Individual Resource Games and Resource Redistributions

Nicolas Troquard

KRDB, Univ. Bozen-Bolzano, Italy
LACL, Univ. Paris-Est Créteil, France
Nicolas.Troquard@unibz.it

December, 2017

Abstract

Research in multiagent systems is advancing and one can predict its future widespread implementation in real-world systems. One needs however to acknowledge that the agents evolving in the real world have limited access to resources. They have to seek after resource objectives and compete for those resources. We introduce a class of resource games where resources and preferences are described with the language of a resource-sensitive logic. We study three decision problems, the first of which is deciding whether an action profile is a Nash equilibrium. When dealing with resources, interesting questions arise as to whether some equilibria can be eliminated or constructed by a central authority by redistributing the available resources among the agents. In our economies, division of property in divorce law exemplifies how a central authority can redistribute the resources of individuals, and why they would desire to do so. We thus study two related decision problems: rational elimination and rational construction. We consider them in the contexts of dichotomous or parsimonious preferences, and of logics that admit or not the weakening rule. This permits us to offer a variety of algorithms and complexity results that are applicable to a large number of settings.

1 Introduction

Research in multiagent systems is advancing and one can predict its future widespread implementation in real-world systems. One needs however to acknowledge that the agents evolving in the real world have limited access to resources. They have to seek after resource objectives and compete for those resources. Accordingly, the research in the formal and computational aspects of resource-conscious agents is active (e.g., [31, 20, 5, 9, 28, 1, 27]).

In this paper, we study games of resources that are aimed at representing the strategic interactions between rational agents [17] where some combinations of resources replace the abstract notions of action and preferences. In these games, players are endowed with some resources and have preferences upon some resources to be available after the game is played. Players’ actions also consist in making available some of the resources they are endowed with.

We propose a class of games of resources that exploits the formalisms and reasoning methods coming from the literature in knowledge representation and computational logics, namely resource-sensitive logics: e.g., Linear Logic, Separation Logic, BI Logic [8, 23, 16]. The languages of these logics allow a fine-grained description of resources, processes, and their harmonious combinations. In computer science, they have been quite successful at modeling systems for multi-party access and modification of shared structures, by allocation and deallocation of resources. Not based on a trivial naïve set theory and
a trivial truth, the resources used in this paper will thus be supported with a rich logical language, and elaborate reasoning features.

A resource is represented by one formula of a resource-sensitive logic $\text{LOG.}$ More specifically, we assume here that $\text{LOG}$ is some propositional variant of Linear Logic. In fact, the technical aspects of the paper can be grasped without a great understanding of Linear Logic. Nonetheless, we provide an informal presentation of the resource interpretation of Linear Logic in Section 2. In addition, the interested reader can see [20, 21] for an illustration of the modeling power of Linear Logic in social choice theory.

We will consider individual resource games defined formally in Section 3. Each player $i$ of a game will be endowed with a multiset of resources $\mathcal{E}_i$. An action for Player $i$ will be to contribute a subset of $\mathcal{E}_i$. An (action) profile specifies a contribution for every players. An outcome will be a context consisting of a multiset of resources resulting from a profile. Then, each player $i$ has a goal $\gamma_i$, which is a resource, represented by one formula of LOG. An outcome $X$ satisfies the goal of Player $i$ if there is a proof of $X \vdash \gamma_i$ in the logic $\text{LOG}$. This will mean that the resources in $X$ can be consumed so as to produce $\gamma_i$.\footnote{Intuitively, we can imagine a game taking place around a table. Each player has an objective to create some resource. Each player has also a bag of resources. To play, each player chooses to take some resources (possibly none) from their respective bags and put them on the table in front of them. The outcome is the collection of resources on the table after every player has chosen. A player is satisfied if we can transform the resources on the table so as to produce her goal. It is a Nash equilibrium when no player has an incentive to take back any resources she put on the table, or to add more resources from her bag.}

What should be an incentive to take back or to add resources? We will study these games of resources two kinds of preferences. We will first consider, in Section 4, preferences over outcomes that are dichotomous. We can thus initially say that Player $i$ prefers an outcome $X$ over an outcome $Y$ iff $X \vdash \gamma_i$ and $Y \not\vdash \gamma_i$. Some formal results will lead us to define in Section 5, parsimonious preferences, a finer notion of preference where $i$ may be qualitatively indifferent between $X$ and $Y$, but still prefer $X$ over $Y$ because $i$’s contribution is strictly less in $X$ than in $Y$.

We will study three decision problems defined also in Section 3, the first of which is deciding whether an action profile is a Nash equilibrium. When dealing with resources, interesting questions arise as to whether some equilibria can be eliminated or constructed by a central authority by redistributing the available resources among the players [9]. In the tradition of social mechanism design, redistribution schemes can be used by a central authority to enforce some behavior, either by disincentivizing a behavior or incentivizing a behavior. Formal frameworks dealing with redistribution schemes and economic policies have been studied [6, 13, 15].

Some profiles that are not equilibria can have desirable outcomes. Some equilibria can have outcomes that are undesirable. Desirability must here be understood from the point of view of a system designer. A system designer can redistribute the resources of the players in a game so as to steer the interaction to or away from a particular outcome.

A redistribution consists in reallocating the resources endowed to the players. To every redistribution corresponds a new game where the players maintain their objectives, but their possible actions have changed. If $G^\epsilon$ is the original game, and $G'$ is a redistribution of the endowment function $\epsilon$, then $G'\epsilon'$ is a new game.

