5], seems to converge, while for other properties, such as *ownership transfer* in RFID protocols, discussions have only recently started [6].

In this paper, we study the development of the requirement of *fairness* for *non-repudiation protocols*. The main goal of a non-repudiation protocol is to allow two (or more) parties to exchange goods or messages without any of the parties being able to falsely deny having taken part in the exchange. Such a protocol is designed so that the sender of the message obtains a *non-repudiation of receipt* (NRR) evidence and the receiver of the message a *non-repudiation of origin* (NRO) evidence. The main security requirement is *fairness*, which roughly states that if the receiver obtains NRO, then the sender can obtain NRR, and vice versa. An example of a non-repudiation protocol is a *certified e-mail* protocol [7].

Although other requirements, such as *abuse-freeness*, also apply to non-repudiation protocols (and the wider class of *fair-exchange protocols*), we will only investigate fairness and its relation to *effectiveness* and *strategic timeliness*. Effectiveness (sometimes also called *viability*) is not a security requirement, but a functional requirement, stating that the protocol can actually achieve the exchange of an NRR and an NRO evidence. Strategic timeliness requires that an agent always has an honest strategy to stop execution of the protocol.

In the literature on non-repudiation protocols, a variety of different interpretations of the fairness requirement have been described. Most of these were formalized in the modal logic ATL [8] as to allow for the automated verification of protocols through model checking, for example in the Mocha model checker [9]. The observed variations seem to be due to differences in the assumed execution models of the agents involved, to differences in the adversary model, and to differences in the intended application of the protocol. Some authors already provided insight in the relation between some of the fairness definitions [10].

Nevertheless, we observe two limitations of the existing definitions. The first concerns the implicit assumption of *perfect information*, as it is called in game theory. By this we mean that, at each moment, all agents have full knowledge of the global state of the system. In practice this does not seem a realistic assumption for a security protocol. One would expect an agent to only know his own state and use a protocol to infer knowledge of the other agents' states. This assumption has a significant impact on the formulation of fairness in ATL.

The second limitation concerns the combination of fairness and effectiveness. In the game-theoretical setting, both properties are expressed in terms of the existence of strategies. By taking the conjunction of the two properties, one does not necessarily obtain a single strategy that enforces both fairness and effectiveness. Here, we propose a new property which blends fairness and effectiveness properly.

The contribution of this paper is as follows. *(i)* We revisit existing notions of fairness (Sec. 3.1). *(ii)* We introduce a notion of fairness based on the assumption of imperfect information (Sec. 3.2). *(iii)* We combine fairness and effectiveness (Sec. 3.3). *(iv)* We develop the hierarchy of fairness requirements and prove correctness and strictness of the inclusions (Sec. 4). *(v)* We consider implications for the practical use of various notions of fairness in the literature (Sec. 5). These contributions are preceded by a short introduction to non-repudiation protocols and an overview of the logic ATL (Sec. 2).

2 Preliminaries

2.1 Non-Repudiation Protocols

Non-repudiation guarantees that an agent cannot deny having taken part in a message exchange, if it has actually done so in the course of the protocol [11]. To achieve this, protocol participants usually collect evidences that can later be presented to a judge. If Alice sends a message m to Bob, we can distinguish a *non-repudiation of origin* (NRO) evidence, which proves that Alice cannot deny having sent m, and a *non-repudiation of receipt* (NRR) evidence, which proves that Bob cannot deny having received m. Both Alice and Bob have an incentive to cheat. This means that, e.g., Bob may try to obtain m without providing an NRR. Evidences are typically implemented with cryptographic signatures over the message (and possibly some other data).

It is often desirable to have the guarantee of fair exchange [12] of non-repudiation. For example, when Alice sends message m to Bob, it should hold that Alice receives NRR if and only if Bob receives NRO.

Fairness cannot be ensured without at least one external agent, which is trusted by both parties, and is called a Trusted Third Party (TTP) [13]. The TTP can be *inline*, *online* or *offline*. An inline TTP handles the items to be exchanged. An online TTP does *not* handle the items to be exchanged, but is necessary in each invocation of the main protocol. An offline TTP is only invoked in dispute resolution.

The communication channels between the TTP and the other agents are assumed to be *resilient*, i.e. all data is delivered after a finite, but unknown amount of time. The communication channels between the other agents are assumed to be *unreliable*, i.e. data may be lost. We assume a standard Dolev-Yao attacker who has full control over the unreliable channels of the network and who may co-operate with any of the possibly dishonest parties to disrupt the protocol.

In this paper, we assume that all messages and evidences that are being transmitted are labeled with the type of the message or evidence, the name of the sender, the name of the intended recipient, the name of the TTP which is agreed upon, and an identifier linking the message to the protocol session. Further, as to focus on non-repudiation, we assume in the example protocols that all exchanged messages are cryptographically protected, thereby preventing possible attacks on confidentiality and authenticity of the exchanged messages.

Protocol 1

```
1. $A \to B$: $m, \mathsf{NRO}$
2. $B \to A$: $\mathsf{NRR}$
```

Protocol 2

```
1. $A \to T$: $m$, $\mathsf{NRO}$
2. $T \to B$: $m$
3. $B \to T$: $\mathsf{NRR}$
4. $T \to B$: $\mathsf{NRO}$
5. $T \to A$: $\mathsf{NRR}$
```

Protocol 1 is an example of a simple non-repudiation protocol, where Alice and Bob exchange non-repudiation of origin and receipt of message m. The protocol specifies that first Alice sends message m and NRO to Bob, and then Bob sends NRR to Alice. Here, NRO could be implemented as $[f_{\mathsf{NRO}}, A, B, m]_A$ and NRR as $[f_{\mathsf{NRR}}, A, B, m]_B$, where $[M]_C$ is the signature of agent C over message M, and f_{NRR} and f_{NRO} are flags indicating the type of the evidence. Note that this protocol is not fair, as Bob can abort after step 1, leaving Alice without NRR. Protocol 2 is an example of a fair NR-protocol (with inline TTP). Fairness is intuitively guaranteed because the TTP will not send out NRO and NRR before he has collected both evidences.

Non-repudiation protocols with inline TTP are generally inefficient, as the TTP becomes easily a bottleneck. Protocols with offline TTP do not suffer from this problem, but also tend to be more complex, as they typically comtain non-determinism and various sub-protocols. This means that it is less easy to check by hand that fairness is satisfied.. Therefore, a formal way of verifying fairness is needed.

2.2 Alternating-time Temporal Logic

We use alternating-time temporal logic (ATL) [8] to specify requirements of fair exchange. ATL is very suitable for specification of security protocols, because it allows to express that there exists *a strategy* with which an agent obtains a desired property, instead of requiring that all protocol runs have to satisfy the property, independent of the agent's behavior. We only give a brief introduction to ATL; we refer to [8] for the full definition.

