A Theory of Agreements and Protection

Massimo Bartoletti[1], Tiziana Cimoli[1], and Roberto Zunino[2]

[1] Università degli Studi di Cagliari, Italy
[2] Università di Trento and COSBI, Italy

Abstract. We present a theory of contracts. Contracts are interacting processes with an explicit notion of obligations and objectives. We model processes and their obligations as event structures. We define a general notion of *agreement*, by interpreting contracts as multi-player concurrent games. A participant agrees on a contract if she has a strategy to reach her objectives (or make another participant chargeable for a violation), whatever the moves of her adversaries. We then tackle the problem of *protection*. A participant is protected by a contract when she has a strategy to defend herself in all possible contexts, even in those where she has not reached an agreement. We show that, in a relevant class of contracts, agreements and protection mutually exclude each other. We then propose a novel formalism for modelling contractual obligations: event structures with circular causality. Using this model, we show how to construct contracts which guarantee both agreements and protection.

1 Introduction

The lack of precise guarantees about the reliability and security of cloud services is a main deterrent for industries wishing to move their applications and business to the cloud [1]. A key problem is how to drive safe and fair interactions among distributed participants which are possibly mutually distrusted, and have possibly conflicting individual goals. In addition to the well-known difficulties of distributed software systems (distribution, concurrency, heterogeneity, mobility, *etc.*), cloud components and infrastructures are often under the governance of different providers, possibly competing among each other. Analysis and verification techniques can be applied only on the software components under one's control, while no assumptions can be made about the components made available by other providers. Therefore, standard compositional techniques have to be adapted to cope with the situation where providers fail to keep the promises made, or even choose not to.

We envision a *contract-oriented computing* paradigm [3], where reliable interactions are driven by contracts which formalise Service-Level Agreements. Contracts specify the behavior of a software component, from the point of view of the interactions it may participate in, and the goals it tries to reach. Differently from most of the approaches based on behavioural types [15], which use contracts only in the "matchmaking" phase, a contract-oriented component is not supposed to be *honest*, in that it may not keep the promises made.

D. Basin and J.C. Mitchell (Eds.): POST 2013, LNCS 7796, pp. 186–205, 2013.

In a contract-oriented application, participants advertise their contracts to some *contract brokers*, which are the contract-oriented analogous of service repositories in the Web Service paradigm. Then, participants wait until the contract broker finds an *agreement* among the contracts in its hands. When this happens, a session is created among the participants involved in the contract, so that they can interact. Agreement is a property of contracts which guarantees that each honest participant may reach her objectives, provided that the other participants cooperate honestly. Moreover, if an honest participant does not reach her goals, then some other participant can be blamed. An external judge may then inspect the contract and the status of the session. In case a violation is found, the judge will eventually provide the prescribed compensations/punishments.

The underlying assumption of this view is that participants trust in the contract broker. In a context populated by attackers, it may happen that a dishonest contract broker creates a fraudulent session, making participants interact in the absence of an agreement. In this way, the contract broker may allow an accomplice to swindle an unaware participant. Note that the accomplice may perform his scam while adhering to his contract, and so he will not be liable for violations.

A crucial problem is how to devise contracts which protect participants from malicious contract brokers, while at the same time allowing honest brokers to find agreements. A good contract should allow a participant to reach her goals in contexts where the other participants are cooperative, and prevent her from performing imprudent actions which could be exploited by malicious participants.

In this paper we propose a foundational model for contracts. We specify the behaviour of participants as event structures [21], a basic semantic model for interactive systems. We then provide a formal definition for the two key notions intuitively introduced above, i.e. *agreement* and *protection*. To do that, we borrow techniques from game theory, by interpreting contracts as multi-player concurrent games. By abstracting away from the concrete details of contract languages, our model is a first step towards a unifying framework for reasoning about contracts, in the same spirit that event structures can be used as an underlying semantics for a variety of concrete models of concurrency.

A first result is that agreement and protection cannot coexist for a broad class of objectives. That is, if we are given the objectives of a set of participants, it is impossible to construct a contract which protects them all, and at the same time admits an agreement. Roughly, the problem is that, when the offers of the participants mutually depend on their requests, the participant which risks in doing the first step is not protected.

To overcome this negative result, we extend event structures with a new notion of causality. While in classical event structures an action a which causally depends on an action b can only be performed after b, in our extension we also consider a relaxed version of causality, which allows a to happen before b, under the (legally binding) promise that b will be eventually performed.

The main result of the paper is that, using this model for contracts, it is possible for a wide class of objectives to construct a set of contracts which protect their participants and still admit an agreement.

2 Contracts

At an abstract level, contracts are concurrent systems, enriched with a notion of *obligation* (what I must do in a given state) and *objective* (what I expect to obtain in a given state). Event structures (ES) are one of the classical models for concurrency [21]. Notwithstanding the variety of formalisations, ES are at least equipped with an *enabling* relation modelling causality (usually written $\vdash$), and another relation modelling non-determinism (usually written $\#$). ES can provide a basic semantic model for contractual clauses, by interpreting the enabling $\{b\} \vdash a$ as: "I am obliged to do a *after* you have done b".

2.1 Event Structures

Event structures [21] have a deep theory. Here we only report some basic definitions, which are needed in our technical development. We assume an enumerable universe of *events*, ranged over by $a, b, e, \ldots$. For a set of events X, the predicate $CF(X)$ is true iff X is *conflict-free*, i.e. $CF(X) \triangleq (\forall e, e' \in X : \neg(e \# e'))$.

Definition 1 (Event structure [21]). *An event structure $\mathcal{E}$ is a triple $\langle E, \#, \vdash \rangle$, where (1) E is a set of events, (2) $\# \subseteq E \times E$ is an irreflexive and symmetric* conflict *relation (3) $\vdash \subseteq \{X \subseteq_{fin} E \mid CF(X)\} \times E$ is the* enabling *relation, which is* saturated*, i.e. $\forall X \subseteq Y \subseteq_{fin} E.\ X \vdash e \wedge CF(Y) \implies Y \vdash e$.*

An ES is *finite* when E is finite; it is *conflict-free* when the conflict relation is empty. We shall often use the following shorthands: $X \vdash Y$ for $\forall e \in Y.\ X \vdash e$, $a \vdash b$ for $\{a\} \vdash b$, and $\vdash e$ for $\emptyset \vdash e$.

Definition 2 (Persistent conflict). *An event $e \in E$ is* persistently conflictable *in $\mathcal{E}$ iff the set $\{\bar{e} \in E \mid e \# \bar{e}\}$ is infinite. A set $X \subseteq E$ is persistently conflictable iff some $e \in X$ is persistently conflictable.*

For a sequence $\sigma = \langle e_0\, e_1 \cdots \rangle$ in E (possibly infinite), we write $\overline{\sigma}$ for the set of elements in σ; we write σ_i for the subsequence $\langle e_0 \cdots e_{i-1} \rangle$. If $\sigma = \langle e_0 \cdots e_n \rangle$, we write $\sigma\, e$ for the sequence $\langle e_0 \cdots e_n\, e \rangle$. The empty sequence is denoted by ε. For a set S, we denote with S^* the set of finite sequences over S, and with S^ω the set of finite and infinite sequences over S.

A configuration C is a "snapshot" of the behaviour of the system modeled by an ES, where for each event $e \in C$ it is possible to find a finite justification, *i.e.* a sequence of events containing all the causes of e.

Definition 3 (Configuration [21]). *For an* ES $\mathcal{E} = \langle E, \#, \vdash \rangle$*, a set $C \subseteq E$ is a* configuration of $\mathcal{E}$ *iff $CF(C)$, and*

$$\forall e \in C.\ \exists \sigma = \langle e_0 \cdots e_n \rangle.\ e \in \overline{\sigma} \subseteq C \ \wedge\ \forall i \leq n.\ \overline{\sigma_i} \vdash e_i$$

The set of all configurations of $\mathcal{E}$ is denoted by $\mathcal{F}_\mathcal{E}$.

Definition 4 (LTS of an ES). *For an* ES $\mathcal{E}$*, the labelled transition system* $\mathrm{LTS}_{\mathcal{E}} = \langle \wp_{fin}(E), E, \rightarrow_{\mathcal{E}} \rangle$ *is defined as follows:*

$$C \xrightarrow{e}_{\mathcal{E}} C \cup \{e\} \qquad \text{iff } C \vdash e,\ e \notin C \text{ and } CF(C \cup \{e\})$$

Definition 5. *For two* ES $\mathcal{E}, \mathcal{E}'$*, we define* $\mathcal{E} \sqcup \mathcal{E}'$ *as the pointwise union of* $\mathcal{E}, \mathcal{E}'$*.*

2.2 An Event-Based Model of Contracts

A contract (Def. 6) specifies the obligations and the objectives of a set of participants. The atomic entities of a contract are the *events*, which are uniquely associated to participants through a labelling π. Obligations are modelled as an event structure. Intuitively, an enabling $X \vdash e$ models the fact that, if all the events in X have happened, then e is an obligation for $\pi(e)$. Such obligation may be discharged only by performing e, or any event in conflict with e. For instance, consider an internal choice between two events a and b. This is modelled by an ES with enablings $\vdash a$, $\vdash b$ and conflict $a\#b$. After the choice (say, of a), the obligation b is discharged. Objectives are modelled as a function Φ, which associates each participant A and each trace of events σ to a *payoff* $\Phi\mathsf{A}\sigma$. We assume a rather coarse notion of payoffs: we only have three possible outcomes which represent, respectively, success (1), failure (-1), and tie (0).