\footnote{Individual resource games were called ideal resource games in [26] to reflect the fact that any subset of the endowments can be used by the players.}

\footnote{Indeed, $X \vdash \gamma_i$ indicates that the resources $X$ are sufficient to produce $\gamma_i$, and $X \vdash \gamma_j$ indicates that the resources $X$ are sufficient to produce $\gamma_j$. It may be however that the resources $X$ are not sufficient to produce $\gamma_i$ and $\gamma_j$ simultaneously.}
We will investigate how resource distribution schemes can contribute to eliminate undesirable game equilibria, and construct desirable game equilibria. They are a form of redistribution of wealth, which consists in wealth being transferred from some individuals to others. In our economies, it exists in the form of social mechanisms such as taxation, public services, and confiscation. Division of property and division of debt in divorce law are good imagery of what a designer can do in the mechanisms we propose in this paper. This toy example will be formalized later.

Example 1. Ann and Bernard have filed for divorce. Ann recently owns an alarm clock that she recently bought. Bernard recently bought a breadmaker. He also owns enough flour to make bread for two year. Ann likes homemade bread, and would like to be able to make her own bread for one year. Bernard wants to keep the alarm clock. In this context, the outcome equilibrium is very likely to be the one where Ann does not use the alarm clock and Bernard retains the breadmaker and the flour. Neither of them satisfy their objective.

However, an arbitrator can redistribute their endowments. He can give the breadmaker and half the flour to Ann, and give the alarm clock to Bernard. Doing so, the outcome where Ann and Bernard do not use any of their endowment can be eliminated. Moreover, a new outcome equilibrium can be constructed where both satisfy their objectives.

We will thus look at two decision problems related to Nash equilibria: rational elimination and rational construction of Nash equilibria.

In a game $G^\epsilon$, a profile can be rationally eliminated from a game if there exists a redistribution $\epsilon'$ of $\epsilon$ such that there is no profile with the same outcome which is a Nash equilibrium in $G^{\epsilon'}$. A profile can be rationally constructed if there exists a redistribution $\epsilon'$ such that there is a profile in $G^{\epsilon'}$ with the same outcome, which a Nash equilibrium.

Outline. We make an brief presentation of Linear Logic in Section 2. We explain how the language can be used to capture a variety of resources which we will put to use in the remained of the paper. We present individual resource games formally in Section 3. We also introduce the decision problems NASH EQUILIBRIUM, RATIONAL ELIMINATION, and RATIONAL CONSTRUCTION. We will use and study two kinds of preferences over action profiles. We define dichotomous preferences in Section 4. We study all three decision problems. We propose general algorithms and general complexity results depending on the complexity of sequent validity in $\text{LOG}$, and on whether $\text{LOG}$ admits the weakening rule or not (that is, whether $\text{LOG}$ is linear or affine). We also illustrate the decision problems with a few small examples. We do the same for parsimonious preferences in Section 6. In particular, we formalize Example 1 in Section 6.2 and solve some problems for it. We briefly explain how individual resources games can form a basis for defining cooperative resource games in Section 7. Some concluding remarks are offered in Section 8.

We provide a technical appendix. Section A presents the sequent rules of Affine MALL. Section B briefly summarizes some elements of computational complexity that can be useful to the reader.

2 Resources and Linear Logic

One contribution of this paper is to show that resource-sensitive logics are a useful tool for studying the formal aspects of resources in game theoretical settings. Another contribution is to demonstrate that it is possible to obtain rather general results for a large class of games of resources depending on the formal properties of the logic $\text{LOG}$ we start with. This offers the opportunity to tailor a game to the needs of a certain application without changing the framework. We can indeed choose any sensible fragment of a resource-sensitive logic.
We will work with some fragments of Linear Logic [8]. The results of this paper will draw upon the proof theory and its rules presented in the Appendix A, even though the technical aspects of the paper can be grasped without a great understanding of the language. However, some intuitions about the resource interpretation of Linear Logic can hopefully contribute to make reading through the remainder less dull.

2.1 Formulas and sequents

A good introduction to Linear Logic and its variants is [25]. We will use logics defined on the language of propositional Linear Logic. The classical tautology splits into the additive $\top$ and the multiplicative $1$. The classical falsum splits into the additive $0$ and the multiplicative $\bot$. The multiplicative conjunction and disjunction are respectively $\otimes$ and $\oplus$. The multiplicative conjunction and disjunction are respectively $\otimes$ and $\oplus$. The linear implication is $A \multimap B$ and combines with the multiplicative conjunction such that $(A \otimes (A \multimap B)) \multimap B$ is a valid principle. The linear negation is $\sim A$.

MLL is the multiplicative fragment, whose language is formalized by the grammar $A ::= 1 | \bot | p \multimap A | A \otimes A \multimap A$, where $p$ is an atomic formula. It only contains the multiplicative connectives. MALL is the fragment with both additive and multiplicative operators $A ::= \top | 0 | \bot | A \otimes A | A \oplus A | A \multimap A | A \& A | A \oplus A$.

We now introduce some terminology and notations. A sequent is a statement $\Gamma \vdash \Delta$ where $\Gamma$ and $\Delta$ are finite multisets of occurrences of formulas of LOG. Often, we can conveniently write a multiset $\{A_1, \ldots, A_n\}$ as the list of formulas $A_1, \ldots, A_n$. Also, we use the notation $\Gamma^* = \bigotimes_{A \in \Gamma} A$ and $\emptyset^* = 1$.

An intuitionistic sequent is a sequent $\Gamma \vdash A$ with only one formula to the right. Sequent provability will play an important part in the technical work of the paper. A sequent $\Gamma \vdash \Delta$ is provable in LOG if there exists a linear proof using the rules of the logic LOG. Intuitively, $\Gamma \vdash \Delta$ being provable means that the resources in $\Gamma$ can be transformed into either of the resources in $\Delta$. If a sequent $\Gamma \vdash \Delta$ is not provable, we can write $\Gamma \not\vdash \Delta$, although we will also often simply write “not $\Gamma \vdash \Delta$”.

