ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240641

A Logic for Policy Based Resource Exchanges in Multiagent Systems

Lorenzo Ceragioli [a], Pierpaolo Degano [a,b], Letterio Galletta [a] and Luca Viganò [c]

[a]IMT School for Advanced Studies Lucca, Italy
[b]Dipartimento di Informatica, University of Pisa, Italy
[c]Department of Informatics, King's College London, UK

Abstract. In multiagent systems autonomous agents interact with each other to achieve individual and collective goals. Typical interactions concern negotiation and agreement on resource exchanges. Modeling and formalizing these agreements pose significant challenges, particularly in capturing the dynamic behaviour of agents, while ensuring that resources are correctly handled. Here, we propose exchange environments as a formal setting where agents specify and obey exchange policies, which are declarative statements about what resources they offer and what they require in return. Furthermore, we introduce a decidable extension of the computational fragment of linear logic as a fundamental tool for representing exchange environments and studying their dynamics in terms of provability.

1 Introduction

Multiagent systems represent complex environments where autonomous agents interact to achieve individual and collective goals. Central to these interactions is the concept of resource exchange, which requires agents to negotiate and reach an agreement about the allocation of resources. Modeling and formalizing these agreements pose significant challenges to capture the behaviour of agents, in particular for ensuring that resources are allocated profitably for the agents involved in exchanges. Besides AI, the study of such models has an effect on a broad spectrum of research domains, ranging from cooperative problem-solving to sharing economy scenarios. In cooperative problem-solving, the allocation of resources to agents is essential for enabling each agent to fulfill its assigned tasks effectively. In a typical sharing economy scenario, a community of users rely on a digital platform to foster collaboration and to share and transfer to each other resources and assets via peer-to-peer transactions. Dynamic resource management has been addressed from diverse perspectives, including the design of negotiation and optimization strategies, game-theoretical analysis and logics. In particular, the last approach emphasizes the development of frameworks to represent and study resource allocation and negotiation in multiagent systems, leveraging logic as a fundamental tool for this purpose. Here, we follow the logical approach, and provide a twofold contribution. First, we introduce the notion *exchange environment* to formalize a multiagent system that aims to model both cooperative and competitive behavior. An exchange environment is a transition system, where states record the ownership of the resources, and transitions represent resource exchanges between agents, who can form coalitions. Furthermore, exchanges are constrained by declarative statements, called *exchange policies*, which agents specify in isolation to prescribe what resources they offer and what they require in return (examples are in Section 2). Using such policies, agents regulate competitiveness and foster cooperation. The exchanges in a transition must guarantee that each participant gives the promised resources and gets the required ones, namely it is an *agreement*.

In addition, we consider agent's *evaluation functions*, and characterize when each permitted exchange is beneficial to all the agents of a coalition, namely it is a *deal*. Our second contribution is checking that a resource exchange is a deal. To achieve that, it suffices to inspect all the rules of a policy and to check that the utility value of the obtained resources is greater than that of those given away.

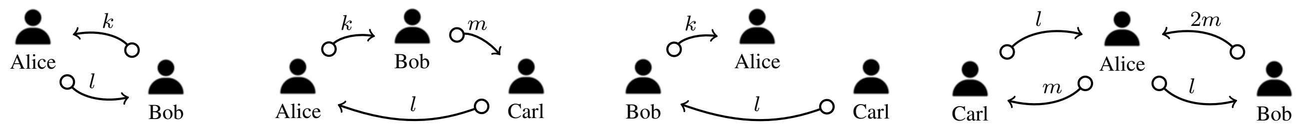
Moreover, a crucial issue arises when verifying that the policies of all the participants are met, i.e. the exchange is an agreement. Agreements may be circular, as it is typical of human and of virtual contracts. To address this issue, we extend the computational fragment of linear logic to obtain *Computational Exchange Logic*, CEL for short. This logic handles circularity through a specific operator, called *linear contractual implication*, inspired by PCL [2]. To the best our knowledge, CEL is the first logic that combines linear and contractual aspects. Every exchange *exc* is then encoded as a CEL formula, and verifying that *exc* is indeed an agreement is reduced to proving the corresponding formula. This procedure is effective because the validity of CEL formulas is decidable, which is another main technical result of ours.

Another advantage of CEL is that it provides us with the means to consider as legal sequences of exchanges where an agents can contract temporary debts by offering resources that they do not currently have, but will acquire in a subsequent exchange. Remarkably, the cut rule of the logic suffices to handle debts.

We proceed as follows. Section 2 presents a running example. We formalize exchange environments, policies and their relation with valuation functions in Section 3. Our logic CEL is in Section 4, and its extension to deal with debts in Section 5. Finally, Section 6 presents related works and Section 7 draws conclusions. Please, refer to the extended version for the full proofs [3].

2 Working examples and exchange policies

Consider three agents Alice, Bob and Carl (abbreviated `A`, `B` and `C`). Let their set of resources consists of kiwis, lemons and mandarins (written `k`, `l` and `m`). Assume agents freely form a coalition and interact by exchanging resources among them to obtain others more



(a) Direct exchange. (b) Circular exchange. (c) Coalition payment. (d) Debit.

Figure 1: Examples of agreements among players.

valuable for them (some examples of exchanges are in Figure 1.). Instead of directly bargaining one with the others, agents define their policies in advance whose application is automatic: an agent accepts to perform an exchange if and only if it obeys her policy. Our goal is twofold: (1) to propose a model rich enough to handle the circular reasoning needed for resolving the conditions expressed by the policies; and (2) to guarantee that the policy of a coalition accepts an exchange if and only if it advantages its members.

Example 1. *Assume A(lice) prefers k(iwis) over l(emons), while B(ob) preferences are opposite. Assume also that A accepts to perform exchanges with B. Then, A's policy includes a rule stating that she wills to exchange kiwis for lemons with B. Say that B accepts exchanges with A as far as he gets l in return for k from someone. Their policies can be roughly described as the following statements, which permit the direct exchange depicted in Figure 1a:*

A1 *I give B a l if I get a k in return from him;*
B1 *I give A a k if I get a l in return.*

Direct exchanges can be composed, e.g. A and B can exchange two fruits at a time. However, not every agreement involves two parts only: circular agreements allow agents to perform exchanges that cannot be expressed as combination of direct exchanges.

Example 2. *Assume A accepts to exchange a k for a l, B a m(andarin) for a k, and C a l for a m. Their policies include:*

A2 *I give any agent a k if I get a l in return;*
B2 *I give any agent a m if I get a k in return;*
C1 *I give any agent a l if I get a m in return.*

Suppose that A, B, C have each a single k, m, and l, and want a l, a k, and a m, respectively. No exchange between two agents satisfies their policies at the same time. But a circular exchange as in Figure 1b can take place, where each agent gives something to another and is payed by a third one so that all of them are satisfied at the end.

It is not always the case that an agent wants something in return for herself: resources can be given for free (e.g. if their value is less than zero for the owner) or for helping someone else, e.g. a member of the same coalition.