An ATL formula is one of the following:

- p, for propositions $p \in \mathit{\Pi}$
- $\neg\varphi$ or $\varphi_1 \vee \varphi_2$, where φ, φ_1 and φ_2 are ATL formulas.
- $\langle\langle A\rangle\rangle\bigcirc\varphi$, $\langle\langle A\rangle\rangle\Box\varphi$ or $\langle\langle A\rangle\rangle\varphi_1\ \mathcal{U}\ \varphi_2$, where $A \subseteq \mathit{\Sigma}$ is a set of agents, and φ, φ_1 and φ_2 are ATL formulas.

The *strategic operator* $\langle\langle\mathcal{A}\rangle\rangle$ can be seen as a path quantifier that ranges over all paths that the agents in $\mathcal{A}$ can force the game into, irrespective of how the other agents proceed. Furthermore, $\bigcirc$ ("next"), $\Box$ ("always") and $\mathcal{U}$ ("until") are *temporal operators.* Sometimes we write $\langle\langle a_1, \ldots, a_n\rangle\rangle$ instead of $\langle\langle\{a_1, \ldots, a_n\}\rangle\rangle$. Additional Boolean connectives are defined in the usual manner. We also define $\Diamond$ ("eventually") as $\Diamond\varphi \equiv \mathsf{true}\ \mathcal{U}\ \varphi$.

ATL formulas are interpreted in a *concurrent games structure* (*CGS*), which is a tuple $S = \langle \mathbb{A}\mathrm{gt}, \mathit{Act}, Q, \mathit{\Pi}, \pi, d, \delta\rangle$ with the following components: a finite set $\mathbb{A}\mathrm{gt}$ of *agents*; a finite set Q of *states*; a finite set $\mathit{\Pi}$ of *propositions*; for each state $q \in Q$, a set $\pi(q) \subseteq \mathit{\Pi}$ of propositions true at q; for each agent $A \in \mathbb{A}\mathrm{gt}$ and each state $q \in Q$, a set $d_A(q) \subseteq \mathit{Act}$ of actions available at state $q \in Q$ to agent $A \in \mathbb{A}\mathrm{gt}$; and a transition function δ that assigns a new state $\delta(q, j_1, \ldots, j_k) \in Q$ to every combination of state q and actions $j_1, \ldots, j_k$, one per agent in $\mathbb{A}\mathrm{gt}$.

A *path* in S is an infinite sequence $\lambda = q_0, q_1, q_2, \ldots$ of states such that for all positions $i \geq 0$, we have $q_{i+1} = \delta(q, j_1, \ldots, j_k)$ for some actions $j_1, \ldots, j_k$. We refer to a path starting at state q as a *q-path.* For a path λ and a position $i \geq 0$, we use $\lambda[i]$ and $\lambda[0, i]$ to denote the i-th state of λ and the finite prefix $q_0, q_1, \ldots, q_i$ of λ, respectively. A *strategy* f_A for agent A determines, for every finite sequence of states s, an action $f_A(s)$ for agent A. A collective strategy $F_{\mathcal{A}}$ is simply a tuple of strategies f_A, one for each agent $A \in \mathcal{A}$. We define the *outcome* of $F_{\mathcal{A}}$ from $q \in Q$ as the set $\mathit{out}(q, F_{\mathcal{A}})$ of q-paths that the agents in $\mathcal{A}$ enforce when executing $F_{\mathcal{A}}$. The semantics of ATL is defined as follows:

- $S, q \models p$ for propositions $p \in \mathit{\Pi}$, iff $p \in \pi(q)$.
- $S, q \models \neg\varphi$ iff $S, q \not\models \varphi$.
- $S, q \models \varphi_1 \vee \varphi_2$ iff $S, q \models \varphi_1$ or $S, q \models \varphi_2$.
- $S, q \models \langle\langle\mathcal{A}\rangle\rangle\bigcirc\varphi$ iff there exists a collective strategy $F_{\mathcal{A}}$ such that for all paths $\lambda \in \mathit{out}(q, F_{\mathcal{A}})$, we have $S, \lambda[1] \models \varphi$.

- $S, q \models \langle\langle \mathcal{A} \rangle\rangle \Box \varphi$ iff there exists $F_{\mathcal{A}}$ such that for all $\lambda \in out(q, F_{\mathcal{A}})$ and all positions $i \geq 0$, we have $S, \lambda[i] \models \varphi$.
- $S, q \models \langle\langle \mathcal{A} \rangle\rangle \varphi_1 \,\mathcal{U}\, \varphi_2$ iff there exists $F_{\mathcal{A}}$ such that for all $\lambda \in out(q, F_{\mathcal{A}})$ there exists $i \geq 0$ with $S, \lambda[i] \models \varphi_2$ and for all $0 \leq j < i$ we have $S, \lambda[j] \models \varphi_1$.

The universal path quantifier of the branching-time temporal logic CTL can be captured in ATL as $\forall \equiv \langle\langle \emptyset \rangle\rangle$. The existential path quantifier $\exists$ will be interpreted as usual in CTL. The expressiveness of ATL can be illustrated by the following examples. The formula $\neg\langle\langle A \rangle\rangle \Diamond \varphi$ means that A does not have a strategy to ever obtain φ. The formula $\forall \Box (\langle\langle B \rangle\rangle \Box \neg \varphi \vee \exists \Diamond \psi)$ means that in every reachable state, either B has a strategy that always avoids φ, or there exists a path that eventually results in a state where ψ holds.

The following properties will be used later. Proofs are straightforward.

Fact 1 *$q \models \neg \exists \varphi$ implies $q \models \neg \langle\langle \mathbb{A}\mathrm{gt} \rangle\rangle \varphi$.*

Fact 2 *$S, q \models \langle\langle A \rangle\rangle (\varphi \,\mathcal{U}\, \psi)$ implies $S, q \models \langle\langle A \rangle\rangle \Diamond \psi$.*

3 Capturing Fairness of Exchange in ATL

Various ATL definitions of fairness have been proposed in the literature on non-repudiation protocols and other fair exchange protocols. In this section, we give an overview of the proposed definitions. Then, we have a look at two fundamental problems with the existing formalizations and propose how they can be repaired.

3.1 Existing Formalizations

When Alice sends a message to Bob, one can distinguish *fairness for Alice* (whenever Bob receives NRO, Alice is guaranteed to receive NRR), and *fairness for Bob* (whenever Alice receives NRR, Bob is guaranteed to receive NRO). We only consider fairness for Alice; fairness for Bob can be formulated symmetrically.

Fairness for an agent only needs to be guaranteed when the agent complies with the protocol: if an agent does not follow the protocol, he does that at his own risk. An agent that complies with the protocol is called *honest*. Fairness should be guaranteed for honest agents even if the other agents are *dishonest*, i.e., behave in a way that is not foreseen by the protocol. Therefore, when studying fairness for Alice, we assume that Alice is honest and that Bob might be dishonest. We do not require recovery of fairness after unintended dishonest behavior caused by system failures, as has been considered in [14, 15].