In the individual resource games introduced in this paper, the action of a player $i$ consists in making available a multiset $C_i$ of formulas/resources. The outcome of an action is the multiset union of all the individual actions: $\Gamma = \bigcup_i C_i$. The goal of a player is a formula/resource $\gamma$. To decide whether the profile with outcome $\Gamma$ satisfies the goal $\gamma$ of a player, we will evaluate the validity of the (intuitionistic) sequent $\Gamma \vdash \gamma$.

The logic captured by all the rules in the Appendix A is Affine MALL.

A rule that is not part of the calculus is the structural rule of contraction. One rule of contraction (left contraction) says that if something can be proved with two occurrences of $A$, then it can be proved with only one occurrence. Symbolically,

$$
\frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}.
$$

This is prohibited in every resource-sensitive logic. Integrating it into Linear Logic, one consequence would be that $A \vdash A \& A$. If we interpret formulas as resources—as we do—contraction would be a license to duplicate resources at will. (See [22] for a detailed account of logics without contraction.)

We must concede that some of the connectives of MLL and MALL do not have an intuitive interpretation in terms of resources, in and of themselves. This is the case of the multiplicative and the additive falsums ($\bot, 0$), and of the somehow infamous multiplicative disjunction $\otimes$. Fortunately, we do not need them to enjoy the full expressivity of Linear Logic. To see that, Table 1 shows how the connectives interact. From it, it is clear that we can as well make without some language redundancy. The language of MLL is

$$
A ::= 1 \mid p \mid \sim A \mid A \& A \mid A \multimap A,
$$

\footnote{We use $\bigcup$ for the multiset union, and $\bigcup$ for the set union.}
and the language of MALL is

\[ A ::= \top \mid 1 \mid p \mid A \mid A \otimes A \mid A \multimap A \mid A \& A \mid A \oplus A . \]

It suffices to see the other connectives as definitions, following the equivalences of Table 1. We define \( \bot = \sim 1 , 0 = \sim \top \), and \( A \& B = (\sim A) \multimap B \).

### 2.2 Resources as propositions

A resource captured by a proposition of Linear Logic, can be atomic like one mole of hydrogen \( H \) or one mole of oxygen \( O \). It can be a tensor combination of resources, e.g., \( O \otimes O \) being two moles of oxygen. A resource can be a process transforming resources, e.g., \( 2H_2O \otimes H_2O \multimap 2H_2 \otimes H_2 \otimes O_2 \) would be the well known chemical reaction of electrolysis. It consumes two moles of water to produce two moles of dihydrogen and one mole of dioxygen. Working harmoniously with resources and resource transformation processes with this meticulous control over their combination is made possible using resource-sensitive logics. In a game where a player is endowed with \( 2q \) moles of water and a player is endowed with \( q \) processes of electrolysis, it is possible to consume these resources and produce \( 2q \) moles of hydrogen gas and \( q \) of oxygen gas. But not more!

In Section 6.1, we will illustrate our games with an example using chemical reactions. But for the time being, we explain in more details how the refined operators of Linear Logic can be used to formalize and grasp a variety of resources. Table 2 reports possible readings of the connectives.

| \( A \otimes B \) | A and B simultaneously |
| \( A \& B \) | a deterministic choice between A and B; not both |
| \( A \oplus B \) | A or B non-deterministically; not both |
| \( A \multimap B \) | A is sufficient to produce B (losing A in the process) |
| 1 | vacuous resource |
| \( \top \) | some resource |

Table 2: Possible resource interpretations of formulas.
Now, whether the occurrence of a resource indicates a consumption or a production of the resources depends on where a formula appears in the sequent. The sequent of Linear Logic
\[ A \vdash B \]
can be read as

“if you give \( A \) you can receive \( B \)”.

Hence, as it should be expected, we give the resources at the left of the sequent, and receive the resources at the right of the sequent. Table 3 reports possible readings of the sequents. The linear negation allows one to switch the give/receive mode. The sequent \( A \vdash \sim B \) represents “give \( A \) and \( B \), and receive nothing”. The sequent \( A, \sim B \vdash \bot \) represents “give \( A \) and receive \( B \)”.

**Example 2.** A few items can be obtained from vending machine in exchange of money. For instance, giving \$1 you can choose to receive a chocolate bar or a soft-drink. This is captured by

\[ \$1 \vdash \text{chocobar} \& \text{drink} . \]

Also, giving \$0.8 you can receive 2 packs of gum. This is captured by:

\[ \$0.8 \vdash \text{gum} \otimes \text{gum} . \]

In the previous example, the formula chocobar\&drink denotes a deliberative choice between chocobar and drink. One \& the other can be obtained from \$1, but not both. This is significantly different from \$1 \vdash chocobar \& drink which denotes something more akin to the classical disjunction: chocobar or drink can be obtained from \$1. But for all we know, it might be impossible to actually get one or to get the other, and we don’t get to decide.

**Example 3.** We can represent a simple act of gambling. The sequent

\[ \$1 \vdash (\$1 \otimes \$1) \oplus 1 \]

captures the fact that you can give \$1 to receive \$2 or nothing; but you don’t choose what you get.

The next example uses a good part of the connectives.
Example 4. We can capture the fact that $17$ get you a menu:

$17 \vdash \text{menu}$.

The menu consists of a main dish, a side dish, and a dessert:

$\text{menu} \vdash \text{dish} \otimes \text{side} \otimes \text{dessert}$.

As main dish, you can choose between fish and meat:

$\text{dish} \vdash \text{fish} \& \text{meat}$.