Example 3. *Assume A and C are in a coalition: A accepts to pay a m to everyone who gives a k to C, and C accepts to give a l in return for any k given to A. The policy of their coalition includes*

AC1 *A gives you a m if you give a k to C in return;*
AC2 *C gives you a l if you give a k to A in return.*

*Since B accepts to exchange k for l (rule **B1**), the exchange in Figure 1c can occur: B is happy to receive a l for a k, A to receive k, and C to pay for her (in the rules, "you" stands for* any agent*).*

So far we have only evaluated exchanges in terms of satisfaction of agent's policies. However, another constraint is put on exchanges: an agent a offering a resource r must possess it before getting the wanted resource. In Section 5, we show that a can still obtain what she wants, incurring a temporary debt that is paid back by acquiring r by some other agent.

Example 4. *Assume that B and A want to exchange a l for two m of B. Their policies then contain the rules:*

A3 *I give B a l if he gives me two m in return;*
B3 *I give A two m if she gives me a l in return.*

*A direct exchange can take place, if both have the needed resources, but assume that A has no l and m. A can however perform an exchange with C (allowed by rule **C1**) incurring a temporary debt of l, that is given to B to obtain two ms, and pay the debt (see Figure 1d).*

3 Policy Based Exchange Environments

In this section, we first introduce the notion of *exchange environment* as a formal model of scenarios where agents join coalitions and exchange resources according to their preferences and goals. This model is a transition system where states are resources allocations and transitions are resource exchanges between agents of a coalition. Then, we introduce the notion of *exchange policies* that are statements defined by coalitions in isolation to express what they offer to others and what they want in return. The transitions must obey the policies of the involved agents and we call them *agreements*. Moreover, we introduce the notion of *valuation function* to capture the utility that a given resource allocation has for each agent. We characterize then a *deal* as an exchange that increases the utility for the agents of a coalition, and a policy as rational when it leads to deals.

3.1 Exchange Environments

Below, we assume the following finite sets: a set $\mathcal{R}$ of *resources*, ranged over by r, r', r'' each associated with a (fixed) quantity $q(r) \in \mathbb{N}$; a set $\mathcal{A}$ of *agents*, ranged over by a, a', a''.

We start by defining resource allocations. Intuitively, they specify the resources each agent owns under the condition that we cannot assign more resources than the available ones. Formally,

Definition 1 (Resource Allocation)**.** *A resource allocation st is a function associating each agent with a multiset of $\mathcal{R}$ such that $\sum_{a \in \mathcal{A}} st(a)(r) = q(r)$.*

Next, we introduce the notions of transfer and exchange. A *transfer* occurs when a agent a sends her resource r to another agent a'. An *exchange* is a finite multiset of transfers.[1] Then, we define an *exchange environment* as a transition system with allocations as states and exchanges as transitions.

[1] Hereafter we use multisets, i.e. sets with many occurrences of the same object. As usual, we represent multisets by functions $f, g, \ldots$ from each element to the natural number of its occurrences in the multiset. Also, the disjoint union for multisets $(f \uplus g)(x)$ is defined as $f(x) + g(x)$ for every x in the domain. For simplicity, we carry the set notation over multisets.

Definition 2 (Transfer, Exchange and Exchange Environment)**.** *An* exchange *is a multiset* $exc \in Exc$ *of* transfers $tr \in Tr$*, where* tr *is a triple* $a \overset{r}{\mapsto} a'$*, with* $a' \neq a$*.*

An exchange environment *is a pair* $(St, \rightarrow)$*, where* $\rightarrow \subseteq St \times Exc \times St$ *is the* transition relation *that contains the triple* $st \xrightarrow{exc} st'$ *if and only if for all* $a \in \mathcal{A}$ *and* $r \in \mathcal{R}$ *the following conditions hold*

(1) $\sum_{a'} exc(a \overset{r}{\mapsto} a') \leq st(a)(r)$ *and*

(2) $st'(a)(r) = st(a)(r) + \sum_{a''} exc(a'' \overset{r}{\mapsto} a) - \sum_{a'} exc(a \overset{r}{\mapsto} a')$

Condition (1) ensures that an exchange exc is possible only when an agent a owns enough resources. Condition (2) ensures that the allocation is correctly updated and that no resource is created or destroyed. We say that an agent a is *involved* in a transition if it appears in the exchange labeling it.

3.2 Exchange Policies

So far, agent intents play no role, and thus there is no guarantee that a transition of the exchange environment complies with their preferences. We propose an operational characterization of these preferences via *exchange policies* where each policy specifies under which conditions an agent accepts an exchange. The next subsection also introduces *valuation functions* to provide a quantitative measure of exchanges, associating each allocation with a utility value for each agent. As common in real world, agents can join *coalitions*, i.e. sets of agents that define shared policies to obtain mutual benefits. Coalitions define their policies in isolation, and rely on them to perform decisions about exchanges.

Roughly, an exchange policy is a set of *exchange rules*, written $exc \lhd exc'$ to be read as follows: the agents in the coalition are willing to perform the exchanges exc in return for the exchange exc'. Of course, it is not possible to promise transfers on behalf of agents not in the coalition. The exchange policy determines whether an exchange requires the pay-off to be given directly to the agent who gives away some resources or to another agent in the coalition, therefore allowing some agent to pay for others (and to accept that others are paying for them). Formally:

Definition 3 (Exchange Rules and Policies)**.** *Given a coalition* $C \subseteq \mathcal{A}$*, an* exchange rule *is a pair* $exc \lhd exc' \in Exc \times Exc$ *such that for all* $a \overset{r}{\mapsto} a' \in exc$ *and for all* $a' \overset{r}{\mapsto} a \in exc'$*,* $a \in C$*.*

The exchange policy pol_C *of* C *is a set of exchange rules.*

The simplest policies are for single agents (i.e. singleton coalitions) giving resources for something in return.

Example 5. *We restate the rule* ***A1*** *of the example above, i.e.* $\mathtt{A}$ *wants to exchange* $\mathtt{l}$ *resources with* $\mathtt{B}$ *for* $\mathtt{k}$*.*

$$pol_{\{\mathtt{A}\}} \supseteq \{\{\mathtt{A} \overset{\mathtt{l}}{\mapsto} \mathtt{B}\} \lhd \{\mathtt{B} \overset{\mathtt{k}}{\mapsto} \mathtt{A}\}\}$$

The rule ***B1*** *is more general, as* $\mathtt{B}$ *does not care who is paying him.*

$$pol_{\{\mathtt{B}\}} \supseteq \bigcup_{a \in \mathcal{A}} \{\{\mathtt{B} \overset{\mathtt{k}}{\mapsto} \mathtt{A}\} \lhd \{a \overset{\mathtt{l}}{\mapsto} \mathtt{B}\}\}$$

Finally, in ***C1****,* $\mathtt{C}$ *accepts to perform exchanges with everybody.*

$$pol_{\{\mathtt{C}\}} \supseteq \bigcup_{a, a' \in \mathcal{A}} \{\{\mathtt{C} \overset{\mathtt{l}}{\mapsto} a\} \lhd \{a' \overset{\mathtt{m}}{\mapsto} \mathtt{C}\}\}$$

The same is also true for the rules ***A2*** *and* ***B2****.*

$$pol_{\{\mathtt{A}\}} \supseteq \bigcup_{a, a' \in \mathcal{A}} \{\{\mathtt{A} \overset{\mathtt{k}}{\mapsto} a\} \lhd \{a' \overset{\mathtt{l}}{\mapsto} \mathtt{A}\}\}$$

$$pol_{\{\mathtt{B}\}} \supseteq \bigcup_{a, a' \in \mathcal{A}} \{\{\mathtt{B} \overset{\mathtt{m}}{\mapsto} a\} \lhd \{a' \overset{\mathtt{k}}{\mapsto} \mathtt{B}\}\}$$

Note that we can write the rules $\emptyset \lhd exc$ and $exc' \lhd \emptyset$: the first means that the coalition accepts to receive the resources in exc for free, the second that they are happy to perform the exchange exc' without receiving anything in return.