Definition 6 (Contract). *A contract* $\mathcal{C}$ *is a 4-tuple* $\langle \mathcal{E}, \mathcal{A}, \pi, \Phi \rangle$*, where:*

- $\mathcal{E} = \langle E, \#, \vdash \rangle$ *is an event structure;*
- $\mathcal{A}$ *is a set of* participants *(ranged over by* $\mathsf{A}, \mathsf{B}, \ldots$*);*
- $\pi : E \to \mathcal{A}$ *associates each event with a participant;*
- $\Phi : \mathcal{A} \rightharpoonup E^{\omega} \to \{-1, 0, 1\}$ *associates each participant and trace with a* payoff.

Hereafter, we shall assume that contracts respect two basic requirements. For all $X \vdash e$ in $\mathcal{E}$, we ask that (i) $\Phi(\pi(e)) \neq \bot$, and (ii) e is not persistently conflictable in $\mathcal{E}$. Notice that Φ is a partial function (from $\mathcal{A}$ to functions), hence a contract does not need to define payoffs for all the participants in $\mathcal{A}$ (typically, when A advertises her contract, she will not speculate about the objectives of B). Constraint (i) asks that if a contract defines some obligations for A, then A must also declare in $\mathcal{C}$ her payoffs. Constraint (ii) rules out those ill-formed contracts where some obligations can be persistently discharged.

Example 1. Suppose there are two kids who want to play together. Alice has a toy airplane, while Bob has a bike. Both kids are willing to share their toys, but they do not trust each other. Thus, before starting to play they advertise the following contracts. Alice will lend her airplane only *after* Bob has allowed her ride his bike. Bob will lend his bike without conditions. We model the events "Alice lends her airplane" and "Bob lends his bike" as a and b, respectively. The obligations of Alice and Bob are modelled by the following event structures:

$$\mathcal{E}_{\mathsf{A}}: \ \{b\} \vdash a \qquad\qquad \mathcal{E}_{\mathsf{B}}: \ \emptyset \vdash b$$

The objectives of the two kids are modelled by the functions Φ_{A} (which establishes Alice's payoff) and Φ_{B} (for Bob). Alice has a positive payoff in those traces where b has been performed, while she has a negative payoff when she performs a while not obtaining b in return. The payoffs of Bob are dual. Formally:

$$\Phi_{\mathsf{A}}\mathsf{A} \;=\; \lambda\sigma.\; \begin{cases} 1 & \text{if } b \in \overline{\sigma} \\ 0 & \text{if } a, b \notin \overline{\sigma} \\ -1 & \text{otherwise} \end{cases} \qquad \Phi_{\mathsf{B}}\mathsf{B} \;=\; \lambda\sigma.\; \begin{cases} 1 & \text{if } a \in \overline{\sigma} \\ 0 & \text{if } b, a \notin \overline{\sigma} \\ -1 & \text{otherwise} \end{cases}$$

Summing up, the contracts of Alice and Bob are $\mathcal{C}_{\mathsf{A}} = \langle \mathcal{E}_{\mathsf{A}}, \mathcal{A}, \pi, \Phi_{\mathsf{A}} \rangle$ and $\mathcal{C}_{\mathsf{B}} = \langle \mathcal{E}_{\mathsf{B}}, \mathcal{A}, \pi, \Phi_{\mathsf{B}} \rangle$, respectively, where $\mathcal{A} = \{\mathsf{A}, \mathsf{B}\}$, $\pi(a) = \mathsf{A}$, and $\pi(b) = \mathsf{B}$. □

Observe that the definition of payoff functions in Def. 6 is quite liberal. Indeed, it also allows for uncomputable functions, which are of little use in doing anything with a contract. One may then be interested in considering relevant subclasses of payoff functions, in the same spirit of the rich classification of winning conditions in game theory [9].

Assume that participant A has a sequence $\langle O^0\, O^1 \cdots \rangle$ of sets of events, and a sequence $\langle R^0\, R^1 \cdots \rangle$ of the same cardinality. The sets O^i are called *offers*, while R^i are the *requests*. A function Φ is an *Offer-Request (O-R) payoff* for A if, whenever A performs in σ some offer O^i (in whatever order), then σ also contains the corresponding request R^i. For instance, the payoff functions Φ_{A} and Φ_{B} in Ex. 1 are O-R payoffs for A and B. The offers and the requests of A and B are, respectively $O^0_{\mathsf{A}} = \{a\} = R^0_{\mathsf{B}}$ and, dually, $O^0_{\mathsf{B}} = \{b\} = R^0_{\mathsf{A}}$.

Definition 7 (Offer-Request payoff). *Let $\pi : E \to \mathcal{A}$. We say that Φ is a* Offer-Request payoff for A *iff there exist (possibly infinite) sequences $(O^i)_i$, $(R^i)_i$ of equal cardinality such that for all i, $O^i \subseteq \pi^{-1}(\mathsf{A})$, $\emptyset \neq R^i \subseteq E \setminus \pi^{-1}(\mathsf{A})$, and*

$$\Phi\mathsf{A} \;=\; \lambda\sigma.\; \begin{cases} 1 & \textit{if } (\exists i.\; R^i \subseteq \overline{\sigma}) \;\wedge\; (\forall j.\; O^j \subseteq \overline{\sigma} \implies R^j \subseteq \overline{\sigma}) \\ 0 & \textit{if } (\forall i.\; R^i \not\subseteq \overline{\sigma} \;\wedge\; O^i \not\subseteq \overline{\sigma}) \\ -1 & \textit{otherwise} \end{cases}$$

A contract $\mathcal{C} = \langle \mathcal{E}, \mathcal{A}, \pi, \Phi \rangle$ has O-R payoffs iff Φ is a O-R payoff for all $\mathsf{A} \in \mathcal{A}$. If, additionally, all the sets O^i (resp. R^i) are finite for all $\mathsf{A} \in \mathcal{A}$, we say that $\mathcal{C}$ has finite offers *(resp.* finite requests*). If Φ has a finite number of finite offers-request, then Φ is* finite*.*

Example 2. In [8] contracts are modelled in a variant of CCS which includes prefixing, internal/external choice, and recursion. Consider e.g. a server A which repeatedly offers to her clients a choice between two actions a and b. The client B internally chooses one of his (co-)actions $\overline{a}$ and $\overline{b}$. This is modelled in [8] as:

$$c_{\mathsf{A}} = \mathit{rec}\; X.\; (a.X + b.X) \qquad\qquad c_{\mathsf{B}} = \mathit{rec}\; Y.\; (\overline{a}.Y \oplus \overline{b}.Y)$$

In our theory we model c_{A} and c_{B} as the contracts $\mathcal{C}_{\mathsf{A}}$ and $\mathcal{C}_{\mathsf{B}}$, defined below. For all $i \geq 0$, let a_i, b_i be events of A, and let $\overline{a_i}, \overline{b_i}$ be events of B. The event structures of A and B have the following enablings and conflicts, for all $i \geq 0$:

$$\mathcal{E}_{\mathsf{A}}: \ \overline{a_i} \vdash a_i, \ \ \overline{b_i} \vdash b_i, \ \ a_i \# b_i$$
$$\mathcal{E}_{\mathsf{B}}: \ \vdash \overline{a_0}, \ \ \vdash \overline{b_0}, \ \ a_i \vdash \overline{a_{i+1}}, \ \ a_i \vdash \overline{b_{i+1}}, \ \ b_i \vdash \overline{a_{i+1}}, \ \ b_i \vdash \overline{b_{i+1}}, \ \ \overline{a_i} \# \overline{b_i}$$

The payoff of a A is positive in a play σ if A has no obligations; similarly for B.

$$\Phi_{\mathsf{P}}\mathsf{P} \ = \ \lambda\sigma. \ \begin{cases} 1 & \text{if } \nexists e \in \pi^{-1}(\mathsf{P}).\ \overline{\sigma} \xrightarrow{e}_{\mathcal{E}_{\mathsf{P}}} \\ -1 & \text{otherwise} \end{cases} \qquad \square$$

Given two contracts $\mathcal{C}, \mathcal{C}'$, we denote with $\mathcal{C} \mid \mathcal{C}'$ their composition. If $\mathcal{C}'$ is the contract written by an adversary of $\mathcal{C}$, then a naïve composition of the two contracts could easily lead to an attack, e.g. when Mallory's contract says that Alice is obliged to give him her airplane. To prevent from such kinds of attacks, contract composition is a partial operation. We do *not* compose contracts which assign payoffs to the same participant, neither those which disagree on the association between events and participants.

Definition 8 (Composition of compatible contracts). *Two contracts* $\mathcal{C} = \langle \mathcal{E}, \mathcal{A}, \pi, \Phi \rangle$ *and* $\mathcal{C}' = \langle \mathcal{E}', \mathcal{A}', \pi', \Phi' \rangle$ *are* compatible *whenever:*

$$\forall e \in \mathcal{E} \cap \mathcal{E}'. \ \ e = e' \implies \pi(e) = \pi'(e) \tag{1}$$
$$\forall \mathsf{A} \in \mathcal{A} \cup \mathcal{A}'. \ \ \Phi(\mathsf{A}) = \bot \ \ \vee \ \ \Phi'(\mathsf{A}) = \bot \tag{2}$$

If $\mathcal{C}$, $\mathcal{C}'$ *are compatible, we define their composition as:*

$$\mathcal{C} \mid \mathcal{C}' = \langle \mathcal{E} \sqcup \mathcal{E}', \mathcal{A} \cup \mathcal{A}', \pi \sqcup \pi', \Phi \sqcup \Phi' \rangle$$

Two contracts which both assign obligations to A are not compatible.