The side dish depends on the season; you don’t choose; it is either aubergine, or parsnip with leek, or asparagus:

$\text{side} \vdash \text{aubergine} \oplus (\text{parsnip} \otimes \text{leek}) \oplus \text{asparagus}$.

Finally, as dessert, you choose between the strudel and the chocolate tart. Moreover, you choose whether to have ice cream for $1$ extra.

$\text{dessert} \vdash (\text{strudel} \& \text{chocotart}) \otimes ((\dollar 1 \rightarrow \text{icecream}) \& 1)$.

We have not illustrated the additive unit $\top$ yet. The next example hints at the upcoming formalization of Example 1 in Section 6.2.

Example 5. We can formalize the function of a breadmaker as the resource transformation process flour $\rightarrow$ bread. That is, a breadmaker transforms flour into bread. (Arguably ignoring that we would also need water and electricity. For simplicity, water and electricity could here be considered resources that are provably equivalent to the vacuous resource $1$.) The sequent

$\text{flour}, \text{flour}, \text{flour} \vdash \text{bread} \otimes \top$

indicates that with two ‘tokens’ of flour and a breadmaker, one can make bread, and some resources will remain in excess, viz., flour.

The additive unit $\top$ has some connection with the relationship between linear and affine reasoning that we now discuss briefly.

2.3 Linear vs. affine reasoning and preferences

Weakening (rules $W$) in the Appendix A in the logic $\text{LOG}$ or lack of it thereof, can play a crucial role in the satisfaction of the goals of the players. It will also have striking consequences for the algorithmic solutions of the decision problems that we study in this paper.

In the context of resource-sensitive logic, one rule of weakening (left weakening) says that if something can be obtained from a set of resources then it can also be obtained from more resources. Symbolically,

\[
\Gamma \vdash \Delta \\
\Gamma, A \vdash \Delta
\]

Weakening gives a monotonic flavor to the process of deduction in the logic. Following the terminology in Linear Logic, a logic $\text{LOG}$ admitting weakening will be said to be affine and a logic $\text{LOG}$ without weakening will just be said to be linear.

Despite the fact the Affine Logic more inference rules than Linear Logic, the unit $\top$ allows one to simulate the reasoning in Affine Logic with the provability of Linear Logic. Indeed, the sequent $\Gamma \vdash A$ is
provable in a logic LOG with the rule of weakening iff the sequent \( \Gamma \vdash A \otimes \top \) is provable the logic LOG without using weakening.

In the affine case, \( A, B \vdash A \) is a provable sequent. If a player has a goal \( \gamma = A \), then she will find her objective satisfied with an outcome \( \{A, B\} \). In the linear case, we have in general \( A, B \nvdash A \) (unless \( B \) is a vacuous resource equivalent to \( 1 \)). A player with a goal \( \gamma = A \) will not be satisfied with an outcome \( \{A, B\} \) as she wants \( A \) and nothing more. If she is indeed indifferent to leftover resources, her goal can be expressed as \( \gamma = A \otimes \top \), when LOG is linear.

2.4 Some complexities

Before moving to the technical part of this paper, we quickly summarize the complexity of some fragments and variants of Linear Logic that could be used as the LOG parameter in our analysis resource games.\(^4\) The results of this paper will be applicable to every fragment mentioned here. MALL is PSPACE-complete; MLL is NP-complete; Affine MLL is NP-complete; Affine MALL is PSPACE-complete; Intuitionistic MALL is PSPACE-complete; Intuitionistic MLL is NP-complete. Something particularly interesting is that these fragments of Linear Logic behave well computationally also in the first-order case. First-Order MLL is NP-complete and First-Order MALL is NEXPTIME-complete. See [14, 10].

3 Individual resource games and decision problems

We formally define our models of individual resource games.

**Definition 6.** An individual resource game (IRG) is a tuple \( G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n) \) where:

- \( N = \{1, \ldots, n\} \) is a finite set of players;
- \( \gamma_i \) is a formula of LOG (i’s goal, or objective);
- \( \epsilon_i \) is a finite multiset of formulas of LOG (i’s endowment).

Let \( G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n) \), we define: the set of possible actions of \( i \) as the set of multisets \( \text{ch}_i(G) = \{C \mid C \subseteq \epsilon_i\} \), and the set of profiles in \( G \) as \( \text{ch}(G) = \prod_{i \in N} \text{ch}_i(G) \). When \( P = (C_1, \ldots, C_k) \in \text{ch}(G) \) and \( 1 \leq i \leq k \), then \( P_{-i} = (C_1, \ldots, C_{i-1}, C_{i+1}, \ldots, C_k) \). That is, \( P_{-i} \) denotes \( P \) without player \( i \)’s contribution. The outcome of a profile \( P = (C_1, \ldots, C_n) \) is given by the multiset of resources \( \text{out}(P) = \bigcup_{1 \leq i \leq n} C_i \).

We will define “\( i \) strongly prefers \( P^* \)” in due time, reflecting dichotomous preferences first (Sec. 4) and parsimonious preferences second (Sec. 5).

**Definition 7.** Let \( G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n) \). A profile \( P \in \text{ch}(G) \) is a Nash equilibrium iff for all \( i \in N \) and for all \( C_i \in \text{ch}_i(G) \), we have that \( i \) does not strongly prefer \( (P_{-i}, C_i) \) over \( P \).

Let us note \( NE(G) \) the set of Nash equilibria in \( \text{ch}(G) \).

A basic decision problem is the one of determining whether a choice profile is a Nash equilibrium.