To check fairness of a protocol using ATL, the protocol is modeled as a concurrent game structure [10]. We set agents $\mathbb{A}\mathrm{gt} = \{A_h, B, T\}$, where A stands for Alice, B stands for Bob, X_h signifies that agent X is restricted to honest behavior, and T stands for the TTP (which is always honest). Furthermore we set propositions $\Pi = \{\mathsf{NRO}, \mathsf{NRR}\}$. The proposition NRO is true in these states where Bob possesses non-repudiation of origin, and the proposition NRR is true in these states where Alice possesses non-repudiation of receipt. We assume that the model is *turn-based* (i.e., agents do not act simultaneously), and that the behavior of the TTP is deterministic (given the current state of the system). We do not model the communication channel explicitly to simplify the notation and avoid the necessity to formalize channel resilience, which cannot be done in "pure" ATL.

Strong Fairness One of the definitions of fairness proposed by Kremer and Raskin [16] is *strong fairness.* It can be formulated as follows:

$$\textsc{StrongFair} \equiv \forall\Box(\mathsf{NRO} \to \forall\Diamond\mathsf{NRR})$$

Strong fairness for Alice states that in every reachable state where Bob has NRO, Alice should eventually obtain NRR, whatever the agents do. Strong fairness can be seen as *enforced fairness*: if due to underspecification the protocol is non-deterministic and thus gives Alice multiple available strategies, each of these strategies should guarantee her NRR.

Non-enforced Fairness If we assume that Alice is rational, $\textsc{StrongFair}$ is stronger than necessary. A weaker form of fairness, which requires Alice to play rational, has also been proposed by Kremer and Raskin [16] through the following ATL formula:

$$\text{NE}\textsc{Fair} \equiv \neg\langle\langle B\rangle\rangle\Diamond(\mathsf{NRO} \wedge \neg\langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR})$$

This formula states that Bob should not have a strategy to reach a state where he has NRO while Alice at the same time does not have a strategy to obtain NRR. We will call this notion *non-enforced fairness*, because a protocol that satisfies this requirement does not *enforce* fairness: if Alice has multiple strategies, one "good" strategy is sufficient; the other strategies might still result in an unfair situation.

Strategic Fairness An intermediate notion of fairness, called *strategic fairness*, has been proposed by Chadha et al. [10].

$$\textsc{StratFair} \equiv \forall\Box(\mathsf{NRO} \to \langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR})$$

A protocol satisfies *strategic fairness* for Alice if and only if in every reachable state, it holds that whenever Bob has received NRO, there exists a strategy for honest Alice that gives her NRR.

It seems to us, however, that this definition is counterintuitive, as it combines the enforced and non-enforced approach. If one assumes that Alice has enough rationality to resolve non-determinism in the correct way, then it is not necessary to require that she obtains the fair situation $\mathsf{NRO} \to \langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR}$ independently of her strategy; it would suffice if there *exists* a strategy for Alice that guarantees the fair situation. On the other hand, if one does not assume that Alice is able to resolve non-determinism in the correct way, then it is not enough to require that there *exists* a strategy that gives her NRR; she might still never receive NRR when she never plays the right strategy. Therefore, strategic fairness is too strong for rational agents, and too weak for agents without rationality.

Weak Fairness Another definition of fairness, proposed by Chadha et al. [10] to simplify verification, is *weak fairness*.

$$\textsc{WeakFair} \equiv \forall\Box(\mathsf{NRO} \rightarrow \exists\Diamond\mathsf{NRR})$$

A protocol satisfies weak fairness for Alice if and only if in every reachable state, it holds that whenever Bob has received NRO, if all agents cooperate, Alice will eventually get NRR.

Invariant Fairness One disadvantage with the above formulations of fairness is that counterexamples cannot always be expressed as single paths. An alternative definition of fairness is proposed based on invariants. *Invariant fairness* [10] for Alice only tests those states in which Alice has stopped the protocol, allowing counterexamples to be expressed as traces. We define the proposition Stop_A to be true exactly when Alice has stopped executing the protocol. It is assumed that as soon as Alice has stopped executing the protocol, she cannot receive NRR anymore, i.e., $\forall\Box((\mathrm{Stop}_A \wedge \neg\mathsf{NRR}) \rightarrow \forall\Box\neg\mathsf{NRR})$. Now invariant fairness is defined as follows:

$$\textsc{InvFair} \equiv \forall\Box(\mathrm{Stop}_A \rightarrow (\mathsf{NRO} \rightarrow \mathsf{NRR}))$$

This formula states that in all states where Alice has stopped executing the protocol, Alice should possess NRR whenever Bob possesses NRO.

3.2 Fair Exchange and Imperfect Information

ATL formulas are normally evaluated in a model that assumes *perfect information*, that is, agents are assumed to know precisely the current global state of the system, including the local states of the other agents [8]. This is also the way in which Mocha evaluates ATL formulas. This assumption is unrealistic for communication protocols: if all agents knew the local state of all other agents, no communication would be needed. We will look at NEFair, and see that assuming perfect information, as is done in [16], leads to counterintuitive results.

A perfect information strategy for Alice can be *non-executable* under imperfect information: the strategy might require executing different actions in situations that look the same to Alice. Furthermore, even if she has an executable strategy, she may be unaware of having it, and unable to identify it [17]. For example, one can construct a protocol in which the message that Alice needs to send depends on whether Bob did or did not receive some other message. Alice does not know which messages have been received by Bob, so although she has a strategy to send the right message if she had perfect information, she is not able to follow this strategy under imperfect information.

An example of this is Protocol 3, in which Alice sends message m to Bob, and NRO and NRR are exchanged. First, Alice sends m and NRO to the TTP. The TTP forwards m to Bob, who replies by sending NRR and a boolean p back to the TTP. Then the TTP sends NRO to Bob. Alice continues by sending a boolean p' to the TTP. Only if Bob's boolean p equals Alice's boolean p', the TTP sends NRR to Alice.

Protocol 3

1. $A \to T$: m, NRO
2. $T \to B$: m
3. $B \to T$: NRR, bool p
4. $T \to B$: NRO
5. $A \to T$: bool p'
6. If $p = p'$:
   - (a) $T \to A$: NRR

Protocol 4

1. $T \to A$: `start`
2. $T \to B$: `start`
3. Choose between:
   - (a) 1. $A \to T$: NRO, `request_id`
     2. $B \to T$: NRR, id
     3. $T \to B$: NRO
     4. $A \to T$: `re-request_id`
   - (b) 1. $B \to T$: NRR, id
     2. $A \to T$: NRO, `request_id`
     3. $T \to B$: NRO
4. $T \to A$: NRR, id

Intuitively, Protocol 3 is not a fair protocol, as Alice can only obtain NRR by sending p' in step 5 such that p' equals p. However, she does not have a way of knowing p, and therefore does not know the correct value of p'. Nevertheless, the protocol satisfies NEFair, as $\langle\!\langle A_h \rangle\!\rangle \Diamond \mathsf{NRR}$ is true in step 5, since Alice has a correct (perfect information) strategy: if $p = \mathsf{false}$, she sends false, and if $p = \mathsf{true}$, she sends true. The problem is that this strategy is not executable if Alice has imperfect information.