Example 6. *In the policy below,* $\{\mathtt{A}, \mathtt{C}\}$ *is a coalition, and* $\mathtt{C}$ *will pay with an* $\mathtt{l}$ *every agent that gives a* $\mathtt{k}$ *to* $\mathtt{A}$ *(rule* ***AC2*** *of Example 2).*

$$pol_{\{\mathtt{A},\mathtt{C}\}} \supseteq \bigcup_{a \in A} \{\{\mathtt{C} \overset{\mathtt{l}}{\mapsto} a\} \lhd \{a \overset{\mathtt{k}}{\mapsto} \mathtt{A}\}\}$$

We now move towards the definition of agreements as exchanges that satisfy the policies of all agents involved. First, we say that an exchange is a accepted by a coalition when it respects its policy. Intuitively, this happens if all the agents of the coalition receive in return (as a payoff) what they are asking for each resource that they are giving away (as a contribution). Note that this check can be done by the agents of the coalition in isolation. Formally:

Definition 4 (Accepted Exchanges)**.** *Let* $pol_C \vDash_{exc'} exc$ *be the smallest relation over* $Pol \times Exc \times Exc$ *such that*

1. $pol_C \vDash \emptyset \lhd \emptyset$*;*
2. $pol_C \vDash exc \lhd exc'$*, if* $exc \lhd exc' \in pol_C$*;*
3. $pol_C \vDash (exc_1 \uplus exc_2) \lhd (exc'_1 \uplus exc'_2)$*, if* $pol_C \vDash exc_i \lhd exc'_i$*,* $i = 1, 2$*.*

The coalition C *accepts the exchange* $exc \uplus exc'$ *if* $pol_C \vDash exc' \lhd exc$*, and we call* exc *its contribution and* exc' *its payoff.*

Intuitively, an exchange is an *agreement* if it satisfies the policies of all the involved coalitions. Consider Example 1: the exchange $exc = \{\mathtt{A} \overset{\mathtt{l}}{\mapsto} \mathtt{B}, \mathtt{B} \overset{\mathtt{k}}{\mapsto} \mathtt{A}\}$ is an agreement because the left part of the rule of each agent matches the right part of the other, and their union is exc. This condition is lifted up to sets of rules: the union of the left parts of some rules of the agents must equate the union of all their right parts. An example is the circular agreement $\{\mathtt{A} \overset{\mathtt{k}}{\mapsto} \mathtt{B}, \mathtt{B} \overset{\mathtt{m}}{\mapsto} \mathtt{C}, \mathtt{C} \overset{\mathtt{l}}{\mapsto} \mathtt{A}\}$ in Figure 1c, which is obtained by using the rules **A2**, **B2** and **C1** of Example 5.

Definition 5 (Agreement)**.** *An exchange* exc *is an* agreement *if and only if for all coalition* C *such that* $C \subseteq \mathcal{A}$ *there exists a pair of exchanges* exc_C *and* exc'_C *such that* $pol_C \vDash exc_C{}' \lhd exc_C$ *and* $\biguplus_{C \subseteq \mathcal{A}} exc'_C = \biguplus_{C \subseteq \mathcal{A}} exc_C = exc$*.*

Note that the disjoint union of agreements is still an agreement. Actually, disjoint union is critical for defining agreements. Verifying disjointness requires a sort of global check on the partitioning of the exchanged resources. Otherwise, the same resource can be *offered more than once* to different agents (a sort of double spending), and this may go unnoticed since each coalition only knows its policy and its portion of the overall exchange.

Example 7. *Consider Example 5 and 6, and the following exchanges where* $\mathtt{B}$ *gives an* $\mathtt{k}$ *to* $\mathtt{A}$ *and both* $\mathtt{A}$ *and* $\mathtt{C}$ *pay for it with an* $\mathtt{l}$ *(i.e. a double spending occurs):*

$$exc = \{\mathtt{B} \overset{\mathtt{k}}{\mapsto} \mathtt{A}, \mathtt{A} \overset{\mathtt{l}}{\mapsto} \mathtt{B}, \mathtt{C} \overset{\mathtt{l}}{\mapsto} \mathtt{B}\}.$$

The exchange may seem fair to both $\mathtt{A}$ *(for* ***A1****) and* $\mathtt{C}$ *(for* ***AC2****), if* exc *is decomposed as the (non-disjoint) union of*

$$exc_{\{\mathtt{A},\mathtt{C}\}} = \{\mathtt{C} \overset{\mathtt{l}}{\mapsto} \mathtt{B}\} \qquad exc'_{\{\mathtt{A},\mathtt{C}\}} = \{\mathtt{B} \overset{\mathtt{k}}{\mapsto} \mathtt{A}\}$$

$$exc_{\{\mathtt{A}\}} = \{\mathtt{A} \overset{\mathtt{l}}{\mapsto} \mathtt{B}\} \qquad exc'_{\{\mathtt{A}\}} = \{\mathtt{B} \overset{\mathtt{k}}{\mapsto} \mathtt{A}\}$$

$$exc_{\{\mathtt{B}\}} = \{\mathtt{B} \overset{\mathtt{k}}{\mapsto} \mathtt{A}, \mathtt{B} \overset{\mathtt{k}}{\mapsto} \mathtt{A}\} \qquad exc'_{\{\mathtt{B}\}} = \{\mathtt{A} \overset{\mathtt{l}}{\mapsto} \mathtt{B}, \mathtt{C} \overset{\mathtt{l}}{\mapsto} \mathtt{B}\}$$

Of course, $\mathtt{B}$ is scamming $\mathtt{A}$ and $\mathtt{C}$: they are both paying for the same resource. Indeed exc is not an agreement, but it is $exc' = \{\mathtt{B} \overset{k}{\mapsto} \mathtt{A}, \mathtt{B} \overset{k}{\mapsto} \mathtt{A}, \mathtt{A} \overset{1}{\mapsto} \mathtt{B}, \mathtt{C} \overset{1}{\mapsto} \mathtt{B}\} = exc_{\{\mathtt{A},\mathtt{C}\}} \uplus exc_{\{\mathtt{A}\}} \uplus exc_{\{\mathtt{B}\}} = exc'_{\{\mathtt{A},\mathtt{C}\}} \uplus exc'_{\{\mathtt{A}\}} \uplus exc'_{\{\mathtt{B}\}}$ (note that exc' is the disjoint union of the exchanges in Figure 1a and 1c).

3.3 Policies and Valuation Functions

We now characterize when a coalition increases its evaluation of an assignment through an agreement. First, each resource is assigned a value by each agent a, using which a gets the overall value of an assignment. Note that the following definition introduces special valuation functions that are called additive in [16].