**NASH EQUILIBRIUM (NE)**

**(in)** An individual resource game \( G \) and \( P \in \text{ch}(G) \).

**(out)** \( P \in NE(G) \)?

\(^4\)See Appendix B for some elements of complexity that will be useful in the proofs in this paper.
Some profiles that are not equilibria can have desirable outcomes. Some equilibria can have outcomes that are undesirable. Hence, it is interesting to investigate how resource distribution schemes influence how undesirable game equilibria can be eliminated and how desirable game equilibria can be constructed.

In the tradition of social mechanism design, redistribution schemes can be used by a central authority to enforce some behavior, either by disincentivizing a behavior or incentivizing a behavior.

We will study redistribution schemes in individual resource games. Let \( \epsilon \) be an endowment function such that for every player \( i \) we have \( \epsilon(i) = \epsilon_i \), a multiset of formulas of LOG. A redistribution scheme of \( \epsilon \) is an endowment function \( \epsilon' \) such that

\[
\bigcup_{i \in N} \epsilon(i) = \bigcup_{i \in N} \epsilon'(i).
\]

We note \( \operatorname{redis}(\epsilon) \) the set of redistributions of the endowment function \( \epsilon \).

Given the individual resource game \( G_\epsilon = (N, \gamma_1, \ldots, \gamma_n, \epsilon(1), \ldots, \epsilon(n)) \) we can apply a redistribution scheme where we modify the endowment function \( \epsilon \) into \( \epsilon' \). We thus obtain the individual resource game \( G_{\epsilon'} = (N, \gamma_1, \ldots, \gamma_n, \epsilon'(1), \ldots, \epsilon'(n)) \).

We will investigate two decision problems inspired by [9], which are related to resource redistributions. We will look at whether the outcome of a resource game can be rationally eliminated. That is whether there is a resource redistribution such that no Nash equilibrium of the new resource game yields this outcome.

**RATIONAL ELIMINATION (RE)**

**in** An individual resource game \( G_\epsilon \) and \( P \in \operatorname{ch}(G_\epsilon) \).

**out** Is there a redistribution \( \epsilon' \) of \( \epsilon \) such that for all \( P' \in \operatorname{ch}(G_{\epsilon'}) \), if \( \operatorname{out}(P') = \operatorname{out}(P) \) then \( P' \notin \operatorname{NE}(G_{\epsilon'}) \)?

Conversely, we will look at whether the outcome of a resource game can be rationally constructed. That is whether there is a resource redistribution such that the outcome is the outcome of some Nash equilibrium in the new resource game.

**RATIONAL CONSTRUCTION (RC)**

**in** An individual resource game \( G_\epsilon \) and \( P \in \operatorname{ch}(G_\epsilon) \).

**out** Is there a redistribution \( \epsilon' \) of \( \epsilon \) such that there is \( P' \in \operatorname{ch}(G_{\epsilon'}) \) where \( \operatorname{out}(P') = \operatorname{out}(P) \) and \( P' \in \operatorname{NE}(G_{\epsilon'}) \)?

Note that being a game equilibrium is without ambiguity a property of profile. However, after a redistribution of resources in an individual resource game, the space of actions and the space of profiles change. Thus, elimination and construction are more about the outcomes of profiles. Section 4.2 and Section 5.1 will illustrate these decision problems in due time.

### 4 Dichotomous preferences

Let \( G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n) \) be an individual resource game. Player \( i \), whose goal is \( \gamma_i \), realizes her objectives in a profile \( P \) when \( \operatorname{out}(P) \vdash \gamma_i \). That is, the resources in \( \operatorname{out}(P) \) can be transformed into a shareable resource \( \gamma_i \). For \( P \in \operatorname{ch}(G) \) and \( Q \in \operatorname{ch}(G) \), we say that player \( i \in N \) (dichotomously) strongly prefers \( P \) over \( Q \) (noted \( Q \prec_i P \) iff \( \operatorname{out}(P) \vdash \gamma_i \) and not \( \operatorname{out}(Q) \vdash \gamma_i \).
Proposition 8. Let $G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)$ be an individual resource game, two profiles $P \in \text{ch}(G)$ and $Q \in \text{ch}(G)$, and a player $i \in N$. When sequent validity in LOG is NP-complete, deciding whether $Q \prec_i P$ is an NP \& coNP = BH$_2$-complete. When sequent validity in LOG is PSPACE-complete, deciding whether $Q \prec_i P$ is PSPACE-complete.

Proof. The language corresponding to the problem is $L = \{(P, Q) \mid Q \prec_i P\} = L_1 \cap L_2$ with $L_1 = \{(P, Q) \mid \text{out}(P) \vdash \gamma_i\}$, and $L_2 = \{(P, Q) \mid \text{not out}(Q) \vdash \gamma_i\}$. In particular, when the problem of sequent validity of LOG is in NP, we clearly have that $L_1$ is a NP language and $L_2$ is a coNP language.

For hardness, consider the decision problem VALID-NONVALID that takes in input two sequents of LOG $\Gamma_1 \vdash \Delta_1$ and $\Gamma_2 \vdash \Delta_2$, and outputs true iff $\Gamma_1 \vdash \Delta_1$ is valid and $\Gamma_2 \vdash \Delta_2$ is not valid. It is easy to see that if LOG is NP-complete, then VALID-NONVALID is BH$_2$-complete, and if LOG is PSPACE-complete, then VALID-NONVALID is PSPACE-complete.

We propose a reduction of VALID-NONVALID into the problem of deciding whether in an individual resource game, a profile is strongly preferred to another profile by a player.