In the previous example, it is immediately obvious that Alice's lack of uinformation causes the protocol to be broken. Protocol 4 is a less contrived example (to simplify the presentation, it is assumed that the TTP stops sending messages to agents from which he receives messages that do not correspond to the protocol). Here, the non-determinism is caused by the order of arrival of messages, instead of by a boolean chosen by the other agent. In this protocol, first the TTP sends the message `start` to Alice and Bob. Then Alice sends NRO and a message `request_id` to request Bob's id to the TTP, and Bob sends NRR and his id to the TTP. However, the behavior of the TTP depends on the order in which these messages arrive. If the request arrives before the id, as in branch (a), the TTP sends NRO to Bob, but Alice's request is ignored until Alice sends an additional message `re-request_id`, on which the TTP sends her the id and NRR. If the request arrives after the id, as in branch (b), the TTP sends NRO to Bob and, immediately, NRR and the id to Alice.

This implies that Alice will never receive NRR in case she does not send `re-request_id` in branch (a). On the other hand, in branch (b) Alice will never receive NRR if she does send `re-request_id`. Alice cannot know or make sure that `request_id` arrives before or after Bob's id, and neither does she know how long the TTP will wait before answering her. Therefore, Alice does not know which branch of the protocol is executed by the TTP, which means that she does not know whether she needs to send `request_id` or not. Still, this protocol satisfies NEFair, as Alice has a perfect information strategy to obtain NRR, namely sending `re-request_id` in branch (a) and not sending it in (b).

The problem can be solved by interpreting specifications in *ATL with imperfect information* [18], where agents can only observe a part of the global state, and their strategy is required to choose the same action in states they cannot distinguish. That version of ATL is interpreted in an *imperfect information concurrent game structure* (*iCGS*)), which is a concurrent game structure extended with an *indistinguishability relation* $\sim_A$ for every agent $A \in \mathbb{A}\mathrm{gt}$. Strategies are required to be *uniform*, that is, if sequences s, s' are indistinguishable for agent A, written $s \sim_A s'$, then the strategy for agent A assigns the same action to s and s', i.e., $f_A(s) = f_A(s')$. Now, the semantics of $\langle\langle \mathcal{A} \rangle\rangle \Box$ is changed as follows: $q \models \langle\langle \mathcal{A} \rangle\rangle \Box \varphi$ if and only if there exists a uniform collective strategy $F_{\mathcal{A}}$ such that for all agents $A \in \mathcal{A}$, states $q' \sim_A q$, paths $\lambda \in out(q', F_{\mathcal{A}})$ and positions $i \geq 0$, we have $\lambda[i] \models \varphi$. The semantics of "next" and "until" are changed in the same way. Note that: (1) the set of uniform strategies in S is always a subset of perfect information strategies in S; (2) perfect information semantics of ATL is well-defined in iCGS (it simply ignores the indistinguishability relations); (3) each CGS can be seen as an iCGS where for every agent a, $\sim_a$ is the minimal reflexive relation.

Imperfect information semantics is sufficient to give an intuitive interpretation to StratFair, WeakFair, StrongFair and InvFair (for the latter too, the choice of semantics only matters if the initial state is unknown). However, it is not enough to "repair" NEFair. If Alice wants to be sure that she can obtain NRR, it is also necessary to use *non-enforced controled fairness* (NECFair) instead of NEFair.

$$\text{NECFair} \equiv \langle\langle A_h \rangle\rangle \Box \neg(\mathsf{NRO} \wedge \neg \langle\langle A_h \rangle\rangle \Diamond \mathsf{NRR})$$

To see the difference between NEFair and NECFair, we define an *unfair situation* (in which Bob has NRO and Alice does not have a strategy to obtain NRR) as $\text{Unfair} \equiv (\mathsf{NRO} \wedge \neg \langle\langle A_h \rangle\rangle \Diamond \mathsf{NRR})$. Then we can write:

$$\text{NEFair} \equiv \neg \langle\langle B \rangle\rangle \Diamond \text{Unfair},$$
$$\text{NECFair} \equiv \langle\langle A_h \rangle\rangle \Box \neg \text{Unfair}.$$

Note also that for models with $\mathbb{A}\mathrm{gt} = \{A, B, T\}$ and deterministic T, the NEFair requirement is equivalent to $\neg \langle\langle \mathbb{A}\mathrm{gt} \backslash A_h \rangle\rangle \Diamond \text{Unfair}$. That is, NEFair requires that all agents but Alice have no common strategy to reach an unfair situation, while NECFair states that Alice has a strategy to always avoid an unfair situation, i.e., Alice is *in control* over the outcome. These two formulas are equivalent assuming perfect information and turn-based models [8]. However, in imperfect information models, both NEFair and the negation of NECFair can hold, as for example in Protocol 3 (on the other hand, NECFair does imply NEFair, even in imperfect information models). Because Protocol 3 is intuitively unfair, we have that under imperfect information, NEFair is not sufficient for Alice to avoid an unfair situation and NECFair should be required.

In some situations, Alice might accept that she cannot avoid an unfair situation, as long as Bob does not have a strategy to bring Alice in an unfair situation. In that case, NEFair, the weaker form of fairness, is sufficient. Consider for example the case where Bob wants to rob Alice's locker by opening the lock with the right code. Bob could be lucky in guessing the right code and therefore Alice has no strategy to avoid an unfair situation. Alice might accept this, as long as Bob does not have a (imperfect information) strategy that guarantees that he opens the locker, and the number of possible codes is sufficiently large.

From now on, we will follow Schobbens [18] and use subscripts I (respectively i) to denote that the specification is interpreted in the perfect (resp. imperfect) information semantics of ATL whenever the type of semantics has impact on the truth of the specification. We will also write that φ_x *implies* ψ_y iff, for every iCGS S and state q in it, we have that $S, q \models_x \varphi$ implies $S, q \models_y \psi$.

Fact 3 *If φ includes no strategic operators then $(\langle\langle\mathcal{A}\rangle\rangle\varphi)_i$ implies $(\langle\langle\mathcal{A}\rangle\rangle\varphi)_I$. The converse does not hold in general.*

3.3 Effective Fairness

Now we show that fairness is not sufficient for a fair-exchange protocol, and discuss an additional requirement, called *effectiveness* (in some papers also *viability*). It turns out that combine these two requirements is not trivial.

To see the need for effectiveness, consider the *empty protocol*, i.e., the (admittedly useless) protocol, that specifies that no message will be sent. It is obvious that this protocol satisfies all definitions of fairness discussed above, as no unfair situation can possibly occur. Still the protocol is clearly not a good fair-exchange protocol, because even if the agents want to, they cannot exchange evidences.

To prevent protocols like this, we need to impose a second requirement (besides fairness), that states that the protocol is *effective*. This means that Alice and Bob have a collective strategy to run the protocol such that both agents obtain their evidence. This requirement can be formulated in ATL as follows:

$$\textsc{Effective} \equiv \langle\langle A_h, B_h\rangle\rangle\Diamond(\mathsf{NRO} \wedge \mathsf{NRR})$$

Requiring effectiveness excludes the empty protocol. However, requiring both effectiveness and non-enforced fairness is not sufficient to rule out bad protocols. To see this, let us consider Protocol 5.