Definition 6. *Let $av_a : \mathcal{A} \times \mathcal{R} \to \mathbb{Z}$ represent the (positive or negative) value that the agent a associates with the case that a' holds a resource r.*
The valuation function *$v_a\colon St \to \mathbb{Z}$ of the agent a is the function $v_a(st) = \sum_{a' \in \mathcal{A}} \sum_{r \in \mathcal{R}} st(a')(r) \cdot av_a(a', r)$.*

Definition 7 (Deal)**.** *Given a coalition C and an exchange exc, let $exc{\downarrow_C} = \{a_1 \overset{r}{\mapsto} a_2 \in exc \mid a_1 \in C \text{ or } a_2 \in C\}$.*
Then $exc{\downarrow_C}$ is a deal *for C if and only if $\forall a \in C$ all the transitions $st \xrightarrow{exc\downarrow_C} st'$ are such that $v_a(st') \geq v_a(st)$.*

Theorem 1 (Policies and deals)**.** *Let C be a coalition and let exc be an exchange accepted by pol_C. If every $exc'' \lhd exc' \in pol_C$ is such that $\forall a \in C. W(a, exc'' \uplus exc') \geq 0$, then exc is a deal for C, where $W(a, \{a_i \overset{r_i}{\mapsto} a'_i\}_{i\in I}) = \sum_{i \in I}(av_a(a'_i, r_i) - av_a(a_i, r_i))$.*

An immediate consequence of Theorem 1 is that a policy is *sound*, in that it accepts only deals for C if its rules $exc' \lhd exc$ are such that the increment of value granted by its payoff exc' is greater than the loss of its contribution exc. One would like to have in addition that the policy pol_C accepts all the deals for C, a sort of *completeness*. If both cases hold, one has the so-called *rational* policies.

While Theorem 1 suggests an easy way of checking a policy sound, verifting it is complete may require a brute force analysis in the case of finite resources. Also, one can build one (out of the equivalent) less restrictive correct policy, namely a rational one, starting from the valuation function in hand, because the proof of the following theorem is constructive. The idea is to split an $exc{\downarrow_C}$ in all possible pairs exc'', exc' such that their weight W is non negative and insert the rule $exc'' \lhd exc'$ in the policy pol_C.

Theorem 2. *Given the valuation functions of all the agents of a coalition C, there exists a rational policy for C.*

Even when the policies of all the coalitions are rational, it is not always the case that a transition is a deal for *every* coalition. E.g. let $C = \{a_1, a_2\}$ and $C' = \{a'_1, a'_2\}$ be two disjoint coalitions and let a_1 and a'_1 exchange some resources with a_2 and a'_2, respectively. Even if the gain of a_1 performing the exchange with a_2 is positive for C, it may be less than the loss of value for a_1 caused by a'_1 and a'_2 performing their exchange. This is never the case with special valuation functions, through which an agent assigns the same value to all the resources that does not belong to her. In this case an agreement, which is also a deal, causes a quasi-Pareto improvement, because the value of the allocation does not decrease for all agents.

Theorem 3. *If every policy is rational and $av_a(a', r) = k_a$ for some fixed $k_a \in \mathbb{Z}$ when $a \neq a'$, then each transition $st \xrightarrow{exc} st'$ with exc an agreement is such that $v_a(st') \geq v_a(st)$ for all a.*

4 A Logic for Characterizing Agreements

So far, we have characterised agreements at the basic level of exchange environments and coalition policies. We develop now a linear logic for modeling exchange environments, policies and agreements, and reduce the check of an exchange being an agreement to the validity of the sequent modelling it.

We choose linear logic for its unique features to declaratively represent resources and their usage. In this view, resources are represented as logical assumptions in a proof, and each of them can only be used once and only once during the proof: resources can neither be duplicated nor thrown away at will. A deduction in the logic models the way resources are manipulated, and this is convenient in our approach to formalize exchange environments and their behaviour.

4.1 Contractual Exchange Logic

To define Contractual Exchange Logic (CEL) we start from a computational fragment of linear logic, following [9], and we then extend it with a new operator, inspired by PCL [2], to express the typical offer/return actions of contracts.

Definition 8 (CEL Propositions)**.** *A CEL proposition ϕ is defined as*

$$\phi ::= \sigma \mid \delta \mid \theta \mid \omega \mid \xi \qquad \sigma ::= I \mid r@a \mid \sigma \otimes \sigma$$
$$\delta ::= I \mid r@a \multimap r@a \mid \delta \otimes \delta \qquad \theta ::= I \mid \delta \multimap\!\!\!\circ\, \delta \mid \theta \otimes \theta$$
$$\xi ::= !\delta \mid \xi \otimes \xi \qquad \omega ::= !\theta \mid \omega \otimes \omega$$

We denote multisets of propositions using (the corresponding) Greek capital letters: $\Phi, \Sigma, \Delta, \Theta, \Xi, \Omega$.

We refer to the common resource-based interpretation of linear logic for describing the intuitive meaning of the propositions above. In this interpretation $r@a$ stands for a resource association, meaning that r currently belongs to the agent a. A predicate $r@a \multimap r'@a'$ is a consumable processes (they can run only once) transforming the atomic $r@a$ into $r'@a'$. Predicates of the form $\delta \multimap\!\!\!\circ\, \delta'$, composed with our new operator called *linear contractual implication*, are promises stating that δ' will be performed if also δ is. Finally, ω and ξ represents (non-linear) information about promises and processes that can be used ad libitum, and tensor product allows composing multisets of previous entities, where I (representing *true* in linear logic) is the empty multiset. The CEL sequents are as follows.

Definition 9 (CEL Sequent)**.** *A CEL* sequent *is of form*

$$\Omega, \Xi, \Theta, \Delta, \Sigma \vdash \phi.$$

A sequent is initial *if $\Theta, \Delta = \emptyset$ and $\phi = \sigma$ for some σ, i.e. if it has the form $\Omega, \Xi, \Sigma \vdash \sigma$ (we omit hereafter the empty components).*

The CEL sequent $\Xi, \Omega, \Theta, \Delta, \Sigma \vdash \sigma$ intuitively means that the state σ is a possible transformation of Σ using the processes and promises in the assumptions $\Xi, \Omega, \Theta, \Delta$.

The deduction system for $\vdash$ are in Figure 2. They mostly result from instantiating the standard ones of the multiplicative fragment of linear logic on the CEL sequents. Note that we omit the cut rule in this fragment. In addition there are two rules for the linear contractual implication: the ($\multimap\!\!\!\circ$-left) rule introduces the operator on the left if what is required by the contract is satisfied by the consequences; the ($\multimap\!\!\!\circ$-split) rule deals with composition of contracts.

$$\frac{!\phi, !\phi, \Phi \vdash \phi'}{!\phi, \Phi \vdash \phi'}\text{(Cont)} \qquad \frac{\Phi \vdash \phi'}{!\phi, \Phi \vdash \phi'}\text{(Weak)} \qquad \frac{\phi, \Phi \vdash \phi'}{!\phi, \Phi \vdash \phi'}\text{(!-left)} \qquad \frac{\Phi \vdash \phi}{I, \Phi \vdash \phi}(I\text{-left}) \qquad \frac{}{\vdash I}(I\text{-right})$$

$$\frac{}{\phi \vdash \phi}\text{(Ax)} \qquad \frac{\phi, \phi', \Phi \vdash \phi''}{\phi \otimes \phi', \Phi \vdash \phi''}(\otimes\text{-left}) \qquad \frac{\Phi \vdash \phi \qquad \Phi' \vdash \phi'}{\Phi, \Phi' \vdash \phi \otimes \phi'}(\otimes\text{-right})$$

$$\frac{\Sigma \vdash \sigma}{\Sigma, \sigma \multimap \sigma' \vdash \sigma'}(\multimap\text{-left}) \qquad \frac{\delta \vdash \delta' \qquad \delta' \vdash \delta \qquad \Phi, \delta' \vdash \sigma}{\Phi, \delta -\!\!\infty\, \delta' \vdash \sigma}(-\!\!\infty\text{-left}) \qquad \frac{\Phi, \delta \otimes \delta'' -\!\!\infty\, \delta' \otimes \delta''' \vdash \sigma}{\Phi, \delta -\!\!\infty\, \delta', \delta'' -\!\!\infty\, \delta''' \vdash \sigma}(-\!\!\infty\text{-split})$$

Figure 2: CEL rules.