Lemma 1. *If* $\mathcal{C} = \langle \mathcal{E}, \mathcal{A}, \pi, \Phi \rangle$ *and* $\mathcal{C}' = \langle \mathcal{E}', \mathcal{A}', \pi', \Phi' \rangle$ *are compatible, then:*

$$X \vdash e \in \mathcal{E} \ \wedge \ X' \vdash e' \in \mathcal{E}' \implies \pi(e) \neq \pi'(e') \ \ \wedge \ \ e \neq e'$$

Example 3. The contracts $\mathcal{C}_{\mathsf{A}}$ and $\mathcal{C}_{\mathsf{B}}$ in Ex. 1 are compatible, and their composition is the contract $\mathcal{C} = \mathcal{C}_{\mathsf{A}} \mid \mathcal{C}_{\mathsf{B}} = \langle \mathcal{E}, \mathcal{A}, \pi, \Phi \rangle$ defined as follows:

$$\begin{array}{ll} \mathcal{E}: & \{b\} \vdash a, \ \ \emptyset \vdash b \\ \mathcal{A}: & \{\mathsf{A}, \mathsf{B}\} \\ \pi: & \pi(a) = \mathsf{A}, \ \ \pi(b) = \mathsf{B} \end{array} \qquad \Phi\mathsf{P} \ = \ \begin{cases} \Phi_{\mathsf{A}}\mathsf{A} & \text{if } \mathsf{P} = \mathsf{A} \\ \Phi_{\mathsf{B}}\mathsf{B} & \text{if } \mathsf{P} = \mathsf{B} \end{cases} \qquad \square$$

2.3 Agreements

A crucial notion on contracts is that of *agreement*. Intuitively, when Alice agrees on a contract $\mathcal{C}$, then she can safely initiate an interaction with the other participants, and be guaranteed that the interaction will not "go wrong" — even in the presence of attackers. This does not mean that Alice will always succeed in all interactions: in case Bob is dishonest, we do not assume that an external

authority (e.g. Bob's mother) will lend the bike to Alice. We intend that Alice agrees on a contract where, in all the interactions where she does not succeed, then some other participant must be found dishonest. That is, we consider Alice satisfied if she can blame another participant. In real-world applications, a judge may provide compensations to Alice, or impose a punishment to the participant who has violated the contract. Here, we shall not explicitly model the judge, and we shall focus instead on how to formalise the agreement property.

We interpret a contract $\mathcal{C} = \langle \mathcal{E}, \mathcal{A}, \pi, \Phi \rangle$ as a nonzero-sum concurrent multi-player game. The game involves the players in $\mathcal{A}$ concurrently performing events in order to reach the objectives defined by Φ. A *play* of $\mathcal{C}$ is a (finite or infinite) sequence of events of $\mathcal{E}$. We postulate that the permitted moves after a (finite) sequence of steps σ are exactly the events enabled by $\mathcal{E}$ in $\overline{\sigma}$, i.e. e is permitted in σ iff $\overline{\sigma} \xrightarrow{e}_{\mathcal{E}}$. A *strategy* Σ for A is a function which associates to each finite play σ a set of events of A (possibly empty), such that if $e \in \Sigma(\sigma)$ then σe is still a play. A play $\sigma = \langle e_0\, e_1 \cdots \rangle$ *conforms* to a strategy Σ for A if, for all $i \geq 0$, if $e_i \in \pi^{-1}(\mathsf{A})$, then $e_i \in \Sigma(\sigma_i)$.

As usual in concurrency, we shall only consider those *fair* plays where an event permanently enabled is eventually performed. Indeed, contracts would make little sense in the presence of unfair plays, because an honest participant willing to perform a promised action could be perpetually prevented (by an unfair scheduler) from keeping her promise.

Definition 9 (Fair play). *A play* $\sigma = \langle e_0\, e_1 \cdots \rangle$ *is* fair *w.r.t. strategy* Σ *iff:*

$$\forall i \leq |\sigma|.\ \big(\forall j : i \leq j \leq |\sigma|.\ e \in \Sigma(\sigma_j)\big) \implies \exists h \geq i.\ e_h = e$$

Our notion of agreement takes into account whether participants behave honestly in their plays. Informally, a participant is *innocent* in a play if she always keeps the promises made. An innocent participant has no persistently enabled events, i.e. all her enabled events are either performed or conflicted.

Definition 10 (Innocence). *We say that* A *is* innocent in σ *iff:*

$$\forall i \geq 0.\ \forall e \in \pi^{-1}(\mathsf{A}).\ \big(\overline{\sigma_i} \xrightarrow{e}_{\mathcal{E}} \implies \exists j \geq i.\ e_j \# e \ \vee\ e_j = e\big)$$

If A *is not innocent in* σ*, then we say she is* culpable.

There always exist strategies which guarantee A to be innocent in every (fair) play. The greatest of such strategies is the *eager strategy*, which prescribes A to do all her enabled events.

Lemma 2. *Say a strategy* Σ *for* A *is* innocent *iff* A *is innocent in all fair plays which conform to* Σ*. The* eager strategy $\Sigma^e_{\mathsf{A}} = \lambda\sigma.\, \{e \in \pi^{-1}(\mathsf{A}) \mid \overline{\sigma} \xrightarrow{e}_{\mathcal{E}}\}$ *is the greatest innocent strategy for* A.

We now define when a participant *wins* in a play. If A is culpable, then she loses. If A is innocent, but some other participant is culpable, then A wins. Otherwise, if all participants are innocent, then A wins if she has a positive payoff in the play. This is formalised as the function $\mathcal{W}$ in Def. 11 below.

Definition 11 (Winning play). *Define* $\mathcal{W} : \mathcal{A} \rightharpoonup E^\omega \to \{1, 0, -1\}$ *as:*

$$\mathcal{W}\mathsf{A}\sigma = \begin{cases} \Phi\mathsf{A}\sigma & \text{if all participants are innocent in } \sigma \\ -1 & \text{if } \mathsf{A} \text{ is culpable in } \sigma \\ +1 & \text{otherwise} \end{cases}$$

For a participant A *and a play* σ*, we say that* A wins *(resp.* loses*) in* σ *iff* $\mathcal{W}\mathsf{A}\sigma > 0$ *(resp.* $\mathcal{W}\mathsf{A}\sigma < 0$*).*

We can now define when a participant *agrees* on a contract. Intuitively, A is happy to participate in an interaction regulated by contract $\mathcal{C}$ when she has a strategy Σ which allows her to win in all fair plays conform to Σ. More formally, we say that Σ is *winning* (resp. *losing*) for A iff A wins (resp. loses) in every fair play which conforms to Σ.

Definition 12 (Agreement). *A participant* A agrees *on a contract* $\mathcal{C}$ *if and only if* A *has a winning strategy in* $\mathcal{C}$*. A contract* $\mathcal{C}$ admits an agreement *whenever all the involved participants agree on* $\mathcal{C}$*.*

Example 4. The contract $\mathcal{C}$ of Ex. 3 admits an agreement. The winning strategies for A and B are, respectively:

$$\Sigma_{\mathsf{A}}(\sigma) = \begin{cases} \{a\} & \text{if } b \in \overline{\sigma} \text{ and } a \notin \overline{\sigma} \\ \emptyset & \text{otherwise} \end{cases} \qquad \Sigma_{\mathsf{B}}(\sigma) = \begin{cases} \{b\} & \text{if } b \notin \overline{\sigma} \\ \emptyset & \text{otherwise} \end{cases}$$

For A, the only fair plays conform to Σ_{A} are ε and $\langle b\, a\rangle$. B is culpable in ε, while in $\langle b\, a\rangle$ the payoff of A is positive. For B, the only fair plays conform to Σ_{B} are $\langle b\rangle$ and $\langle b\, a\rangle$. A is culpable in $\langle b\rangle$, while in $\langle b\, a\rangle$ the payoff of B is positive. □

Example 5. The contracts in Ex. 2 above admit an agreement. The winning strategies for A and B are the eager strategies Σ^e_{A} and Σ^e_{B}, respectively. □

Example 6. Note that Σ^e_{A} is not necessarily winning for A. For instance, consider the contract with $\vdash a$, $\vdash b$, $a \# b$, $\pi^{-1}(\mathsf{A}) = \{a, b\}$, and $\Phi\mathsf{A}\sigma = 1$ iff $a \in \overline{\sigma}$. We have that $\Sigma^e_{\mathsf{A}}(\varepsilon) = \{a, b\}$, but A is losing in the fair play $\sigma = \langle b\rangle$. However, A agrees on $\mathcal{C}$, because the strategy $(\lambda\sigma.\,\text{if } \overline{\sigma} \xrightarrow{a} \text{ then } \{a\} \text{ else } \emptyset)$ is winning for A.

Example 7. The union of two winning strategies is not necessarily a winning strategy. For instance, consider the contract with enablings $\vdash a$, $\vdash b$, $\{a\} \vdash a'$, $\{b\} \vdash b'$, and conflicts $a \# b'$, $a' \# b$ (all the events are of participant A). Let:

$$\Sigma_a(\sigma) = \begin{cases} \{a\} & \text{if } \overline{\sigma} \xrightarrow{a} \\ \{a'\} & \text{if } \overline{\sigma} \xrightarrow{a'} \\ \emptyset & \text{otherwise} \end{cases} \qquad \Sigma_b(\sigma) = \begin{cases} \{b\} & \text{if } \overline{\sigma} \xrightarrow{b} \\ \{b'\} & \text{if } \overline{\sigma} \xrightarrow{b'} \\ \emptyset & \text{otherwise} \end{cases}$$

and let $\Phi\mathsf{A}\sigma$ be positive if either $a, a' \in \overline{\sigma}$, or $b, b' \in \overline{\sigma}$. Both Σ_a and Σ_b are winning strategies for A in $\mathcal{C}$, but their union $\Sigma = \lambda\sigma.\Sigma_a(\sigma) \cup \Sigma_b(\sigma)$ is not. Indeed, $\Sigma(a) = \{a', b\}$, and so $\sigma = \langle a\, b\rangle$ is a fair play w.r.t. Σ such that $\Phi\mathsf{A}\sigma \leq 0$.