Protocol 5

1. Choice for A:
   (a) 1. $A \rightarrow B$: NRO
       2. $B \rightarrow A$: NRR
   (b) End of protocol.

In this protocol, Alice can choose to either send NRO to Bob and wait for NRR to be sent to her, or immediately stop the protocol.

This protocol is effective (if Alice chooses 1a and both parties continue the protocol, they get their evidence). Furthermore, the protocol satisfies non-enforced fairness, because Alice has a strategy to achieve fairness (by choosing

1b). Thus, the protocol satisfies both (non-enforced) fairness and effectiveness. However, intuitively, it is still not a good protocol, as Bob might be dishonest and stop the protocol after 1(a)1, leaving Alice without her evidence. This problem arises because A's strategy that guarantees effectiveness is different from A's strategy that guarantees fairness. To solve this problem, we need to require that there exists a strategy for Alice that satisfies both effectiveness and fairness at the same time. The following ATL formula accomplishes this:

$$\langle\langle A_h, B_h\rangle\rangle(\text{NE}\textsc{Fair}\ \ \mathcal{U}\ (\mathsf{NRO} \wedge \mathsf{NRR}))$$

This formula expresses that A and B have a collective strategy that guarantees NEFair for Alice until both Bob and Alice have their evidence.

The formula requires that Bob is honest in the outer quantifier, but allows Bob to be dishonest in the quantifier inside NEFair. This is a problem, as agents need to be either modeled as honest or dishonest. Therefore, we introduce an additional proposition Honest_B, which is true as long as Bob has only sent messages allowed by the protocol. Now we can reformulate the requirement for Bob's honesty so that it applies only to effectiveness and not fairness:

$$\textsc{EffFair} \equiv \langle\langle A_h, B\rangle\rangle(\text{NEC}\textsc{Fair}\ \ \mathcal{U}\ (\mathsf{NRO} \wedge \mathsf{NRR} \wedge \text{Honest}_B))$$

Now we show that effective fairness indeed guarantees both effectiveness and non-enforced fairness.

Theorem 1. $\textsc{EffFair}_I$ *implies* $\textsc{Effective}_I$ *and* $\text{NEC}\textsc{Fair}_I$*, and* $\textsc{EffFair}_i$ *implies* $\textsc{Effective}_i$ *and* $\text{NEC}\textsc{Fair}_i$.

Proof. That $\textsc{EffFair}_I$ implies $\textsc{Effective}_I$ follows directly from Fact 2. To prove that $\textsc{EffFair}_I$ implies $\text{NEC}\textsc{Fair}_I$, we show first that $\mathsf{NRO} \wedge \mathsf{NRR}$ implies $\text{NEC}\textsc{Fair}_I$. Assume $S, q \models_I \mathsf{NRO} \wedge \mathsf{NRR}$. Let λ be a q-path and $i \geq 0$. Then we have that $S, \lambda[i] \models_I \mathsf{NRR}$ (as NRR is a property that stays true after it has been true for the first time). Then it holds that $S, \lambda[i] \models_I \langle\langle A_h\rangle\rangle \mathsf{true}\, \mathcal{U}\ \mathsf{NRR}$ and thus $S, \lambda[i] \models_I \langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR}$, and therefore $S, \lambda[i] \not\models_I \mathsf{NRO} \wedge \neg\langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR}$. This implies that $S, q \models_I \forall\Box\neg(\mathsf{NRO}\wedge\neg\langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR})$, and thus $S, q \models_I \langle\langle A\rangle\rangle\Box\neg(\mathsf{NRO}\wedge \neg\langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR}) = \text{NEC}\textsc{Fair}$.

That $\textsc{EffFair}_I$ implies $\textsc{Effective}_I$ follows directly from Fact 2. In order to prove that $\textsc{EffFair}_I$ implies $\text{NE}\textsc{Fair}_I$, we show first that $\mathsf{NRO} \wedge \mathsf{NRR}$ implies $\text{NE}\textsc{Fair}_I$. Assume $S, q \models_I \mathsf{NRO} \wedge \mathsf{NRR}$. Let λ be a q-path and $i \geq 0$. Then we have that $S, \lambda[i] \models_I \mathsf{NRR}$ (as NRR is a property that stays true after it has been true for the first time). Then it holds that $S, \lambda[i] \models_I \langle\langle A_h\rangle\rangle \mathsf{true}\, \mathcal{U}\ \mathsf{NRR}$ and thus $S, \lambda[i] \models_I \langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR}$, and therefore $S, \lambda[i] \not\models_I \mathsf{NRO} \wedge \neg\langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR}$. Therefore, it holds that $S, q \models_I \neg\langle\langle B\rangle\rangle\Diamond(\mathsf{NRO} \wedge \neg\langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR}) = \text{NE}\textsc{Fair}$.

Now assume $\textsc{EffFair}_I$. Then there exists $F = \{F_A, F_B\}$ for A_h, B_h such that for all $\lambda \in out(q, F)$ there is $i \geq 0$ with $S, \lambda[i] \models_I \mathsf{NRO} \wedge \mathsf{NRR}$, and for all $0 \leq j < i$, we have $S, \lambda[j] \models_I \text{NEC}\textsc{Fair}$. If $i = 0$, we have that $S, \lambda[0] \models_I \text{NEC}\textsc{Fair}$ as $\mathsf{NRO} \wedge \mathsf{NRR}$ implies $\text{NEC}\textsc{Fair}_I$. Otherwise, we have $S, \lambda[j] \models_I \text{NEC}\textsc{Fair}$ directly.

The proof for imperfect information is analogous.

Note that the converse implications do not hold. For example, Protocol 5 satisfies both $\textsc{Effective}$ and NE$\textsc{Fair}$, but not $\textsc{EffFair}$.

We observe that $\textsc{EffFair}_I$ suffers from the problems concerning imperfect information mentioned in Sec. 3.2. Moreover, even if a protocol satisfies $\textsc{EffFair}_i$, it can still be the case that the strategies for Alice behind the outer and the nested strategic operators cannot be combined into a single uniform strategy (cf. [19]). Consider the situation where Alice can either stop, resulting in fairness but not effectiveness, or continue, only resulting in fairness (and effectiveness) if Bob plays honest and neither fairness nor effectiveness otherwise. This is problematic if Alice does not know whether Bob plays honest: in that case, $\textsc{EffFair}_I$ is satisfied, but Alice does not have a strategy that results in both fairness and effectiveness.

We have shown that fairness is not sufficient for fair-exchange protocols, and that effectiveness is also needed. Moreover, non-enforced fairness and effectiveness cannot be combined trivially. We give a new specification, $\textsc{EffFair}$, that handles this combination. This problem does not occur for weak, strategic, strong or invariant fairness and effectiveness. For these specifications, it is sufficient to require the conjunction of fairness and effectiveness.