Example 8. *A linear implication $r@a \multimap r@a'$ represents a transfer where a predicate $r@a$ is consumed and a new $r@a'$ is created. Note that $r@a \multimap r@a', r@a \vdash r@a'$ is indeed valid.*

A linear contractual implication $\delta -\!\!\infty\, \delta'$ encodes a promise of δ' in return of δ. Direct exchanges, that will be used to encode agreements like the one of Figure 1a, are expressed by a sequent of the form

$$\delta -\!\!\infty\, \delta', \delta' -\!\!\infty\, \delta, \Sigma \vdash \sigma$$

where the exchange δ' is promised in return for δ and vice versa. The following derivation proves that the exchange δ, δ' that transforms the state Σ in σ can be performed (we omit the proofs Π and Π', using (Ax), ($\otimes$-left) and ($\otimes$-right) rules, as they are straightforward):

$$\dfrac{\dfrac{\dfrac{\Pi}{\delta \otimes \delta' \vdash \delta' \otimes \delta} \qquad \dfrac{\Pi'}{\delta' \otimes \delta \vdash \delta \otimes \delta'} \qquad \dfrac{\delta, \delta', \Sigma \vdash \sigma}{\delta' \otimes \delta, \Sigma \vdash \sigma}(\otimes\text{-left})}{\delta \otimes \delta' -\!\!\infty\, \delta' \otimes \delta, \Sigma \vdash \sigma}(-\!\!\infty\text{-left})}{\delta -\!\!\infty\, \delta', \delta' -\!\!\infty\, \delta, \Sigma \vdash \sigma}(-\!\!\infty\text{-split})$$

For the circular exchange of Figure 1b, we use $\delta -\!\!\infty\, \delta', \delta' -\!\!\infty\, \delta'', \delta'' -\!\!\infty\, \delta, \Sigma \vdash \sigma$, and apply ($-\!\!\infty$-split) twice.

Noticeably, CEL proofs can be normalized[2].

Definition 10 (Normal Proofs)**.** *A CEL proof for an initial sequent is* normal *if it can be decomposed in either form, where Π_1 only uses (Weak), (Cont), ($\otimes$-left) and (!-left) rules; Π_2 only ($-\!\!\infty$-split); Π_3 and Π_3' only ($\otimes$-right), ($\otimes$-left), (Ax), (I-right) and (I-left); Π_4 only ($\multimap$-left), ($\otimes$-right), ($\otimes$-left), (Ax), (I-right) and (I-left).*

$$\begin{array}{c} \dfrac{\Pi_4}{\Delta, \Sigma \vdash \sigma} \\ \vdots\ \Pi_1 \\ \Omega, \Xi, \Sigma \vdash \sigma \\ \textit{normal form 1} \end{array} \qquad \begin{array}{c} \dfrac{\Pi_3 \quad \Pi_3' \quad \dfrac{\Pi_4}{\Delta, \delta, \Sigma \vdash \sigma}}{\theta, \Delta, \Sigma \vdash \sigma}(-\!\!\infty\text{-left}) \\ \vdots\ \Pi_2 \\ \Theta, \Delta, \Sigma \vdash \sigma \\ \vdots\ \Pi_1 \\ \Omega, \Xi, \Sigma \vdash \sigma \\ \textit{normal form 2} \end{array}$$

These two normal forms are general: a proof exists for an initial sequent only if a proof in normal form exists.

Theorem 4 (CEL Normal Proofs). *For any Ω, Σ, σ, the initial sequent $\Omega; \Sigma \vdash \sigma$ is valid in CEL if and only if a normal proof Π exists for $\Omega; \Sigma \vdash \sigma$.*

For reasoning about exchange environments a logic must be decidable. We now prove that CEL is such and for that it is enough to consider normal proofs only. Note that the normal form 1 corresponds to the case where no contractual rule is applied, hence we can assume $\Omega = \emptyset$. We are thus in the context of (a fragment of) standard linear logic, and decidability follows from a suitable application of Kanovich's technique [9] that reduces validity to reachability in Petri Nets, which can be decided using [15].

Lemma 5. *An always-terminating algorithm exists that decides the existence of a proof in the normal form 1 for a given initial sequent.*

Finally, for the normal form 2 we reduce to the previous case. We prove that a proof in the normal form 2 can be effectively rewritten in the normal form 1. The reduction is performed in an algebraic framework by considering the derivations in a bottom-up fashion, starting with the sequent we are proving and constructing the premises.

Consider a semiring module[3] M over the set of natural numbers $\mathbb{N}$ with subformulas of any CEL predicate ϕ as its basis (we can safely reduce to the finite set of the ones appearing in the considered sequent). For simplicity, we call the elements $\bar{x}$ of M vectors, and write $\bar{x}(\phi)$ for the number associated by $\bar{x}$ to the basis element ϕ.

Roughly, we show that valid premises $\Delta, \Sigma \vdash \sigma$ for Π_1 in normal form 1 correspond to the linear combinations of a finite set of vectors that depends on the δ predicates appearing in the Ξ of the sequent that we are proving. This intuitively encodes the fact that we can take δ formulas ad libitum, because of the ! operator. We then replicate a similar construction for normal form 2. In particular, the existence of Π_1 depends on the encoding of the premises being a linear combinations of the encoding of θ and δ appearing in Ω and Ξ of the conclusions; the premises of Π_2 are uniquely determined by a linear function, and Π_3 and Π_3 correspond to checking an homogeneous system of linear equations.

Then, by the Hilbert basis theorem [7], we can combine these conditions and represent the set of solutions as the linear combinations of a finite set of vectors representing δ predicates, i.e. a Π_1 derivation for a proof in normal form 1.

Lemma 6. *For any $\Omega, \Xi, \Delta, \Sigma, \sigma$, there is a computable multiset Ξ' such that there exists a derivation in the normal form 2 from the sequent $\Delta, \Sigma \vdash \sigma$ to $\Omega, \Xi, \Sigma \vdash \sigma$ if and only if there exists a derivation in the normal form 1 from $\Delta, \Sigma \vdash \sigma$ to $\Xi', \Sigma \vdash \sigma$.*

The following theorem proves that CEL is decidable. Its statement mentions initial sequents only, which are however sufficient to reason about agreement transitions, as Theorem 8 below will make clear.

Theorem 7 (CEL Decidability)**.** *An always-terminating algorithm exists that decides if an initial sequent is valid in CEL.*

[2] In the following, we will call *proof* the derivation of a theorem from the axioms, and only use the term *derivation* for a derivation with open assumptions, i.e. a proof tree where the leaves are not only axioms. We also say that two proofs are *equivalent* if they prove the same sequent.