We now define the composition $\sqcup$ of a set of strategies. Unlike for the union of winning strategies, their $\sqcup$-composition is guaranteed to be winning (Lemma 3).

Definition 13. *For a set of strategies $\mathcal{S}$, we define the strategy $\bigsqcup \mathcal{S}$ as:*

$$(\bigsqcup \mathcal{S})(\sigma) \;=\; \bigcup \{\Sigma(\sigma) \mid \Sigma \in \mathcal{S} \;\wedge\; \sigma \text{ conforms to } \Sigma\}$$

Lemma 3. *Let $\mathcal{S} = \{\Sigma_1, \ldots, \Sigma_n\}$ be a set of strategies. Then:*

(a) A play σ conforms to $\bigsqcup \mathcal{S}$ iff σ conforms to Σ_i, for some i.
(b) If all Σ_i are winning for A in $\mathcal{C}$, then $\bigsqcup \mathcal{S}$ is a winning strategy for A in $\mathcal{C}$.

The following lemma gives a necessary condition for reaching an agreement on a contract with O-R payoffs. The ES must have a configuration containing at least a request set, and all the offers are matched by the respective requests.

Lemma 4. *Let $\mathcal{C} = \langle \mathcal{E}, \mathcal{A}, \pi, \Phi \rangle$ be a contract with O-R payoff $\Phi\mathsf{A} = \lambda\sigma.\ \phi(\overline{\sigma})$ for A. If A agrees on $\mathcal{C}$, then there exists $C \in \mathcal{F}_\mathcal{E}$ such that $\phi(C) > 0$.*

The following theorem establishes a sufficient condition for reaching agreements in conflict-free contracts with O-R payoffs. If there exists a configuration C in $\mathcal{C}$ which contains all the requests of A, then A agrees on $\mathcal{C}$. Since the ES of $\mathcal{C}$ is conflict-free, if the strategy of A prescribes to do all her enabled events in C, then the other participants are obliged to do their events in C. Eventually, either some participant $\mathsf{B} \neq \mathsf{A}$ is culpable, or a state is reached where the payoff of A is positive.

Theorem 1. *Let $\mathcal{C}$ be a contract with O-R payoff for A. If $\mathcal{E}$ is conflict-free and $\bigcup_i R^i_\mathsf{A} \subseteq C$ for some $C \in \mathcal{F}_\mathcal{E}$, then A agrees on $\mathcal{C}$.*

Example 8. Note that conflict-freeness is necessary in Theorem 1. Consider e.g. the contract $\mathcal{C}$ with O-R payoff for A given by $O^0_\mathsf{A} = \{a\}$ and $R^0_\mathsf{A} = \{b\}$. Assume that $\pi^{-1}(\mathsf{A}) = \{a\}$, and $\pi^{-1}(\mathsf{B}) = \{b, b'\}$. The ES of $\mathcal{C}$ has enablings $\vdash a$, $\vdash b$, $\vdash b'$, and conflict $b \# b'$. The set $\{a, b\}$ is a configuration, but A does not agree on $\mathcal{C}$. Indeed, either A does no events, or she performs a. In the first case, A is culpable, while in the second one she will have a negative payoff if B does b'. □

2.4 Protection

In contract-oriented interactions [3], mutually distrusted participants advertise their contracts to a contract broker. The broker composes contracts which admit an agreement, and then establishes a session among the participants involved in such contracts. When a participant agrees on a contract, she is guaranteed that — even in the presence of malicious participants — no interaction driven by the contract will ever go wrong. At worst, if A does not reach her objectives, then some other participant will be found culpable of an infringement.

This model of interaction works fine under the hypothesis that contract brokers are honest, i.e. they never establish a session in the absence of an agreement

among all the participants. Suppose Alice is willing to lend her airplane in exchange of Bob's bike. In her contract, she could promise to lend the airplane (unconditionally), and declare that her objective is to obtain the bike. A malicious contract broker could construct an attack by establishing a session between Alice and Mallory, whose contract just says to take the airplane and give nothing in exchange. Mallory is *not* culpable, because her contract declares no obligations, and so Alice loses.

Formally, a contract $\mathcal{C}_\mathsf{A}$ *protects* A if, whatever contract $\mathcal{C}$ is composed with $\mathcal{C}_\mathsf{A}$, A has a way to non-lose in the composed contract.

Definition 14 (Protection). *A contract* $\mathcal{C}_\mathsf{A}$ protects *participant* A *if and only if, for all contracts* $\mathcal{C}$ *compatible with* $\mathcal{C}_\mathsf{A}$, A *has a non-losing strategy in* $\mathcal{C}_\mathsf{A} \mid \mathcal{C}$.

Notice that if A agrees with $\mathcal{C}$, then not necessarily $\mathcal{C}$ protects A. For instance, Mallory could join $\mathcal{C}$ with her contract $\mathcal{C}_\mathsf{M}$, and prevent Alice from borrowing Bob's bike in $\mathcal{C} \mid \mathcal{C}_\mathsf{M}$. A sufficient (yet hardly realistic) criterion for protection is to declare nonnegative payoffs for all σ. Less trivially, the following example shows a contract with possible negative payoffs which still offers protection.

Example 9. The contract $\mathcal{C}_\mathsf{B}$ of Ex. 1 does *not* protect Bob. To prove that, consider e.g. the attacker contract $\mathcal{C}' = \langle \mathcal{E}', \mathcal{A}, \pi, \Phi_{\mathcal{C}'} \rangle$, where $\mathcal{A}$ and π are as in Ex. 1, while we define $\mathcal{E}'$ with no enablings, and $\Phi_{\mathcal{C}'}$ is immaterial except for being undefined on B (otherwise $\mathcal{C}'$ and $\mathcal{C}_\mathsf{B}$ are not compatible). Consider then the contract $\mathcal{C}' \mid \mathcal{C}_\mathsf{B}$. There are only two possible strategies for B:

$$\Sigma_\mathsf{B} = \lambda\sigma.\,\emptyset \qquad\qquad \Sigma'_\mathsf{B} = \lambda\sigma.\begin{cases}\{b\} & \text{if } b \notin \overline{\sigma} \\ \emptyset & \text{otherwise}\end{cases}$$

The strategy Σ_B is losing for B, because B is not innocent under Σ_B. The strategy Σ'_B is losing as well, because in the fair play $\sigma = \langle b \rangle$ we have that $\Phi\mathsf{B}\sigma = -1$, but no participant is culpable in σ. By Def. 14, B is not protected by $\mathcal{C}_\mathsf{B}$.

On the other hand, the contract $\mathcal{C}_\mathsf{A}$ protects Alice. To show that, consider a contract $\mathcal{C}$ compatible with $\mathcal{C}_\mathsf{A}$. Let Σ_A be the following strategy for A:

$$\Sigma_\mathsf{A} = \lambda\sigma.\begin{cases}\{a\} & \text{if } b \in \overline{\sigma} \text{ and } a \notin \overline{\sigma} \\ \emptyset & \text{otherwise}\end{cases}$$

Let σ be a fair play in $\mathcal{C} \mid \mathcal{C}_\mathsf{A}$ conforming to Σ_A. There are two cases:

- $b \in \overline{\sigma}$. Then, since σ is fair, by definition of Σ_A there must exist i such that $a \in \overline{\sigma_i}$, and so A is innocent in σ. Furthermore, we have that $\Phi\mathsf{A}\sigma = 1$.
- $b \notin \overline{\sigma}$. By definition of $\mathcal{C}_\mathsf{A}$, A is not culpable in σ. Also, since $b \notin \overline{\sigma}$ and $a \notin \overline{\sigma}$, then $\Phi\mathsf{A}\sigma = 0$.

In both cases, Σ_A is non-losing for A. Therefore, $\mathcal{C}_\mathsf{A}$ protects A. □

The following theorem establishes sufficient conditions for protection in contracts with O-R payoffs. Essentially, A is protected if, whenever she enables an offer O^i_A, the corresponding request R^i_A has been already satisfied.

Theorem 2. *A contract* $\mathcal{C}_\mathsf{A} = \langle \mathcal{E}, \mathcal{A}, \pi, \Phi \rangle$ *with O-R payoffs for* A *protects* A *if* $\forall i, Y.\ Y \vdash O^i_\mathsf{A} \implies R^i_\mathsf{A} \subseteq Y$.

We now consider a relevant subclass of Offer-Request payoffs, where the requests of all participants mutually depend on their offers. An O-R payoff is *circular* when it is not possible to satisfy requests from all participants without each participant doing some offer (item (3)), and each combination of the requests is covered by a set of offers (item (4)). For instance, the payoffs of Alice and Bob in Ex. 1 are circular, because their requests (e.g. a and b, respectively) match exactly their offers.

Definition 15. *An O-R payoff* Φ *for participants* $\mathcal{A}$ *is* circular *when:*

$$\forall \mathcal{J} : \mathcal{A} \to \mathbb{N}.\ \exists \mathcal{L} : \mathcal{A} \to \mathbb{N}.\ \bigcup_{\mathsf{A} \in \mathcal{A}} O^{\mathcal{L}\mathsf{A}}_\mathsf{A} \subseteq \bigcup_{\mathsf{A} \in \mathcal{A}} R^{\mathcal{J}\mathsf{A}}_\mathsf{A} \tag{3}$$

$$\forall \mathcal{J} : \mathcal{A} \to \mathbb{N}.\ \exists \mathcal{L} : \mathcal{A} \to \mathbb{N}.\ \bigcup_{\mathsf{A} \in \mathcal{A}} O^{\mathcal{L}\mathsf{A}}_\mathsf{A} \supseteq \bigcup_{\mathsf{A} \in \mathcal{A}} R^{\mathcal{J}\mathsf{A}}_\mathsf{A} \tag{4}$$

Example 10 (Dining retailers [5]). Around a table, n cutlery retailers are about to have dinner. At the center of the table, there is a large dish of food. Despite the food being delicious, the retailers cannot start eating right now. To do that, and follow the proper etiquette, each retailer needs a complete cutlery set, consisting of n pieces of different kinds. Each of the n retailers owns a distinct set of n piece of cutlery, all of the same kind. The retailers start discussing about trading their cutlery, so that they can finally eat.