4 Hierarchy of Fairness Requirements

We proceed by studying the relations between the different definitions of fairness. Fig. 1 contains a graphical view of these relations. Below we include proof sketches for some of the relations. The other cases are relatively straightforward. Unless explicitly stated otherwise, the same reasoning applies to both semantic variants of ATL.

Strong, Strategic, Weak and Invariant Fairness Chadha et. al [10] prove that

$$\textsc{StrongFair}_I \Rightarrow \ \textsc{StratFair}_I \Rightarrow \ \textsc{WeakFair}_I \Rightarrow \ \textsc{InvFair}_I.$$

The latter three implications extend to imperfect information. Furthermore, they show that $\textsc{StratFair}_I$, $\textsc{WeakFair}_I$ and $\textsc{InvFair}_I$ are equivalent under *strategic timeliness*. Strategic timeliness states that Alice always has an honest strategy that eventually allows her to stop executing the protocol: $\textsc{Timely} \equiv \forall\Box(\langle\langle A_h\rangle\rangle\Diamond\text{Stop}_A)$. Furthermore, $\textsc{InvFair}_I$, $\textsc{WeakFair}_I$ and $\textsc{StrongFair}_I$ are clearly equivalent with $\textsc{InvFair}_i$, $\textsc{WeakFair}_i$ and $\textsc{StrongFair}_i$, respectively, as they do not contain strategic modalities.

These are the only implications that hold between $\textsc{StrongFair}$, $\textsc{StratFair}$, $\textsc{WeakFair}$ and $\textsc{InvFair}$. We show this by providing a number of counterexamples, see Fig. 2. Protocol 6 satisfies $\textsc{StratFair}$, but not $\textsc{StrongFair}$. Protocol 7 (a protocol lacking strategic timeliness) satisfies $\textsc{WeakFair}$ but not $\textsc{StratFair}$. Protocol 8 (another protocol lacking strategic timeliness) satisfies $\textsc{InvFair}$, but not $\textsc{WeakFair}$. Finally, $\textsc{StrongFair}_i \Rightarrow \textsc{StratFair}_i$ and $\textsc{WeakFair}_i \Rightarrow \textsc{StratFair}_i$ are not valid, even under strategic timeliness, as they do not hold in a model where the initial state with $\neg\mathsf{NRO}$ is indistinguishable from an unreachable state with $\mathsf{NRO} \wedge \neg\mathsf{NRR}$ holds.

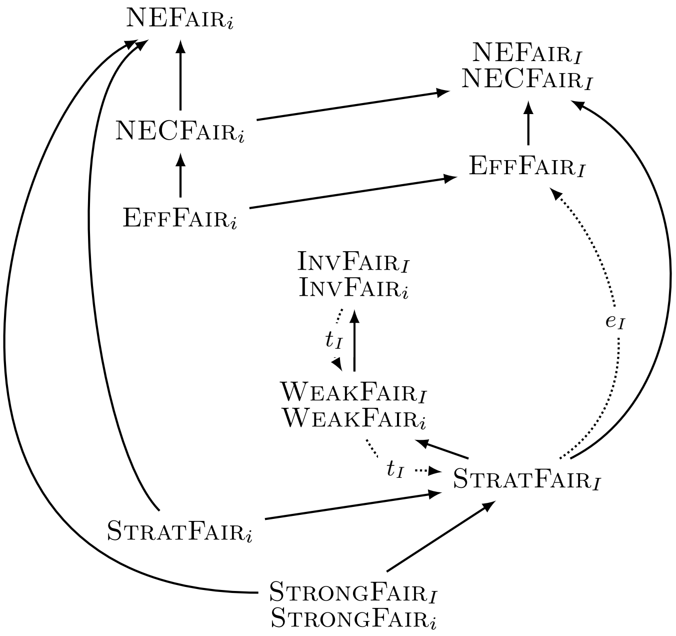

Fig. 1. Relationships between different notions of fairness. Solid arrows stand for implications, i.e., lead from stronger to weaker definitions of fairness. Dashed arrows represent implications that hold only under additional assumptions of effectiveness (e) or strategic timeliness (t). Missing arrows correspond to implications that do not hold. *Note:* we did not include arrows that follow from transitivity of implication.

Non-enforced Fairness Now we study how StrongFair, StratFair, WeakFair and InvFair relate to NEFair.

Theorem 2. StratFair *implies* NEFair*.*

Proof. Assume $\forall\Box(\mathsf{NRO} \to \langle\!\langle A_h\rangle\!\rangle\Diamond\mathsf{NRR})$. Because $\forall\Box\varphi \to \neg\exists\Diamond\neg\varphi$ is a CTL validity, we have $\neg\exists\Diamond(\mathsf{NRO} \wedge \neg\langle\!\langle A_h\rangle\!\rangle\Diamond\mathsf{NRR})$. Therefore, by Fact 1 it holds that $\neg\langle\!\langle B\rangle\!\rangle\Diamond(\mathsf{NRO} \wedge \neg\langle\!\langle A_h\rangle\!\rangle\Diamond\mathsf{NRR})$.

Similarly, StrongFair implies NEFair as well. Also, because StratFair, WeakFair and InvFair are equivalent given strategic timeliness, WeakFair and InvFair imply NEFair given strategic timeliness. Now we show that the other implications do not hold. Protocol 9 satisfies NEFair, but not StrongFair, StratFair, WeakFair or InvFair. Protocol 7, a protocol that does not satisfy strategic timeliness, satisfies InvFair and WeakFair, but not NEFair. Finally, StratFair$_i$ ⇒ NECFair$_i$ is not valid, as it does not hold in a model with a state q with a next state where $\mathsf{NRO} \wedge \neg\mathsf{NRR}$ holds such that q is indistinguishable from the initial state.

Protocol 6

```
1. $B \to T$: NRR
2. $T \to A$: continue
3. $A \to B$: NRO
4. Choice for $A$:
   (a) 1. $A \to T$: true
       2. $T \to A$: NRR
   (b) 1. $A \to T$: false
       2. Go to 4.
```

Protocol 7

```
1. $A \to B$: NRO
2. Choice for $B$:
   (a) 1. $B \to A$: NRR
   (b) 1. $B \to A$: cont.
       2. Go to 2.
```

Protocol 8

```
1. $A \to B$: NRO
2. $B \to A$: continue
3. Go to 2.
```

Protocol 9

```
1. $B \to T$: NRR
2. $A \to B$: NRO
3. Choice for $A$:
   (a) 1. End of protocol.
   (b) 1. $A \to T$: cont.
       2. $T \to A$: NRO
```

Fig. 2. Counterexample protocols

Moreover, as shown in Section 3.2, NEFair and NECFair are equivalent under perfect information, while under imperfect information, NECFair implies NECFair, but not vice versa.