Let $\Gamma_1 \vdash \Delta_1$ and $\Gamma_2 \vdash \Delta_2$ be two sequents of LOG. We can prove using $\perp L$, $\perp R$, (cut) and (E) that $\Gamma \vdash \Delta$ iff $\Gamma \vdash \Delta_1$. Thus, we have $\Gamma_1 \vdash \Delta_1$ iff $\Gamma_1 \vdash \perp$, and we have $\Gamma_2 \vdash \Delta_2$ iff $\Gamma_2 \vdash \perp$. Now we construct the game $G = (\{1\}, \gamma_1 = \perp, \epsilon_1 = \Gamma_1 \vdash \perp \Delta_1 \vdash \perp \Delta_2)$. It is now easy to see that VALID-NONVALID instantiated with $\Gamma_1 \vdash \Delta_1$ and $\Gamma_2 \vdash \Delta_2$ returns true iff Player 1 strongly prefers $\Delta_1$ over $\Delta_2$ in $G$. \hfill \square

4.1 Finding Nash equilibria

We study the complexity of the problem NASH EQUILIBRIUM with dichotomous preferences.

4.1.1 Hardness

We are about to prove the lower bound of the complexity NE with dichotomous preferences. Before we do so, observe that by applying the rules $L \sim$ and $R \sim$,

$$A_1, \ldots, A_n \vdash B_1, \ldots, B_m \iff A_1, \ldots, A_n, \sim B_2, \ldots, \sim B_m \vdash B_1$$

is immediate. Hence, we can, without loss of generality, consider only the intuitionistic sequents of LOG in the many-to-one reductions of this paper.

Proposition 9. NE is as hard as the problem of checking sequent validity in LOG, even when there is only one player.

Proof. We reduce the problem of sequent validity for the logic LOG. W.l.o.g., we only consider intuitionistic sequents. Let $\Gamma \vdash \delta$ be the intuitionistic sequent where $\Gamma$ is an arbitrary multiset of formulas of LOG and $\delta$ is an arbitrary formula.

We can construct the individual resource game $G$ such that $G = (\{1\}, \delta, \Gamma \cup \{\delta\})$. $G$ is thus the one-player individual resource game where Player 1’s goal is to achieve $\delta$, and Player 1 is endowed with $\Gamma \cup \{\delta\}$ (this is a set union but we could have chosen the endowment $\Gamma \uplus \{\delta\}$ as well). A profile in $G$ is a choice of Player 1, that is, a subset $C_1$ of $\Gamma \cup \{\delta\}$. In this case for any profile $P$ in $G$, $\text{out}(P) = P$.

We show that $\Gamma \vdash \delta$ iff $\Gamma \in NE(G)$.

From left to right, suppose that $\Gamma \vdash \delta$. We need to show that $\Gamma \in NE(G)$. That is, for all $C_1 \subseteq \Gamma \cup \{\delta\}$, if $C_1 \vdash \delta$ then $\Gamma \vdash \delta$. Since we supposed $\Gamma \vdash \delta$, this is trivially true.

From right to left, suppose that $\Gamma \in NE(G)$. This means that for all $C_1 \subseteq \Gamma \cup \{\delta\}$, if $C_1 \vdash \delta$ then $\Gamma \vdash \delta$. Let in particular $C_1 = \{\delta\}$. Indeed, $C_1 \subseteq \Gamma \cup \{\delta\}$. Moreover, by (ax) we have $\delta \vdash \delta$. Hence, $\Gamma \vdash \delta$ follows. \hfill \square

---

For $\Gamma = \{A_1, \ldots, A_k\}$ we note $\sim \Gamma$ the set $\{\sim A_1, \ldots, \sim A_k\}$. 

10
4.1.2 Algorithms

To establish an upper bound on the complexity of NE let us first outline an algorithm for solving its complement. That is, checking whether a profile is not a Nash equilibrium. Let $P \in \text{ch}(G)$ be a profile.

To determine whether $P \notin NE(G)$, we can employ a simple non-deterministic algorithm, showed as Algorithm 1.

Algorithm 1 General algorithm for co-NE

1: non-deterministically guess $(i, C_i) \in \mathbb{N} \times \text{ch}(G)$.
2: return $\prec_i (P_{-i}, C_i)$.

Proposition 10. If the problem of sequent validity checking of LOG is in NP then NE is in coNP$^{BH_2}$ and indeed in $\Pi_2^P$. If the problem of sequent validity checking of LOG is in PSPACE then NE is in PSPACE.

Proof. Consider Algorithm 1. If sequent validity in LOG is in NP, we can check $P \prec_i (P_{-i}, C_i)$ in BH$_2$ (Prop. 8). Thus we can check whether $P \notin NE(G)$ in NP$^{BH_2}$. Finally, we can solve NE in coNP$^{BH_2}$. It is the case that BH$_2 \subseteq \Delta_2^p$, and also that NP$^{\Delta_2^p} = \Sigma_2^p$ so we can solve NE in $\Pi_2^P$. The proof for the case of PSPACE also proceeds as an analysis of Algorithm 1.

Affine logic admits the rule of weakening ($W$), which allows one to discard resources. In this setting, if a player can achieve her goal with the resources $\Gamma$, she can as well achieve her goal with the resources $\Gamma \cup \{A\}$. A consequence is the following lemma, which will have a significant impact on the computational complexity of NE.

Lemma 11. Let $G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)$ be an individual resource game. When LOG is affine, $P \notin NE(G)$ iff $\exists i \in \mathbb{N}: P \prec_i (P_{-i}, \epsilon_i)$.

Proof. Suppose $P \notin NE(G)$. There is $i \in \mathbb{N}$ and $C_i \in \text{ch}(G)$ s.t. $P \prec_i (P_{-i}, C_i)$. By definition, out($(P_{-i}, C_i)$) $\vdash \gamma_i$ and out$(P) \not\vdash \gamma_i$. We have $C_i \subseteq \epsilon_i$, so by applying weakening ($W$) with every instance of formulas in $\epsilon_i \setminus C_i$, we can prove that out($(P_{-i}, \epsilon_i)$) $\vdash \gamma_i$. We thus have that there is $i \in \mathbb{N}$ s.t. $P \prec_i (P_{-i}, \epsilon_i)$. The other way around is immediate from the definition of Nash equilibria.