[3] A semiring module is a generalization of the notion of vector space in which the field of scalars is replaced by a semiring.

4.2 Deriving Transitions of Exchange Environments

In the following we show how to encode exchange environments and policies as CEL propositions.

Definition 11. *We write ϕ^n with $n \in \mathbb{N}$ for the tensor product of $n > 0$ instances of the proposition ϕ, meaning I if $n = 0$.*

$$(\!|st|\!) ::= \bigotimes_{a\in\mathcal{A},r\in\mathcal{R}} (r@a)^{st(a)(r)}$$

$$(\!|exc|\!) ::= \bigotimes_{a,a'\in\mathcal{A},r\in\mathcal{R}} (r@a \multimap r@a')^{exc(a\overset{r}{\mapsto}a')}$$

$$(\!|pol_C|\!) ::= \bigotimes_{exc\lhd exc'\in pol_C} !((\!|exc'|\!) \multimap\!\!\circ (\!|exc|\!))$$

Example 9. *Consider a state st with three agents* $\mathtt{A}$, $\mathtt{B}$, *and* $\mathtt{C}$ *owning a* $\mathtt{k}$, *a* $\mathtt{m}$ *and an* $\mathtt{1}$, *respectively. Then its encoding is as follows:*

$$(\!|st|\!) = \mathtt{k}@\mathtt{A} \otimes \mathtt{m}@\mathtt{B} \otimes \mathtt{1}@\mathtt{C}$$

The encoding of $exc = \{\mathtt{A} \overset{\mathtt{k}}{\mapsto} \mathtt{B}, \mathtt{B} \overset{\mathtt{m}}{\mapsto} \mathtt{C}, \mathtt{C} \overset{\mathtt{1}}{\mapsto} \mathtt{A}\}$ *is*

$$(\!|exc|\!) = (\mathtt{k}@\mathtt{A} \multimap \mathtt{k}@\mathtt{B}) \otimes (\mathtt{m}@\mathtt{B} \multimap \mathtt{m}@\mathtt{C}) \otimes (\mathtt{1}@\mathtt{C} \multimap \mathtt{1}@\mathtt{A})$$

Finally, suppose that the policies $pol_{\{\mathtt{A}\}}, pol_{\{\mathtt{B}\}}, pol_{\{\mathtt{C}\}}$ *contain only the rules* ***A2, B2*** *and* ***C1*** *in Example 5, respectively. Their encoding follows.*

$$(\!|pol_{\{\mathtt{A}\}}|\!) = \bigotimes_{a,a'\in\{\mathtt{B},\mathtt{C}\}} !((\mathtt{1}@a \multimap \mathtt{1}@\mathtt{A}) \multimap\!\!\circ (\mathtt{k}@\mathtt{A} \multimap \mathtt{k}@a'))$$

$$(\!|pol_{\{\mathtt{B}\}}|\!) = \bigotimes_{a,a'\in\{\mathtt{A},\mathtt{C}\}} !((\mathtt{k}@a \multimap \mathtt{k}@\mathtt{B}) \multimap\!\!\circ (\mathtt{m}@\mathtt{B} \multimap \mathtt{m}@a'))$$

$$(\!|pol_{\{\mathtt{C}\}}|\!) = \bigotimes_{a,a'\in\{\mathtt{A},\mathtt{B}\}} !((\mathtt{m}@a \multimap \mathtt{m}@\mathtt{C}) \multimap\!\!\circ (\mathtt{1}@\mathtt{C} \multimap \mathtt{1}@a'))$$

The following sequent is valid, as shown in the proof in Figure 3

$$(\!|pol_{\{\mathtt{A}\}}|\!), (\!|pol_{\{\mathtt{B}\}}|\!), (\!|pol_{\{\mathtt{C}\}}|\!), (\!|st|\!) \vdash \mathtt{k}@\mathtt{B} \otimes \mathtt{m}@\mathtt{C} \otimes \mathtt{1}@\mathtt{A}$$

Note that exc *is an agreement labelling the transition* $st \xrightarrow{exc} st'$ *where* $(\!|st'|\!) = \mathtt{k}@\mathtt{B} \otimes \mathtt{m}@\mathtt{C} \otimes \mathtt{1}@\mathtt{A}$.

In the theorem below the current allocation st and the policies pol_C determine the left part of an initial sequent, while the right part is for the candidate next allocation st'. Then, a transition $st \xrightarrow{exc} st'$ where exc is an agreement exists if and only if the obtained initial sequent is valid. Note that this result implies that CEL proofs are witnesses for fairness of exchanges.

Theorem 8. *Let* $(St, \rightarrow)$ *be an exchange environment; let* $st, st' \in St$; *and let* pol_C *be the policies of the coalition* C.

Then $\biguplus_{C\in 2^{\mathcal{A}}} (\!| pol_C |\!), (\!| st |\!) \vdash (\!| st' |\!)$ *is valid if and only if there exists an agreement* exc *such that* $st \xrightarrow{exc} st'$.

The proof is carried on in three steps. First, we note that a proof for $\biguplus_{C\in 2^{\mathcal{A}}} (\!| pol_C |\!), (\!| st |\!) \vdash (\!| st' |\!)$ can always be transformed in one in the normal form 2, where $\Xi = \Delta = \emptyset$. Then we show that the derivations Π_1, Π_2 and Π_3 exist whenever $\delta = (\!| exc |\!)$ for some exc that is an agreement. Finally, a proof Π_4 exists for $(\!| exc |\!), (\!| st |\!) \vdash (\!| st' |\!)$ if and only if $st \xrightarrow{exc} st'$ is a transition of the exchange environment.

5 Extending CEL with debts

So far, we have only considered exchanges where no debts are permitted: an agent must possess a resource she promises as required by condition (1) of Definition 2. In case agents trust each other or there is a regulating trusted third party, it is possible to extend our model and logic to consider a wider class of transitions, as that in Example 4. This requires updating the exchange environments by weakening the condition (1), allowing an agent to incur a temporary debt.

Definition 12. *An* exchange environment with debts *is* $(St, \dashrightarrow)$ *with* St *defined as in Definition 2, and* $\dashrightarrow \subseteq St \times Exc \times St$ *contains the triples* $st \overset{exc}{\dashrightarrow} st'$ *if and only if for all* $a \in \mathcal{A}$ *and* $r \in \mathcal{R}$ *both (2) from Definition 2 and the following hold*

$$(1') \sum_{a'} exc(a \overset{r}{\mapsto} a') \leq st(a)(r) + \sum_{a'} exc(a' \overset{r}{\mapsto} a)$$

For brevity, we write below $\rightarrow_{ok} \subseteq St \times St$ for the transition induced by agreements only: $st \rightarrow_{ok} st$ if and only if $st \xrightarrow{exc} st$ for some agreement exc. As a matter of fact, in an exchange environment, every pair of allocations are connected by a transition, but not by one labelled by an agreement. Now, we similarly filter $\dashrightarrow$ and define $\dashrightarrow_{ok}$ to be the transition relation in an exchange environment with debits for which a fair exchange exists. Note that some allocations reachable with these transitions cannot be reached without permitting debits, i.e. $\rightarrow_{ok} \subsetneq \dashrightarrow_{ok}$.