Effective Fairness We proceed by studying the relations between EffFair and the other definitions of fairness. EffFair implies NEFair, as shown in Theorem 1. The following theorem states that in effective protocols, StratFair$_I$ implies EffFair$_I$. This theorem does only hold assuming perfect information. Under imperfect information, Alice is not guaranteed to know whether Bob plays honest, and cannot decide whether she should continue the cooperation with Bob or not.

Theorem 3. *Whenever* Effective$_I$ *holds,* StratFair$_I$ *implies* EffFair$_I$.

Proof. Assume that Effective$_I$ and StratFair$_I$ hold. We set $\varphi = \neg(\mathsf{NRO} \wedge \neg\langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR})$ and $\psi = \mathsf{NRO} \wedge \mathsf{NRR}$. StratFair$_I$ $= \forall\Box(\mathsf{NRO} \to \langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR})_I$ is equivalent to $\forall\Box\neg(\mathsf{NRO} \wedge \neg\langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR})_I$ and can thus be written as $(\forall\Box\varphi)_I$. This means that for all paths $\lambda \in out(q, \emptyset)$ and all positions $i \geq 0$, we have $\lambda[i] \models_I \forall\Box\varphi$ as well (1). Effective$_I$ can be written as $(\langle\langle A_h, B_h\rangle\rangle\Diamond\psi)_I$. By definition of $\Diamond$, there exists a pair F of strategies for agents A_h and B_h, respectively, such that for all $\lambda \in out(q, F)$ there exists $i \geq 0$ with $\lambda[i] \models_I \psi$ (2). Let F be a pair of strategies for A and B satisfying this condition. Then we have that for all $\lambda \in out(q, F)$ there exists $i \geq 0$ with $\lambda[i] \models_I \psi$ by (2), and for all $0 \leq j < i$, we have $\lambda[j] \models_I \langle\langle A_h\rangle\rangle\Box\varphi$ by (1). By definition of $\mathcal{U}$, we obtain $q \models_I \langle\langle A_h, B_h\rangle\rangle((\langle\langle A_h\rangle\rangle\Box\neg(\mathsf{NRO} \wedge \neg\langle\langle A_h\rangle\rangle\Diamond\mathsf{NRR}))\ \mathcal{U}\ (\mathsf{NRO} \wedge \mathsf{NRR}))$, i.e., EffFair$_I$.

Again, these results, and the transitive closures of them, are all the implications that hold. Protocol 5 satisfies NEFair, but not EffFair. Furthermore, the empty protocol, which obviously does not satisfy effectiveness, satisfies StrongFair, StratFair, WeakFair and InvFair, but not EffFair. Finally, Protocol

7, not satisfying strategic timeliness, satisfies WeakFair and InvFair, but not EffFair.

5 Related Work

Various definitions of non-repudiation and fair exchange have been formalized and verified with LTL, cf. e.g. [2, 3]. However, as we argue in this paper, these definitions are often either too strong or too weak because they do not take into account the agents' ability to choose the right strategy. In this section, we discuss how our results relate to existing proposals about verification of non-repudiation protocols and other fair-exchange protocols with the strategic logic ATL.

Kremer and Raskin [16] use NEFair to verify various non-repudiation protocols. They find flaws in the Zhou-Gollmann optimistic protocol [20], the Asokan-Shoup-Waidner certified mail protocol [7] and the Markowitch-Kremer multi-party non-repudiation protocol [21]. An improved version of the latter protocol, as well as the Kremer-Markowitch non-repudiation protocol [22], are shown to satisfy NEFair. However, as we have seen in Sec. 3.2, the protocols that are shown to satisfy NEFair might still be unfair if the agents' strategies are not executable due to imperfect information. Furthermore, all strategies that guarantee fairness in these protocols might be ineffective, as we proved in Sec. 3.3.

Chadha et al. [10] demonstrate that the GM protocol [23], a multi-party contract signing protocol, does not satisfy InvFair, WeakFair, StratFair and StrongFair for four participants. However, as we have seen, non-enforced fairness might still hold. It can be argued that non-enforced fairness is sufficient, if it is assumed that Alice has the ability to resolve the choices in a non-deterministic protocol in the way that is the most advantageous for her.

Liu et al. [24] propose an extended CEM (certified e-mail) protocol with TTP transparency and use StratFair to prove fairness. However, strategic timeliness is only checked in a perfect information model, which means that the protocol may be intuitively unfair in the presence of imperfect information, as we saw in Sec. 3.2. Furthermore, the extended CEM protocol does not necessarily have strong fairness, as StratFair does not imply StrongFair. This means that it is still important that the agents resolve the non-determinism of the protocol in the correct way.

Finally, Zhang et al. [25] analyze a number of multi-party contract signing protocols. WeakFair and InvFair are used to prove that the MR protocol [26] is fair with up to 5 signers, and that the MRT protocol [27] with 3 signers has a flaw. Furthermore, a corrected MRT protocol for 3 and 4 signers is presented, which is shown to satisfy WeakFair and InvFair. Because strategic timeliness is proven, the results carry over to StratFair. We saw in Sec. 4 that StratFair does not imply StrongFair, and that NEFair does not imply StratFair. Therefore, it could be that both the original and the corrected version of the MRT protocol satisfy NEFair, i.e., are fair assuming agents have enough rationality to take the correct choices. On the other hand, it could be that both the original and corrected version of the MRT protocol lack StrongFair, i.e., that in both protocols, not every way of resolving non-determinism leads to fairness. In the same way does the successful verification of StratFair in the MR protocol not guarantee NEFair. Furthermore, as strategic timeliness is only checked in a perfect information model, the MR protocol and the corrected MRT protocol might be only fair under the unrealistic assumption of perfect information (see Sec. 4).

6 Conclusions and Future Work

We have shown that there are a number of problems involved with the specifications of fairness that are currently used for the verification of non-repudiation and other fair-exchange protocols. First, one of the definitions of fairness, non-enforced fairness, accepts intuitively unfair protocols, because it has been overlooked that agents can have imperfect information. This makes it clear that formal verification should take imperfect information into account. We have proposed a new definition of fairness that can be used in models with imperfect information. Furthermore, we have shown that fairness is not a sufficient requirement for fair-exchange protocols, as protocols are also required to be effective. We have shown that if both fairness and effectiveness are expressed in terms of strategies, the two requirements cannot be combined easily. We have proposed a new definition of fairness that combines the requirements correctly. Moreover, we have given a hierarchy of the various definitions of fairness, and have proven that this hierarchy is correct. Finally, we have indicated the consequences of our results for existing results from literature. We have shown two problems with the specifications of fairness that are currently used for verification of non-repudiation and other fair-exchange protocols, namely the implicit assumption of perfect information and the possible lack of effectiveness. We have also proposed new definitions of fairness that handle the issues appropriately. Moreover, we have established a hierarchy of fairness definitions, and indicated the consequences of our results for existing work.