We formalise this scenario as follows. Each retailer A_i initially owns n pieces of kind i. For all $j \neq i$, the event $e_{i,j}$ models A_i giving a piece of cutlery to retailer A_j. Thus, $\pi^{-1}(\mathsf{A}_i) = \{e_{i,j} \mid j \neq i\}$. Retailer A_i offers $n-1$ pieces of his cutlery of kind i in exchange for $n-1$ pieces of cutlery of the other kinds.

$$O_i = \{e_{i,j} \mid j \neq i\} \qquad\qquad R_i = \{e_{j,i} \mid j \neq i\}$$

By Def. 15, the payoff Φ_i of each retailer is a finite O-R circular payoff. □

Agreement and protection can coexist in contracts with *infinite* circular O-R payoffs (see Ex. 11). Intuitively, when an infinite offer O_A has to match an infinite request R_B, participants A and B may take turns in doing event in $O_\mathsf{A} \cup R_\mathsf{B}$. This strategy is winning for both participants (hence they have an agreement), and protection follows because no participant completes her offer before receiving the corresponding request.

Example 11. Let $\mathcal{C}_\mathsf{A} = \langle \mathcal{E}_\mathsf{A}, \mathcal{A}, \pi, \Phi_\mathsf{A} \rangle$ and $\mathcal{C}_\mathsf{B} = \langle \mathcal{E}_\mathsf{B}, \mathcal{A}, \pi, \Phi_\mathsf{B} \rangle$ be contracts with circular O-R payoffs (with infinite offers/requests) defined as follows:

$$O_\mathsf{A} = \{e_i \mid i \in \mathbb{N}\} = R_\mathsf{B} \qquad\qquad R_\mathsf{A} = \{\overline{e_i} \mid i \in \mathbb{N}\} = O_\mathsf{B}$$

and let $\mathcal{A} = \{\mathsf{A}, \mathsf{B}\}$, $\pi(e_i) = \mathsf{A}$, $\pi(\overline{e_i}) = \mathsf{B}$ for all $i \in \mathbb{N}$. Let the ES $\mathcal{E}_\mathsf{A}$ and $\mathcal{E}_\mathsf{B}$ be defined by the following enablings (and empty conflicts):

$$\mathcal{E}_\mathsf{A} :\ \{\vdash e_0\} \cup \{\overline{e_i} \vdash e_{i+1} \mid i \geq 0\} \qquad\qquad \mathcal{E}_\mathsf{B} :\ \{e_i \vdash \overline{e_i} \mid i \geq 0\}$$

The contract $\mathcal{C} = \mathcal{C}_\mathsf{A} \mid \mathcal{C}_\mathsf{B}$ admits an agreement. We prove separately that A and B agree on $\mathcal{C}$. Let Σ^e_A be the eager strategy for A. Let σ be a fair play of $\mathcal{C}$ conform to Σ^e_A. We prove that A wins in σ. By Lemma 2, the strategy Σ^e_A makes A innocent in σ. There are two subcases. If B is not innocent in σ, then A wins. Otherwise, the play σ must be infinite, i.e. $\overline{\sigma} = \{e_i\}_{i\in\mathbb{N}} \cup \{\overline{e_i}\}_{i\in\mathbb{N}}$. Therefore, $R_\mathsf{A} \subseteq \overline{\sigma}$, and so A wins. To prove that B has a winning strategy in $\mathcal{C}$ we proceed similarly, by choosing the eager strategy Σ^e_B for B.

We now show that $\mathcal{C}_\mathsf{A}$ protects A. Let $\mathcal{C}'$ be compatible with $\mathcal{C}_\mathsf{A}$. The eager strategy Σ^e_A is non-losing for A. Indeed, in every fair play σ conform to Σ^e_A, if there exists $\overline{e_i} \in R_\mathsf{A} \not\subseteq \overline{\sigma}$ then $e_{i+1} \in O_\mathsf{A} \notin \overline{\sigma}$, and so $\Phi\mathsf{A}\sigma \geq 0$. To prove that $\mathcal{C}_\mathsf{B}$ protects B, we proceed similarly, by choosing the eager strategy Σ^e_B for B. □

A remarkable feature of *finite* circular payoffs is that, in each play where all participants win, at some point there exists a participant A which has performed all the offers in O^i_A before having obtained all the requests in R^i_A. Intuitively, the participant A which makes this "first step" is not protected. The proof technique exploited by Lemma 5 is similar to that used in [11] to prove that fair exchange is impossible without a trusted third party.

Lemma 5. *Let $\mathcal{C}$ be a contract with finite circular O-R payoffs. If σ is a winning play for all participants in $\mathcal{A}$, then there exists a prefix η of σ and a participant $\mathsf{A} \in \mathcal{A}$ such that $\Phi\mathsf{A}\eta < 0$.*

Our main result in this section is Theorem 3 below. It states that if a set of contracts with finite circular O-R payoffs admits an agreement, then some of the participants are not protected, and *vice versa*.

Theorem 3. *Let $\mathcal{C}_1, \ldots, \mathcal{C}_n$ be contracts with circular finite O-R payoffs for $\mathsf{A}_1, \ldots, \mathsf{A}_n$, respectively. Then, at most one of the following statements is true:*

(a) $\mathcal{C}_1 \mid \cdots \mid \mathcal{C}_n$ admits an agreement;
(b) for all $i \in 1..n$, $\mathcal{C}_i$ protects A_i.

3 Reconciling Agreement with Protection

In the previous section we have shown that agreement and protection cannot coexist in a relevent class of contracts (Theorem 3). As made evident by Theorem 2, to protect herself A must obtain all her requests R^i_A before doing all her offers O^i_A. If all participants adhere to this principle, agreement is not possible. For instance, Alice and Bob in Ex. 1 would be protected by contracts with enablings $a \vdash b$ and $b \vdash a$, but no agreement would be possible because nobody risks doing the first step.

To reconcile agreements with protection, A could relax her contract, i.e. she could do a in change of the *promise* of B to do b. In this case A can safely do the first step, because either B does b, or he will be culpable of a contract violation.

To model this kind of "conditional" enabling, we propose an extension of Winskel's event structures with a new *circular causality* relation ($\Vdash$). The enabling $b \Vdash a$ (intuitively, "I will do a if you *promise* to do b") together with the

other prescription $a \Vdash b$ has a configuration where both a and b have happened, despite of the circular dependencies. We call our extension *ES with circular causality* (CES in short).

In Sect. 3.1 we introduce CES and we state some basic properties. In Sect. 3.2 we reformulate our theory of contracts by using CES in place of ES. Finally, in Sect. 3.3 we show how CES allow for reconciling agreement with protection.

3.1 Event Structures with Circular Causality

Definition 16. *An event structure with circular causality is an* ES *enriched with a (saturated)* circular enabling *relation* $\Vdash \subseteq \{X \subseteq_{fin} E \mid CF(X)\} \times E$.

We conservatively extend the notion of configuration in [21] to deal with circular causality. Intuitively, for all events e_i in the sequence $\langle e_0 \cdots e_n \rangle$, e_i can either be $\vdash$-enabled by its predecessors, or $\Vdash$-enabled by the *whole* sequence. Note that if C is a finite configuration, and $\{e_0 \dots e_n\}$ is an enumeration of C which satisfies all the enablings, not necessarily $\{e_0 \dots e_{n-1}\}$ is a configuration as well (see *e.g.*, $\mathcal{E}_2$ in Fig. 1). To reason compositionally about configurations, Def. 17 defines a slightly more general notion of configurations.

In an *X-configuration* C, the set C can contain an event e even in the absence of a justification of e through a standard/circular enabling — provided that e belongs to X. This allows, given an X-configuration, to add/remove any event and obtain an Y-configuration, possibly with $Y \neq X$. We shall say that the events in X have been taken "on credit", to remark the fact that they may have been performed in the absence of a causal justification. Configurations (i.e., $\emptyset$-configurations) play a crucial role, as they represent sets of events where all the credits have been honoured.

Definition 17 (Configuration). *Let $\mathcal{E} = (E, \#, \vdash, \Vdash)$ be a* CES. *A conflict-free sequence $\sigma = \langle e_0 \dots e_n \rangle \in E^*$ without repetitions is an X-*trace *of $\mathcal{E}$ iff:*

$$\forall i \leq n.\ (e_i \in X \ \ \vee \ \ \overline{\sigma_i} \vdash e_i \ \ \vee \ \ \overline{\sigma} \Vdash e_i)$$

*For all $C, X \subseteq E$ we say that C is an X-*configuration *of $\mathcal{E}$ iff $CF(C)$ and:*

$$\forall e \in C.\ \exists \sigma\ X\text{-trace.}\ e \in \overline{\sigma} \subseteq C$$

The set of all X-configurations of $\mathcal{E}$ is denoted by $\mathcal{F}_{\mathcal{E}}(X)$, or just $\mathcal{F}_{\mathcal{E}}$ when $X = \emptyset$.

Example 12. Consider the four CES in Fig. 1.