Example 10. *Consider again Example 4, where* $\mathtt{B}$ *and* $\mathtt{A}$ *want to exchange a* $\mathtt{1}$ *for two* $\mathtt{m}$ *of* $\mathtt{B}$. *Assume that the current assignment* st *is such that* $\mathtt{A}$ *has nothing,* $\mathtt{B}$ *has two* $\mathtt{m}$ *and* $\mathtt{C}$ *has one* $\mathtt{1}$.

The relevant policy rules are ***A3, B3, C1***:

$$pol_{\{\mathtt{A}\}} \supseteq \{\{\mathtt{A} \overset{\mathtt{1}}{\mapsto} \mathtt{B}\} \lhd \{\mathtt{B} \overset{\mathtt{m}}{\mapsto} \mathtt{A}\}\}$$

$$pol_{\{\mathtt{B}\}} \supseteq \{\{\mathtt{B} \overset{\mathtt{m}}{\mapsto} \mathtt{A}, \mathtt{B} \overset{\mathtt{m}}{\mapsto} \mathtt{A}\} \lhd \{\mathtt{A} \overset{\mathtt{1}}{\mapsto} \mathtt{B}\}\}$$

$$pol_{\{\mathtt{C}\}} \supseteq \bigcup_{a,a'\in\mathcal{A}} \{\{\mathtt{C} \overset{\mathtt{1}}{\mapsto} a\} \lhd \{a' \overset{\mathtt{m}}{\mapsto} \mathtt{C}\}\}$$

The exchange $exc = \{\mathtt{C} \overset{\mathtt{1}}{\mapsto} \mathtt{A}, \mathtt{A} \overset{\mathtt{1}}{\mapsto} \mathtt{B}, \mathtt{B} \overset{\mathtt{m}}{\mapsto} \mathtt{A}, \mathtt{B} \overset{\mathtt{m}}{\mapsto} \mathtt{A}\}$ *depicted in Figure 1d is forbidden in* st *using exchange environments without debts, since* $\mathtt{A}$ *has no* $\mathtt{1}$*s. Instead, the transition* $st \overset{exc}{\dashrightarrow} st'$ *results in the state* st' *where* $\mathtt{A}$ *and* $\mathtt{C}$ *have each a* $\mathtt{m}$, *and* $\mathtt{B}$ *a* $\mathtt{1}$. *Note that the transfer* $\mathtt{A} \overset{\mathtt{1}}{\mapsto} \mathtt{B}$ *causes a temporary debt of* $\mathtt{A}$, *which is repaid with the transfer* $\mathtt{C} \overset{\mathtt{1}}{\mapsto} \mathtt{A}$.

Again logic comes to our rescue for deciding if a transition in an exchange environment with debts is an agreement. This is done by adding the rule (Cut) in Figure 4. Consequently, we extend the correspondence between exchange environments and CEL in presence of debts through the following corollary of Theorem 8.

Corollary 9. *Under the same conditions of Theorem 8, a transition* $st \dashrightarrow_{ok} st'$ *exists, if and only if* $\biguplus_{C\in 2^{\mathcal{A}}} (\!| pol_C |\!), (\!| st |\!) \vdash (\!| st' |\!)$ *is valid in CEL augmented with the (Cut) rule.*

Decidability of CEL is not affected by the (Cut) rule.

Corollary 10. *An always-terminating algorithm exists that decides if an initial sequent is valid in CEL augmented with the (Cut) rule.*

6 Related Work

The problem of fairly exchanging electronic assets among a set of agents has been addressed by different communities, e.g. artificial intelligence, fair exchange protocols in distributed systems [1, 4, 5, 6]. Below, we focus on those approaches that use linear logic to capture these issues, and we conclude with a comparison with logics having contractual aspects.

Logical Modelling of Resource Exchange Linear logic has been used to model resource-aware games and problems in the artificial intelligence community. They all describe the desire of agents in terms of their goals or valuation functions, and derive or recognise reasonable offers and strategies. A contribution of ours is instead a way of

$$
\dfrac{\dfrac{}{\llparenthesis exc \rrparenthesis \vdash \llparenthesis exc \rrparenthesis}\,(\text{Ax}) \quad \dfrac{}{\llparenthesis exc \rrparenthesis \vdash \llparenthesis exc \rrparenthesis}\,(\text{Ax}) \quad \dfrac{\Pi''}{\llparenthesis exc \rrparenthesis, \llparenthesis \Sigma \rrparenthesis \vdash \mathtt{k@B} \otimes \mathtt{m@C} \otimes \mathtt{1@A}}}{\llparenthesis exc \rrparenthesis \mathrel{-\!\!\circ\!\!\circ} \llparenthesis exc \rrparenthesis, \llparenthesis st \rrparenthesis \vdash \mathtt{k@B} \otimes \mathtt{m@C} \otimes \mathtt{1@A}}\,(\mathrel{-\!\!\circ\!\!\circ}\text{-left})
$$

$$
\vdots\ \Pi'
$$

$$
(\mathtt{1@C} \multimap \mathtt{1@A}) \mathrel{-\!\!\circ\!\!\circ} (\mathtt{k@A} \multimap \mathtt{k@B}), (\mathtt{k@A} \multimap \mathtt{k@B}) \mathrel{-\!\!\circ\!\!\circ} (\mathtt{m@B} \multimap \mathtt{m@C}), (\mathtt{m@B} \multimap \mathtt{m@C}) \mathrel{-\!\!\circ\!\!\circ} (\mathtt{1@C} \multimap \mathtt{1@A}), \llparenthesis st \rrparenthesis \vdash \mathtt{k@B} \otimes \mathtt{m@C} \otimes \mathtt{1@A}
$$

$$
\vdots\ \Pi
$$

$$
\llparenthesis pol_{\{\mathtt{A}\}} \rrparenthesis, \llparenthesis pol_{\{\mathtt{B}\}} \rrparenthesis, \llparenthesis pol_{\{\mathtt{C}\}} \rrparenthesis, \llparenthesis st \rrparenthesis \vdash \mathtt{k@B} \otimes \mathtt{m@C} \otimes \mathtt{1@A}
$$

Figure 3: CEL proof in the normal form 2, the proof Π'' is omitted for simplicity, and uses (Ax), ($\otimes$-left), ($\otimes$-right) and ($\multimap$-left).

$$
\dfrac{\Phi \vdash \sigma \quad \Phi', \sigma \vdash \phi}{\Phi, \Phi' \vdash \phi}\,(\text{Cut})
$$

Figure 4: Cut rule for CEL.

directly modelling what agents offer via exchange policies, combining a descriptive approach and a prescriptive one.

Harland et al. [8] show how linear logic enables reasoning about negotiations, encoding agents' goals and what they offer. Linear logic proofs recognise the negotiation outcomes that satisfy all parties.

Küngas et al. [12, 11] propose a model of cooperative problem solving, and use linear logic for encoding agents' resources, goals and capabilities. Then, each agent determines whether she can solve the problem in isolation. If she cannot, then she starts negotiating with other agents in order to find a cooperative solution. Partial deduction [10] is used to derive possible deals. The authors of [13, 14] extend their work by considering coalition formation.

Porello et al. [16] target distributed resource allocation. They encode resource ownership and transfers, as well as valuation functions representing user preferences in (various fragments of) linear and affine logic. They show how logic proofs discriminate mutually satisfactory exchanges that increase the value of the assignment for every user, thus recovering a notion of social welfare in terms of Pareto optimality. Since the valuation functions of users used to decide exchanges are assumed known, offers and negotiation are not modeled. They prove that any sequence of individually rational deals will always converge to an allocation with maximal social welfare, as known from [17]. In contrast, we directly encode the user exchange policies. Afterward we further constrain the exchanges compliant with policies with valuation functions obtaining deals. Moreover, we extend the computational fragment of linear logic with a contractual implication and we prove decidability results.