It means that, in a profile, if no player has an incentive to deviate by making available their whole endowment, then the profile is a Nash equilibrium. The very profile where all the players make available their whole endowment is trivially such a profile. The next proposition follows immediately:

Proposition 12. Let $G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)$ be an individual resource game. When LOG is affine: $NE(G) \neq \emptyset$ and $(\epsilon_1, \ldots, \epsilon_n) \in NE(G)$.

Lemma 11 also helps us to establish the following result.

Proposition 13. When LOG is affine, if the problem of sequent validity checking of LOG is in NP then NE is in P$^{NP}$[1]. If the problem of sequent validity checking of LOG is in PSPACE then NE is in PSPACE.

Proof. Let $G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)$ be an individual resource game and let $P \in \text{ch}(G)$ be a profile. One can check whether $P \in NE(G)$ with Algorithm 2.

For correctness, note that the instructions of the lines 2–4 are equivalent to a test of whether out$(P) \not\vdash \gamma_i$ and out$(P_{-i}, \epsilon_i) \vdash \gamma_i$, that is, $P \prec_i (P_{-i}, \epsilon_i)$. Lemma 11 ensures that exactly when there is an $i \in N$ such that $P \prec_i (P_{-i}, \epsilon_i)$ we can conclude that $P$ is not a Nash equilibrium.

Suppose sequent validity in LOG is in NP. The algorithm can be simulated by a deterministic oracle Turing machine in polynomial time with $2n$ non-adaptive queries to an NP oracle. Indeed, $P \in NE(G)$ is thus a P$^{NP}$[1][2n] predicate. The problem is in P$^{NP}$[1]. When sequent validity in LOG is in PSPACE, we obtain a complexity of PSPACE = PSPACE.
Algorithm 2: Algorithm for NE with dichotomous preferences and affine LOG

1: for each $i \in N$ do:
2:  if $(\text{out}(P) \vdash \gamma_i)$:
3:    continue;
4:  else if $(\text{out}((P_{-i}, \epsilon_i)) \vdash \gamma_i)$:
5:    return false.
6: return true.

4.2 Elimination

A very simple illustration of RATIONAL ELIMINATION is given by the individual resource game $G^\epsilon = (\{1, 2\}, \gamma_1 = B; \gamma_2 = A, \{A\}, \{B\})$. There are two players. Player 1 wants $B$ but is endowed with $\{A\}$, while Player 2 wants $A$ but is endowed with $\{B\}$. The game $G^\epsilon$ can be represented as on Figure 1. (We indicate the realized objectives assuming that LOG is affine.)

![Figure 1: The game $G^\epsilon$.](image)

Figure 1: The game $G^\epsilon$. $\gamma_1$ and $\gamma_2$ indicate that Player 1 and Player 2 have their goals satisfied, assuming LOG is affine. The symbol $\square$ denotes a Nash Equilibrium.

One can readily check that all profiles are Nash equilibria. However, the profile $(\{A\}, \{B\})$ is more ‘socially desirable’ than the others since it satisfies both players’ goal.

A centralized authority could effectively eliminate the others by redistributing the resources present in $G^\epsilon$ so as to obtain $G^\epsilon' = (\{1, 2\}, \gamma_1 = B, \gamma_2 = A, \{A\}, \{B\})$. The game $G^\epsilon'$ can be represented as on Figure 2.

![Figure 2: The game $G^\epsilon'$.](image)

Figure 2: The game $G^\epsilon'$.

The only Nash equilibrium is now the one with outcome $\{B, A\}$.

4.2.1 Algorithms

As a consequence of Prop. 12, we already know that:

**Proposition 14.** Let $G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)$ be an individual resource game. When LOG is affine, the profile $P$ such that $\text{out}(P) = \biguplus_j \epsilon_j$ is not rationally eliminable.
This is very specific to the affine case (and dichotomous preferences), and even then, it is of course not true of all Nash equilibria. To decide whether some outcome is rationally eliminable, one naïve approach consists in trying all possible redistributions and check whether the outcome is a Nash equilibrium in the resulting individual resource game. Instead, we are going to exploit a pleasant property, analogous to [9, Corollary 4].

Let \( G^\epsilon = (N, \gamma_1, \ldots, \gamma_n, \epsilon(1), \ldots, \epsilon(n)) \) be an individual resource game. For each player \( i \in N \), we define \( G^{\epsilon[i]} \) where \( [\epsilon \triangleright i] \) is the redistribution of \( \epsilon \) where all resources are assigned to \( i \), that is:

\[
[\epsilon \triangleright i](j) = \begin{cases} 
\bigcup_{k \in N} \epsilon(k) & \text{when } j = i \\
\emptyset & \text{otherwise.}
\end{cases}
\]

Because there is only one active player in \( G^{\epsilon[i]} \), we will sometimes write a profile of \( G^{\epsilon[i]} \) as \( (C_i) \) with \( C_i \in \text{ch}_i(G^{\epsilon[i]}) \) instead of \( (\emptyset, \ldots, \emptyset, C_i, \emptyset, \ldots, \emptyset) \), by abuse of notation.

**Lemma 15.** Let \( G^\epsilon \) be an individual resource game and \( P \in \text{ch}(G^\epsilon) \). \( P \) is rationally eliminable iff there is a player \( i \in N \) and a profile \( Q \in \text{ch}(G^{\epsilon[i]}) \), such that \( \text{out}(Q) = \text{out}(P) \) and \( Q \not\subseteq \mathcal{N}E(G^{\epsilon[i]}) \).