(1) $\mathcal{E}_1$ has enablings $\emptyset \vdash a$, $\emptyset \Vdash b$, and conflict $a\#b$. By Def. 17 we have $\emptyset, \{a\}, \{b\} \in \mathcal{F}_{\mathcal{E}_1}$, but $\{a, b\} \notin \mathcal{F}_{\mathcal{E}_1}$.
(2) $\mathcal{E}_2$ has enablings $\{a\} \vdash b$ and $\{b\} \Vdash a$. Here $\emptyset, \{a, b\} \in \mathcal{F}_{\mathcal{E}_2}$, $\{b\} \in \mathcal{F}_{\mathcal{E}_2}(\{b\})$ and $\{a\} \in \mathcal{F}_{\mathcal{E}_2}(\{a\})$, while neither $\{a\}$ nor $\{b\}$ belong to $\mathcal{F}_{\mathcal{E}_2}(\emptyset)$.
(3) $\mathcal{E}_3$ has enablings $\{a\} \vdash b$, $\{a\} \vdash c$, $\{b\} \Vdash a$, $\{c\} \vdash a$, and conflict $b\#c$. The only non-empty configuration of $\mathcal{E}_3$ is $\{a, b\}$.

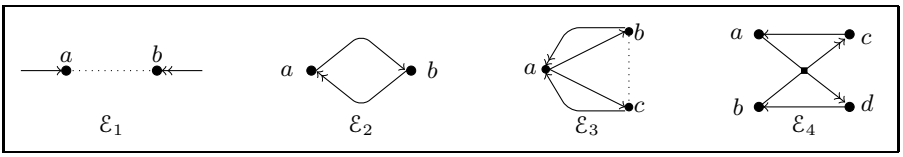

Fig. 1. CES are denoted as directed hypergraphs, where nodes stand for events. An hyperedge from a set of nodes X to node e denotes an enabling $X \circ e$, where $\circ = \vdash$ if the edge has a single arrow, and $\circ = \Vdash$ if the edge has a double arrow. A conflict $a\#b$ is represented by a dotted line between a and b.

(4) $\mathcal{E}_4$ has enablings $\{a,b\} \Vdash c$, $\{a,b\} \Vdash d$, $\{c\} \vdash a$, and $\{d\} \vdash b$. We have that $\{a,b,c,d\} \in \mathcal{F}_{\mathcal{E}_4}$. Note that, were one (or both) of the $\Vdash$ turned into a $\vdash$, then the only configuration would have been the empty one. □

According to Winskel's axiom of finite causes, all events in a configuration (except those taken on credit) have a finite justification. Thus, an event cannot be justified through an infinite chain of events, i.e. in the CES with enablings $\{e_{i+1}\} \Vdash e_i$ for all $i \geq 0$, the set $\{e_i \mid i \geq 0\}$ is *not* a configuration.

The configurations of CES do still enjoy the finiteness and coherence properties of classical ES, though they are not coincidence-free, which is correct from our point of view because of the presence of circular dependencies. A subset A of a set F is *pairwise compatible* iff $\forall e, e' \in \bigcup A.\ \exists C \in F.\ e, e' \in C$.

Theorem 4. *For all* CES $\mathcal{E}$*, and for all* $X \subseteq E$*, the set* $\mathcal{F}_{\mathcal{E}}(X)$ *satisfies:*

- *(Coherence) If* F *is a pairwise compatible subset of* $\mathcal{F}_{\mathcal{E}}(X)$*, then* $\bigcup F \in \mathcal{F}$.
- *(Finiteness)* $\forall C \in \mathcal{F}.\ \forall e \in C.\ \exists C_0 \in \mathcal{F}.\ e \in C_0 \subseteq_{fin} C$

We define below an operational semantics of CES. This is given in terms of an LTS, the states of which are pairs (C, X). The first element is the set of events occurred so far; the second element is a set of events taken "on credit".

Definition 18 (LTS of a CES). *For a* CES $\mathcal{E}$*, we define* $\mathrm{LTS}_{\mathcal{E}} = \langle S, E, \rightarrow_{\mathcal{E}} \rangle$*, where* $S = \wp_{fin}(E) \times \wp_{fin}(E)$*, and* $\rightarrow_{\mathcal{E}}$ *is defined by the following rule:*

$$\frac{e \notin C \qquad CF(C \cup \{e\})}{(C, X) \xrightarrow{e}_{\mathcal{E}} (C \cup \{e\},\ \Delta(C, X, e))}$$

where for all $C, X \subseteq E$ *and for all* $e \in E$*, we define:*

$$\Delta(C, X, e) \ = \ (X \setminus \{x \in X \mid C \cup \{e\} \Vdash x\}) \cup \begin{cases} \{e\} & \text{if } C \cup \{e\} \not\Vdash e \ \wedge \ C \not\vdash e \\ \emptyset & \text{otherwise} \end{cases}$$

The set $\Delta(C, X, e)$ defines how credits change when firing e in a play where the current credits are X, and the events C have already been performed.

The following theorem relates traces of $\mathrm{LTS}_{\mathcal{E}}$ to configurations of $\mathcal{E}$.

Theorem 5. *For all* CES $\mathcal{E}$*, for all* $C, X \subseteq E$*:*

$$C \in \mathcal{F}(X) \iff \forall D \subseteq_{fin} C.\ \exists X_0 \subseteq X.\ \exists C_0.\ D \subseteq C_0 \subseteq C \ \wedge \ (\emptyset, \emptyset) \rightarrow^* (C_0, X_0)$$

3.2 Agreement in CES-Based Contracts

In this section we conservatively extend the contract theory of Sect. 2, by allowing the component $\mathcal{E}$ of a contract to be a CES.

By Def. 18, a conflict-free sequence $\langle e_0\, e_1 \cdots \rangle$ without repetitions uniquely identifies a trace $(\emptyset,\emptyset) \xrightarrow{e_0}_{\mathcal{E}} (C_1, X_1) \xrightarrow{e_1}_{\mathcal{E}} \cdots$ in $\mathrm{LTS}_{\mathcal{E}}$. We denote with (C_k^σ, X_k^σ) the state of $\mathrm{LTS}_{\mathcal{E}}$ reached after k steps of the sequence σ. A *play* of a contract $\mathcal{C}$ is a (finite or infinite) sequence σ of events such that $(\emptyset,\emptyset) \xrightarrow{\sigma}_{\mathcal{E}}$. The notions of *strategy* and *conformance* to a strategy are as in Sect. 2.

The key difference between ES-based and CES-based contracts is the notion of innocence. In the ES-based model, a participant A is culpable in a play σ when some event e of A is enabled in σ. Here, in addition to enabled events, we consider obligations those events which can be done "on credit", under the guarantee that they will be eventually honoured, whatever events are done later on by the other participants. These events are said *prudent*. The definition of prudent strategies and of innocent participants is mutually coinductive. A participant A is innocent in σ when she has no persistently prudent events. Hence, if the strategy of A tells to do all her prudent events, then in all fair plays these events must either become imprudent, or be fired, or be conflicted. Formally (although a bit counter-intuitively), fired and conflicted events are imprudent: therefore, A is innocent when all her prudent events eventually become imprudent.

Definition 19 (Prudence). *A strategy Σ for A is* prudent *if, for all finite plays σ, for all $e \in \Sigma(\sigma)$ such that $\sigma\, e$ conforms to Σ, and for all fair plays $\sigma' = \sigma e \eta$ conform to Σ where all $\mathsf{B} \neq \mathsf{A}$ are innocent,*

$$\exists k > |\sigma|.\ X_k^{\sigma'} \cap \pi^{-1}(\mathsf{A}) \subseteq X_{|\sigma|}^{\sigma'}$$

An event e is prudent *in σ if there exists a prudent strategy Σ such that σ conforms to Σ and $e \in \Sigma(\sigma)$.*
A participant A is innocent *in $\sigma = \langle e_0\, e_1 \cdots \rangle$ iff:*

$$\forall e \in \pi^{-1}(\mathsf{A}).\ \forall i \geq 0.\ \exists j \geq i.\ e \text{ is imprudent in } \sigma_j$$

Example 13. Recall the CES from Ex. 12. In $\mathcal{E}_1$, both a and b are prudent in the empty play ε, because they are enabled in $\emptyset$. In $\mathcal{E}_2$, a is prudent in ε, while b is *not* prudent in ε, yet it is prudent in $\langle a \rangle$. Now consider the CES $\mathcal{E}_3$, and assume that $\pi(a) = \mathsf{A}$ and $\pi(b) = \pi(c) = \mathsf{B}$. We have that a is *not* prudent in ε, because if B chooses to do c, then the credit a can no longer be honoured. Instead, both b and c are prudent in $\langle a \rangle$.

Notice that, after Lemma 6 below, the new definition of innocence conservatively extends that in Def. 10. That is, an event enabled in σ is prudent.

Lemma 6. *For all σ and $e \notin \overline{\sigma}$, if $\overline{\sigma} \vdash e$ or $\overline{\sigma} \cup \{e\} \Vdash e$ then e is prudent in σ.*

Lemma 7. *Let $\mathcal{S}$ be a finite set of prudent strategies. Then, $\bigsqcup \mathcal{S}$ is prudent.*

Lemma 8. *For a contract $\mathcal{C} = \langle \mathcal{E}, \cdots \rangle$, where $\mathcal{E}$ is a finite* CES*, the strategy $\Sigma_{\mathsf{A}}^p = \lambda\sigma.\, \{e \in \pi^{-1}(\mathsf{A}) \mid e \text{ is prudent in } \sigma\}$ is the greatest prudent strategy for A.*

Lemma 9. *Σ^p_{A} is an innocent strategy for A.*

For a CES $\mathcal{E}$ and a set of events C, we say that e is *reachable from* C iff there exists η such that $e \in \overline{\eta}$ and $(C, \emptyset) \xrightarrow{\eta} (C', \emptyset)$. Theorem 6 states that in conflict-free CES which have only circular enablings, the set of prudent events in σ coincides with the set $\mathcal{R}^{\overline{\sigma}}_{\mathcal{E}}$ of events reachable from $\overline{\sigma}$.