Troquard [18] models the interaction of resource-conscious agents who share resources to achieve their goals in cooperative games. Algorithms are proposed for deciding whether a group of agents can form a coalition and act together in a way that satisfies them all. Various problems concerning cooperative games are modelled in suitable fragments of linear and affine logic and their computational complexity is discussed. Our focus is instead on resource exchanges, and our context is a mixture of cooperative and competitive behaviour. In a subsequent work, Troquard [19] studies how a central authority can modify the set of Nash equilibria in a cooperative game by redistributing the initial assignment of resources to agents. The complexity of this optimization problem is discussed in terms of the chosen (fragment of) resource-sensitive logic.

Contractual logics We formalised the contractual aspects following the pioneering PCL proposed by Bartoletti and Zunino [2], which is a logic for modelling contractual reasoning. Our operator $\mathrel{-\!\!\circ\!\!\circ}$ is actually a linear version of their $\twoheadrightarrow$. The main difference with respect to PCL is that from the premise $p \twoheadrightarrow p', p' \twoheadrightarrow p$ one can derive p, p', and $p \wedge p'$. Instead, in CEL only the conjunction $p \otimes p'$ can be derived from the premise $p \mathrel{-\!\!\circ\!\!\circ} p', p' \mathrel{-\!\!\circ\!\!\circ} p$. The syntactic form of CEL sequents is inspired by Kanovich [9], who proposed a computational fragment of linear logic for reasoning on computations with consumable resources.

7 Conclusions and future work

We introduced exchange environments as a formal model for scenarios where agents join coalitions and exchange resources to achieve individual and collective goals. Our model is a transition system where states are resource allocations to agents and transitions are labelled by the exchanged resources. Moreover, we proposed exchange policies to regulate competitiveness and cooperation: agents prescribe in isolation what they offer and what they require in return.

We characterised the notion of agreement as a resource exchange where the policies of all the involved agents are met. Since agreements are often circular, checking an exchange to be such is crucial and hard. For that, we extended the computational fragment of linear logic with a new operator that handles both contracts and circularity. The resulting logic, called CEL, is decidable (Theorem 7), so checking that an exchange is an agreement is reduced to finding a proof for its encoding in CEL. We also modeled the case in which an agent incurs a temporary debt that is paid with resources she can obtain during the same exchange. Extending our logic with the cut rule sufficed to handle these cases, still maintaining decidability (Corollary 10).

In addition, we formalised when an agreement is beneficial to all the agents of a coalition, dubbed a deal, allowing agents to assign a utility value to resource allocations. Checking that a resource exchange is a deal consists in showing that the utility value of the offered resources is less than that of those required back (Theorem 1). We also showed that there exists a rational policy, i.e. accepting all and only the deals for a coalition (Theorem 2), and characterized those that guarantee to reach a Pareto increment (Theorem 3).

Future work includes extending CEL with universal quantifiers, disjunction and negation to express richer policies in a handy and concise manner. Another line of development concerns creating, disposing and transforming resources in exchanges, actions that are already available in linear logic. Finally, we plan to investigate to apply our model to describe real scenarios, such as exchanges of crypto-assets in blockchain systems.

Acknowledgements

Work partially supported by projects SERICS (PE00000014) and PRIN AM∀DEUS (P2022EPPHM) under the MUR-PNRR funded by the European Union — NextGenerationEU.

References

[1] F. Bao, R. Deng, K. Nguyen, and V. Varadharajan. Multi-party fair exchange with an off-line trusted neutral party. In *Proceedings of the Tenth International Workshop on Database and Expert Systems Applications. DEXA 99*, pages 858–862. IEEE Computer Society, 1999. doi: 10.1109/DEXA.1999.795294.

[2] M. Bartoletti and R. Zunino. A calculus of contracting processes. In *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010*, pages 332–341. IEEE Computer Society, 2010.

[3] L. Ceragioli, P. Degano, L. Galletta, and L. Viganò. A logic for policy based resource exchanges in multiagent systems, 2024. URL https://arxiv.org/abs/2408.09516.

[4] S. Even and Y. Yacobi. Relations among public key signature systems. Technical report, Technion, Haifa, 1980.

[5] M. Franklin and G. Tsudik. Secure group barter: Multi-party fair exchange with semi-trusted neutral parties. In *Financial Cryptography*, volume 1465 of *Lecture Notes in Computer Science*, pages 90–102. Springer, 1998.

[6] F. C. Freiling, M. Herlihy, and L. D. Penso. Optimal randomized fair exchange with secret shared coins. In *OPODIS*, volume 3974 of *Lecture Notes in Computer Science*, pages 61–72. Springer, 2005.

[7] P. Gordan. Über die Auflösung linearer Gleichungen mit reellen Coefficienten, Mar. 1873. URL https://doi.org/10.1007/bf01442864.

[8] J. Harland and M. Winikoff. Agent negotiation as proof search in linear logic. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 2*, AAMAS '02, page 938–939. ACM Press, 2002. doi: 10.1145/544862.544957.

[9] M. I. Kanovich. Linear logic as a logic of computations. *Ann. Pure Appl. Log.*, 67(1-3):183–212, 1994. doi: 10.1016/0168-0072(94)90011-6.

[10] J. Komorowski. An introduction to partial deduction. In A. Pettorossi, editor, *Meta-Programming in Logic*, volume 649 of *Lecture Notes in Computer Science*, pages 49–69. Springer, 1992.

[11] P. Küngas and M. Matskin. Symbolic Negotiation with Linear Logic. In *Computational Logic in Multi-Agent Systems (CLIMA 2004)*.

[12] P. Küngas and M. Matskin. Linear Logic, Partial Deduction and Cooperative Problem Solving. In *Proceedings of the International Workshop on Declarative Agent Languages and Technologies (DALT 2003)*, volume 2990 of *Lecture Notes in Computer Science*, 2003.

[13] P. Küngas and M. Matskin. Symbolic Negotiation in Linear Logic with Coalition Formation. *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pages 298–305, 2006.

[14] P. Küngas and M. Matskin. Symbolic negotiation: Partial deduction for linear logic with coalition formation. *Web Intell. Agent Syst.*, 6:193–215, 2008.

[15] E. W. Mayr. An Algorithm for the General Petri Net Reachability Problem. In *Proceedings of the Thirteenth Annual ACM Symposium on Theory of Computing*, page 238–246. ACM Press, 1981. doi: 10.1145/800076.802477.

[16] D. Porello and U. Endriss. Modelling Multilateral Negotiation in Linear Logic. In *Proceedings of ECAI 2010 — 19th European Conference on Artificial Intelligence*, pages 381–386. 2010.

[17] T. Sandholm. Contract types for satisficing task allocation. In *Proceedings of the 1998 AAAI Spring Symposium*, 2002.

[18] N. Troquard. Rich coalitional resource games. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), 2018. doi: 10.1609/aaai.v32i1.11437.

[19] N. Troquard. Individual resource games and resource redistributions. *Journal of Logic and Computation*, 30(5):1023–1062, 2020. doi: 10.1093/logcom/exaa031.