Depending on the assumptions about the agents, different definitions of fairness would be advisable to use. If the agents are not rational and should be protected against taking bad decisions, then StrongFair is clearly the best option. If the agents are rational, the situation is more sophisticated, as we know how to specify fairness and effectiveness under imperfect information but *not* both at the same time. To find as many flaws as possible, we recommend to verify EffFair in imperfect information semantics. However, even protocols that satisfy this specification might be flawed: EffFair guarantees the existence of a strategy that is both fair and executable with imperfect information, and the existance of a strategy that is both fair and effective, but not the existance of a strategy that is both executable, fair and effective. More research is required to find directions to solve this problem.

In the future, we hope to find a specification that imposes *both* fairness and effectiveness under imperfect information. Furthermore, it would be interesting to study ATL specifications of *abuse-freeness*, a property that guarantees that no signer can prove to an outside observer that he is able to determine the result of the protocol. Moreover, we hope to verify the concepts of fairness for existing non-repudiation protocols. This may require a fundamental extension of verification techniques as there are no ATL model checkers for imperfect information. There was an attempt in one of the older versions of MCMAS [28], but because of conceptual as well as computational problems the extension was subsequently abandoned. Also, the ALPAGA model checker [29] can only solve a limited fragment of imperfect information games.

Acknowledgements The authors would like to thank Jun Pang, Jean-François Raskin and Leon van der Torre for valuable comments and suggestions. This work was supported by the FNR (National Research Fund) Luxembourg under projects S-GAMES, C08/IS/03 and GMASec - PHD/09/082.

References

1. Roscoe, A.: Intensional Specifications of Security Protocols. In: Proc. CSFW '96, IEEE (1996) 28–38
2. Lowe, G.: A hierarchy of authentication specifications. In: 10th Computer Security Foundations Workshop (CSFW '97), June 10-12, 1997, Rockport, Massachusetts, USA, IEEE Computer Society (1997) 31–44
3. Cremers, C., Mauw, S., de Vink, E.: Injective synchronisation: an extension of the authentication hierarchy. Theoretical Computer Science **367** (2006) 139–161 Special issue on ARSPA'05, (P. Degano and L. Viganò, eds.).
4. Benaloh, J., Tuinstra, D.: Receipt-free secret ballot elections (extended abstract). In: Proc. 26th ACM Symposium on the Theory of Computing (STOC), ACM (1994) 544–553
5. Delaune, S., Kremer, S., Ryan, M.: Coercion-resistance and receipt-freeness in electronic voting. In: Proceedings of the 19th IEEE Computer Security Foundations Workshop (CSFW'06), Venice, Italy, IEEE Computer Society Press (July 2006)
6. van Deursen, T., Mauw, S., Radomirović, S., Vullers, P.: Secure ownership and ownership transfer in RFID systems. In: Proc. 14th European Symposium On Research In Computer Security (ESORICS'09). Volume 5789 of Lecture Notes in Computer Science., Springer (2009) 637–654
7. Asokan, N., Shoup, V., Waidner, M.: Asynchronous Protocols for Optimistic Fair Exchange. Proc. of the IEEE Symp. in Security and Privacy (1998) 86–99
8. Alur, R., Henzinger, T.A., Kupferman, O.: Alternating-time temporal logic. Journal of the ACM **49** (2002) 672–713
9. Alur, R., Henzinger, T., Mang, F., Qadeer, S., Rajamani, S., Tasiran, S.: MOCHA: Modularity in model checking. In Hu, A., Vardi, M., eds.: Computer Aided Verification. Volume 1427 of LNCS. Springer Berlin / Heidelberg (1998) 521–525
10. Chadha, R., Kremer, S., Scedrov, A.: Formal Analysis of Multiparty Contract Signing. Journal of Automated Reasoning **36** (2006) 39–83
11. Dashti, M.T.: Keeping Fairness Alive. PhD thesis, Vrije Universiteit, Amsterdam (2008)
12. Ben-Or, M., Goldreich, O., Micali, S., Rivest, R.: A fair protocol for signing contracts. IEEE Transactions on Information Theory **IT-36** (1990) 40–46
13. Even, S., Yacobi, Y.: Relations among public key signature systems (1980)

14. Ezhilchelvan, P.D., Shrivastava, S.K.: Systematic Development of a Family of Fair Exchange Protocols. In: Proc. of the 17th Annual IFIP WG 11.3 Working Conference on Database and Applications Security, Kluwer Academic Press (2003) 243–258
15. Liu, P.: Avoiding loss of fairness owing to failures in fair data exchange systems. Decision Support Systems **31** (2001) 337–350
16. Kremer, S., Raskin, J.F.: A game-based verification of non-repudiation and fair exchange protocols. Journal of Computer Security **11** (2003)
17. Jamroga, W., van der Hoek, W.: Agents that know how to play. Fundamenta Informaticae **63** (2004) 185–219
18. Schobbens, P.Y.: Alternating-time logic with imperfect recall. Electronic Notes in Theoretical Computer Science **85** (2004) 82–93
19. Jamroga, W., Bulling, N.: Comparing variants of strategic ability. In: Proceedings of EUMAS2010. (2010)
20. Zhou, J., Gollmann, D.: An efficient non-repudiation protocol. Proceedings 10th Computer Security Foundations Workshop (1997) 126–132
21. Markowitch, O., Kremer, S.: A multi-party optimistic non-repudiation protocol. Proc. of The 3rd Int. Conf. on Information Security and Cryptology (ICISC 2000), volume 2015 of LNCS **2015** (2000) 109–122
22. Kremer, S., Markowitch, O.: Optimistic non-repudiable information exchange. In Biemond, J., ed.: 21th Symp. on Information Theory in the Benelux, Werkgemeenschap Informatie- en Communicatietheorie, Enschede (2000) 139–146
23. Garay, J., MacKenzie, P.: Abuse-free multi-party contract signing. Distributed Computing (1999) 846–846
24. Liu, Z., Pang, J., Zhang, C.: Verification of A Key Chain Based TTP Transparent CEM Protocol. UNU-IIST (2010) 60
25. Zhang, Y., Zhang, C., Pang, J., Mauw, S.: Game-based verification of multi-party contract signing protocols. In Pierpaolo Degano, J.G., ed.: Proc. Formal Aspects in Security and Trust. Volume 5983 of LNCS., Springer-Verlag (2009) 186–200
26. Mukhamedov, A., Ryan, M.: Fair multi-party contract signing using private contract signatures. Information and Computation **206** (2008) 272–290
27. Mauw, S., Radomirovic, S., Dashti, M.T.: Minimal Message Complexity of Asynchronous Multi-party Contract Signing. IEEE (2009)
28. Lomuscio, A., Qu, H., Raimondi, F.: MCMAS: A Model Checker for the Verification of Multi-Agent Systems. In Bouajjani, A., Maler, O., eds.: Computer Aided Verification. Volume 5643 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2009) 682–688
29. Berwanger, D., Chatterjee, K., De Wulf, M., Doyen, L., Henzinger, T.: Alpaga: A Tool for Solving Parity Games with Imperfect Information. In Kowalewski, S., Philippou, A., eds.: Tools and Algorithms for the Construction and Analysis of Systems. Volume 5505 of LNCS. Springer Berlin / Heidelberg (2009) 58–61