Theorem 6. *If $\mathcal{E}$ is conflict-free and $\vdash$-free, then for all plays σ of $\mathcal{C} = \langle \mathcal{E}, \cdots \rangle$:*

$$e \in \mathcal{R}^{\overline{\sigma}}_{\mathcal{E}} \iff e \text{ prudent in } \sigma$$

We now refine the notion of winning strategy given in Def. 11. The items are similar to the corresponding items in Def. 11, except that the definitions of innocence now takes into account the events performed on credit.

Definition 20 (Winning play). *Define the function $\widetilde{\mathcal{W}}$ as follows:*

$$\widetilde{\mathcal{W}}\mathsf{A}\sigma = \begin{cases} \Phi\mathsf{A}\sigma & \mathsf{A} \text{ is credit-free and all participants are innocent in } \sigma \\ +1 & \text{if } \mathsf{A} \text{ is innocent, and some } \mathsf{B} \neq \mathsf{A} \text{ is culpable in } \sigma \\ -1 & \text{otherwise} \end{cases}$$

where we say that A *is* credit-free *in σ iff*

$$\forall e \in \pi^{-1}(\mathsf{A}).\ \forall i \geq 0.\ \exists j \geq i.\ e \notin X^{\sigma}_j$$

The notions of winning/losing play/strategy, agreement and protection are the same as in Sect. 2, except that $\widetilde{\mathcal{W}}$ is now used in place of $\mathcal{W}$.

Lemma 10. *Let Σ_{A} be a prudent strategy for A. For all fair plays σ conform to Σ_{A}, either A is credit-free in σ, or some $\mathsf{B} \neq \mathsf{A}$ is culpable in σ.*

Example 14. In Ex. 9 we have shown that the contract $\mathcal{C}_{\mathsf{A}}$ protects Alice, while $\mathcal{C}_{\mathsf{B}}$ does not protect Bob. Suppose now to change Bob's contract into a contract $\mathcal{C}'_{\mathsf{B}}$ where Bob relaxes his requirements. The contract $\mathcal{C}'_{\mathsf{B}}$ differs from $\mathcal{C}_{\mathsf{B}}$ only in the event structure $\mathcal{E}'_{\mathsf{B}}$, which contains exactly one circular enabling: $\{a\} \Vdash b$. Similarly to Ex. 4, the contract $\mathcal{C}_{\mathsf{A}} \mid \mathcal{C}'_{\mathsf{B}}$ admits an agreement. To show that, let Σ_{A} and Σ_{B} be the following strategies for A and B, respectively:

$$\Sigma_{\mathsf{A}}(\sigma) = \begin{cases} \{a\} & \text{if } b \in \overline{\sigma} \text{ and } a \notin \overline{\sigma} \\ \emptyset & \text{otherwise} \end{cases} \qquad \Sigma_{\mathsf{B}}(\sigma) = \begin{cases} \{b\} & \text{if } b \notin \overline{\sigma} \\ \emptyset & \text{otherwise} \end{cases}$$

Roughly, the only fair play which conforms to Σ_{A} and Σ_{B} where both A and B are innocent is $\sigma = \langle ba \rangle$, which gives rise to the following trace in $\mathrm{LTS}_{\mathcal{E}}$:

$$(\emptyset, \emptyset) \xrightarrow{b} (\{b\}, \{b\}) \xrightarrow{a} (\{a, b\}, \emptyset)$$

We have that A and B win in σ, because $\widetilde{\mathcal{W}}\mathsf{A}\sigma = 1 = \widetilde{\mathcal{W}}\mathsf{B}\sigma$. Thus, Σ_{A} and Σ_{B} are winning strategies for A and B, respectively, and so $\mathcal{C}$ admits an agreement. □

3.3 Protection in CES-Based Contracts

In this section we show that CES-based contracts allow for both agreements and protection in contracts with circular finite O-R payoffs. Before presenting the formal results, we give some intuition through our working example.

Example 15. Differently from the contract $\mathcal{C}_\mathsf{B}$ in Ex. 9, the contract $\mathcal{C}'_\mathsf{B}$ in Ex. 14 protects Bob. Let $\mathcal{C}'$ be a contract compatible with $\mathcal{C}'_\mathsf{B}$. Consider the strategy Σ^p_B for B, as defined in Lemma 9. Let ν be a fair play of $\mathcal{C}'_\mathsf{B} \mid \mathcal{C}'$ conform to Σ^p_B. By contradiction, assume that B loses in ν. By Lemma 9, B is innocent in ν, and so it must be $\Phi\mathsf{B}\nu < 0$. By definition, the payoff of B is negative only when $b \in \overline{\nu}$ and $a \notin \overline{\nu}$. Assume that $\nu = \eta\, b\, \eta'$. By definition of Σ^p_B, the event b was prudent in η, and we have the transition $(\overline{\eta}, X^0) \xrightarrow{b} (\overline{\eta} \cup \{b\}, X^0 \cup \{b\})$. After B has performed b, its only strategy is the empty one. By Def. 19, for all plays $e_0 e_1 \cdots$ starting from $(\overline{\eta} \cup \{b\}, X^0 \cup \{b\})$, there exists some $k > 0$ such that $b \notin X^k$. This means that b has been honoured, and the only way to do that is to perform a. Therefore, $a \in \overline{\nu}$ — contradiction. □

We now construct a CES from an O-R payoff with finite responses. For all clauses (O, R), the CES contains the enablings $R \Vdash O$. Lemma 11 below reveals a key feature of circularity: the CES obtained from a circular O-R payoff has a configuration which comprises all the responses of all participants. Together with Theorem 7, this will allow for constructing a contract which admits an agreement. Theorems 7 and 8 are the CES counterpart of Theorems 1 and 2 for ES-based contracts, respectively.

Definition 21. *For an O-R payoff Φ with clauses $(O^i, R^i)_i$ and finite R^i, define $\mathcal{E}(\Phi)$ as the conflict-free* CES *with (saturated) enablings $\{R^i \Vdash O^i\}_i$.*

Lemma 11. *Let Φ be a finite circular O-R payoff for $\mathcal{A}$ such that $\Phi\mathsf{A} = \lambda\sigma.\, \phi_\mathsf{A}\overline{\sigma}$ for all $\mathsf{A} \in \mathcal{A}$. Then, $\exists C \in \mathcal{F}_{\mathcal{E}(\Phi)}.\ \forall \mathsf{A} \in \mathcal{A}.\ \bigcup_i R^i_\mathsf{A} \subseteq C$.*

Theorem 7. *Let $\mathcal{C}$ be a contract with O-R payoff for A. If $\mathcal{E}$ is conflict-free and $\vdash$-free, and $\bigcup_i R^i_\mathsf{A} \subseteq C$ for some $C \in \mathcal{F}_\mathcal{E}$, then A agrees on $\mathcal{C}$.*

Theorem 8. *For a finite* CES *$\mathcal{E}$ and an O-R payoffs Φ for A, the contract $\langle \mathcal{E}, \mathcal{A}, \pi, \Phi\rangle$ protects A if: $\forall i, Y.\ (\forall e \in O^i_\mathsf{A}.\ Y \vdash e \ \vee\ Y \Vdash e) \implies R^i_\mathsf{A} \subseteq Y$.*

Theorem 9 below states that agreements and protection can coexist in CES-based contracts with circular finite O-R payoffs. Recall that Theorem 3 excluded this possibility for ES-based contracts. Condition (5) in Theorem 9 is technical, yet it makes the theorem applicable to a broad class of contracts with O-R payoffs (e.g. the dining retailers scenario, see Ex. 17). Ex. 16 shows that when condition (5) is not satisfied, Theorem 9 does not hold in general.

Theorem 9. *Let $\Phi_1, \ldots, \Phi_n$ be finite circular O-R payoffs for $\mathsf{A}_1, \ldots, \mathsf{A}_n$, respectively, and such that, for all $\mathsf{A} \in \{\mathsf{A}_1, \ldots, \mathsf{A}_n\}$:*

$$\forall P \subseteq \mathbb{N}.\ \forall j.\ O^j_\mathsf{A} \subseteq \textstyle\bigcup_{i\in P} O^i_\mathsf{A} \implies R^j_\mathsf{A} \subseteq \bigcup_{i\in P} R^i_\mathsf{A} \tag{5}$$

Then, there exist contracts $\mathcal{C}_i = \langle \mathcal{E}_i, \mathcal{A}, \pi, \Phi_i\rangle$ for $i \in 1..n$ such that:

(a) $\mathcal{C}_1 \mid \cdots \mid \mathcal{C}_n$ admits an agreement;
(b) for all $i \in 1..n$, $\mathcal{C}_i$ protects A_i.
(c) for all plays σ of $\mathcal{C}_1 \mid \cdots \mid \mathcal{C}_n$, $\forall e \in \overline{\sigma}.\ \exists i.\ e \in O^i_{\pi(e)}$.

Example 16. Consider the O-R payoff Φ_{A} of participant A defined by:

$$O^0 = \{a_0, a_1\} \qquad O^1 = \{a_1, a_2\} \qquad O^2 = \{a_0, a_2\}$$
$$R^0 = \{b_0\} \qquad R^1 = \{b_1\} \qquad R^2 = \{b_0, b_1\}$$

Condition (5) of Theorem 9 is satisfied, hence the contract with CES $\mathcal{E}(\Phi_{\mathsf{A}})$ protects A, and allows A to reach an agreement with other participants whenever the overall payoff satisfies the conditions of the theorem.

Suppose now to change Φ_{A}, by requiring $R^2 = \{b_2\}$. Notice that such modified payoff no longer satisfies condition (5). Indeed, by choosing $P = \{0, 1\}$ and $j = 2$ we have that $\{a_0, a_2\} = O^2 \subseteq O^0 \cup O^1$, but $\{b_2\} = R^2 \not\subseteq R^0 \cup R^1$. So, Theorem 9 does not apply. The CES $\mathcal{E}(\Phi_{\mathsf{A}})$ contains the enablings $\{b_0\} \Vdash \{a_0, a_1\}$, $\{b_1\} \Vdash \{a_1, a_2\}$, and $\{b_2\} \Vdash \{a_0, a_2\}$. Now A is *not* protected. Indeed, an attacker B could perform b_0 and b_1 to oblige A to do a_0, a_1, a_2. A would lose, because to be innocent she has to do all the offers in O^2, but doing so she is not guaranteed to obtain R^2. As a matter of facts, there exists no CES which guarantees both agreement and protection for the payoff Φ_{A}. □

Example 17. Recall the dining retailers scenario from Ex. 10. The payoff Φ_i of each retailer is a finite O-R circular payoff, and condition (5) is trivially satisfied. Therefore, Theorem 9 allows for constructing contracts which admit an agreement and protects all retailers. The CES of contract $\mathcal{C}_i$ of retailer A_i has enablings $\{e_{j,i} \mid i \neq j\} \Vdash \{e_{i,j} \mid i \neq j\}$. The idea is simple: A_1 offers his pieces of cutlery, in exchange of the commitment of the other retailers to do the same. Since all retailers commit to the analogous contract, we have an agreement. □

4 Related Work and Conclusions

We have studied contracts from a foundational perspective. Our formalisation of contracts builds upon a very abstract model of concurrent computations, namely event structures, to provide general notions of agreement and protection. We expect that specific formalisations of agreement, e.g. the one in [8], can be interpreted as instances of our general notion, in the same spirit that event structures can provide semantics to more concrete models of concurrency, e.g. CCS, π-calculus and Petri nets [21].

An abstract model of contracts is fundamental for the development of the contract-oriented paradigm. In addition to the possibility of relating different formalisations of contracts, such an abstract model would also allow for reasoning uniformly about the properties of contract-based systems. For instance, the static/dynamic notions of *honesty* of a process, which in [4] were specific for the contracts of [8], could be generalised to a broader class of contracts.

Aiming at generality, we have almost neglected some relevant issues, e.g. devising efficient decision procedures for agreements. Although in the most general setting (infinite event structures, arbitrary payoff functions) we come up against the problem of undecidability, such kind of results can be obtained by considering suitable subclasses of event structures/payoff functions (e.g. model checking temporal logic on finite representations of infinite event structures, as in [18]).

A heterogeneous ecosystem of formalisms for contracts has appeared in the literature. Citing a few recent approaches, these formalisms include logics [5,19], behavioural types [6,8], Petri nets [20], multi-player games [13], domain-specific languages [16], c-semirings [7], *etc.*

Most of the existing models do not explicitly deal with the circularity issue, which instead has been a main subject of study in this paper. An exception is [5], where circularity is dealt with at a proof-theoretic level. The logic PCL presented in [5] extends propositional intuitionistic logic with a new connective, that weakens the standard implication $\to$, somehow similarly to the way our $\Vdash$ weakens the standard enabling $\vdash$. CES and PCL are strongly related: preliminary results suggest that finite conflict-free CES correspond to Horn PCL formulae.

In [2] some preliminary work on event structures with circular causality is presented. In the simplified model of [2], where event structures are finite and conflict-free, and the goals are O-R payoffs without offers, it is shown how to decide agreements through an encoding of event structures into PCL formulae.

In [14] event structures are extended with a *response* relation. A relation $a \bullet\!\!\to b$ models the fact that, whenever event a is present in a trace, then b must eventually occur after a. This is quite different from a circular enabling $a \Vdash b$, which instead does not impose any ordering between a and b (it suffices that b is honoured somehow). Also, augmenting the number of $\Vdash$-enablings increases the number of configurations, while adding more response relations reduces it.

Liability issues are the focus of [16,12]. Given a contract and an execution trace, the problem is to establish evidence about the occurrence of a contract violation, and in particular to assign blame to misbehaving participants. While [16,12] are not concerned about how an agreement is found (they just consider the contract as already agreed upon), they explore issues not explicitly modelled in our framework. The notion of liability in [16] takes into account time constraints. Extending our contract model with temporal deadlines and, more in general, with quantitative aspects (like e.g. probabilities) seems to be feasible, along the lines of analogous extensions of events structures [17].

Our model adopts a draconian notion of innocence, in that a participant omitting to perform a single due event in a play is considered culpable, regardless of the fact that the other participants could equally be satisfied with that play. Establishing finer-grained notions of causality between a violation and the resulting failure, as done e.g. in [12], seems a plausible extension of our work.

Our notion of winning play (Def. 11 and Def. 20) is a sort of lexicographic objective, similarly to those in [10]. The secure equilibria introduced in [10] require that a player cannot increase her payoff while decreasing the payoff of the other player. This is stronger than our notion of agreement, where we just require that strategies exist which yield a positive payoff for all players. Indeed, such strategies do not even have to form a Nash equilibrium.

Acknowledgments. Work partially supported by Aut. Region of Sardinia under grants L.R.7/2007 CRP2-120 (TESLA), CRP-17285 (TRICS), P.I.A. 2010 Project "Social Glue", and by MIUR PRIN 2010-11 project "Security Horizons".

References

1. Armbrust, M., et al.: A view of cloud computing. Comm. ACM 53(4), 50–58 (2010)
2. Bartoletti, M., Cimoli, T., Pinna, G.M., Zunino, R.: An event-based model for contracts. In: Proc. PLACES (2012)
3. Bartoletti, M., Tuosto, E., Zunino, R.: Contract-oriented computing in CO_2. Scientific Annals in Computer Science 22(1), 5–60 (2012)
4. Bartoletti, M., Tuosto, E., Zunino, R.: On the Realizability of Contracts in Dishonest Systems. In: Sirjani, M. (ed.) COORDINATION 2012. LNCS, vol. 7274, pp. 245–260. Springer, Heidelberg (2012)
5. Bartoletti, M., Zunino, R.: A calculus of contracting processes. In: LICS (2010)
6. Bravetti, M., Zavattaro, G.: Towards a Unifying Theory for Choreography Conformance and Contract Compliance. In: Lumpe, M., Vanderperren, W. (eds.) SC 2007. LNCS, vol. 4829, pp. 34–50. Springer, Heidelberg (2007)
7. Buscemi, M.G., Montanari, U.: CC-Pi: A Constraint-Based Language for Specifying Service Level Agreements. In: De Nicola, R. (ed.) ESOP 2007. LNCS, vol. 4421, pp. 18–32. Springer, Heidelberg (2007)
8. Castagna, G., Gesbert, N., Padovani, L.: A theory of contracts for web services. ACM Transactions on Programming Languages and Systems 31(5) (2009)
9. Chatterjee, K., Henzinger, T.A.: A survey of stochastic ω-regular games. J. Comput. Syst. Sci. 78(2), 394–413 (2012)
10. Chatterjee, K., Henzinger, T.A., Jurdzinski, M.: Games with secure equilibria. Theor. Comput. Sci. 365(1-2), 67–82 (2006)
11. Even, S., Yacobi, Y.: Relations among public key signature systems. Technical Report 175, Computer Science Department, Technion, Haifa (1980)
12. Gössler, G., Le Métayer, D., Raclet, J.-B.: Causality Analysis in Contract Violation. In: Barringer, H., Falcone, Y., Finkbeiner, B., Havelund, K., Lee, I., Pace, G., Roşu, G., Sokolsky, O., Tillmann, N. (eds.) RV 2010. LNCS, vol. 6418, pp. 270–284. Springer, Heidelberg (2010)
13. Henriksen, A.S.: Adversarial Models for Cooperative Interactions. PhD thesis, Department of Computer Science, University of Copenhagen (2011)
14. Hildebrandt, T.T., Mukkamala, R.R.: Declarative event-based workflow as distributed dynamic condition response graphs. In: Proc. PLACES (2010)
15. Honda, K., Yoshida, N., Carbone, M.: Multiparty asynchronous session types. In: POPL (2008)
16. Hvitved, T., Klaedtke, F., Zălinescu, E.: A trace-based model for multiparty contracts. JLAP 81(2), 72–98 (2012)
17. Katoen, J.-P.: Quantitative and qualitative extensions of event structures. PhD thesis, University of Twente (1996)
18. Penczek, W.: Model-Checking for a Subclass of Event Structures. In: Brinksma, E. (ed.) TACAS 1997. LNCS, vol. 1217, pp. 145–164. Springer, Heidelberg (1997)
19. Prisacariu, C., Schneider, G.: A Formal Language for Electronic Contracts. In: Bonsangue, M.M., Johnsen, E.B. (eds.) FMOODS 2007. LNCS, vol. 4468, pp. 174–189. Springer, Heidelberg (2007)
20. van der Aalst, W.M.P., Lohmann, N., Massuthe, P., Stahl, C., Wolf, K.: Multiparty contracts: Agreeing and implementing interorganizational processes. Comput. J. 53(1), 90–106 (2010)
21. Winskel, G.: Event Structures. In: Brauer, W., Reisig, W., Rozenberg, G. (eds.) APN 1986. LNCS, vol. 255, pp. 325–392. Springer, Heidelberg (1987)