**Proof.** From right to left. Suppose \( Q \not\subseteq \mathcal{N}E(G^{\epsilon[i]}) \) for some \( i \in N \). Let also \( P \in \text{ch}(G^\epsilon) \) be a profile and assume \( \text{out}(P) = \text{out}(Q) \). When there is at most one player with a non-empty endowment, as in \( [\epsilon \triangleright i] \), there is a one-to-one correspondence between the set of profiles and the set of outcomes. Thus, there is one and only one profile in \( G^{\epsilon[i]} \) with outcome \( \text{out}(P) \) and it is \( Q \). So there is a redistribution of \( \epsilon \), namely \( [\epsilon \triangleright i] \), such that for all profiles \( Q \in \text{ch}(G^{\epsilon[i]}) \) with outcome \( \text{out}(P) \), we have \( Q \not\subseteq \mathcal{N}E(G^{\epsilon[i]}) \). So \( P \) is rationally eliminable.

From left to right. Suppose that \( P \) is rationally eliminable. Thus, there is a redistribution \( \epsilon' \) of \( \epsilon \) such that for all \( P' \in \text{ch}(G^\epsilon) \), if \( \text{out}(P') = \text{out}(P) \) then \( P' \not\subseteq \mathcal{N}E(G^\epsilon) \). So let \( R \in \text{ch}(G^\epsilon) \) be an arbitrary profile with \( \text{out}(R) = \text{out}(P) \). By assumption, we have that \( R \not\subseteq \mathcal{N}E(G^\epsilon) \). By definition of Nash equilibria, this means that there is \( i \in N \) and \( C_i \in \text{ch}_i(G^\epsilon) \) such that \( R \not\prec_i (R_{-i}, C_i) \). Now consider the game \( G^{\epsilon[i]} \). We have \( \text{out}(R) \in \text{ch}_i(G^{\epsilon[i]}) \) and \( \text{out}((R_{-i}, C_i)) \in \text{ch}_i(G^{\epsilon[i]}) \). Let the profile \( R^1 \in \text{ch}(G^{\epsilon[i]}) \) with \( R^1 = \text{out}(R) \) and \( R^1_j = \emptyset \) when \( j \neq i \). Let \( R^2 \in \text{ch}(G^{\epsilon[i]}) \) be the profile with \( R^2_i = \text{out}((R_{-i}, C_i)) \) and \( R^2_j = \emptyset \) when \( j \neq i \). Since, \( R \not\prec_i (R_{-i}, C_i) \), we also have \( R^1 \not\prec_i R^2 \). So \( R^1 \not\subseteq \mathcal{N}E(G^{\epsilon[i]}) \). The profile \( R^1 \) is the only profile of \( G^{\epsilon[i]} \) with outcome \( \text{out}(P) \). So we can conclude.

We establish an upper bound on the complexity of \( \mathcal{R}E \) when \( \log \) does not admit the weakening rule.

**Proposition 16.** When \( \text{LOG} \) is linear, \( \mathcal{R}E \) is in \( \text{NP}^{\text{BH}_{\text{L}}} \) and indeed in \( \Sigma_2^p \) when \( \text{LOG} \) is in \( \text{NP} \), and in \( \text{PSPACE} \) when \( \text{LOG} \) is in \( \text{PSPACE} \).

**Proof.** Let \( P \in \text{ch}(G^\epsilon) \) be a profile. To determine whether \( P \) is rationally eliminable, we can use Algorithm 3.

**Algorithm 3** General algorithm for \( \mathcal{R}E \)

1: non-deterministically guess \( (i, C'_i) \in N \times \text{ch}_i(G^{\epsilon[i]}) \).
2: return \( P \not\prec_i (P_{-i}, C'_i) \).

Straightforwardly, it guesses a player \( i \) and a deviation in the game \( G^{\epsilon[i]} \) for Player \( i \) from the profile \( \text{out}(P) \in \text{ch}(G^{\epsilon[i]}) \), and checks whether Player \( i \) has an incentive to do this deviation. By Lemma 15, if such a player and deviation exist and only if they exist, the profile \( P \) is rationally eliminable in \( G^\epsilon \). So the algorithm is correct. It can of course be simulated by a non-deterministic oracle Turing machine with one call to an oracle for \( P \not\prec_i (P_{-i}, C'_i) \). Prop. 8 informs us of a containing class of this oracle.
When \( \text{LOG} \) admits the weakening rule, we can propose a surprisingly simple algorithm, which takes advantage of both Lemma 11 and Lemma 15.

**Proposition 17.** When \( \text{LOG} \) is affine, \( \text{RE} \) is in \( \mathbf{P} \) when \( \text{LOG} \) is in \( \mathbf{NP} \), and in \( \mathbf{PSPACE} \) when \( \text{LOG} \) is in \( \mathbf{PSPACE} \).

**Proof.** Let \( G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n) \) be an individual resource game and let \( P \in \text{ch}(G) \) be a profile. Consider Algorithm 4.

**Algorithm 4** Algorithm for \( \text{RE} \) with dichotomous preferences and affine \( \text{LOG} 

1: for each \( i \in N \) do:
2: if \((P \prec_i ([\epsilon \triangleright i](i)))\):
3: return true.
4: return false.

The algorithm is correct. Indeed, by Lemma 15, \( P \) is eliminable in \( G \) iff there is \( i \in N \) where \((\text{out}(P)) \not\in NE(G^{\epsilon \triangleright i})\). By Lemma 11, we know that \((\text{out}(P)) \not\in NE(G^{\epsilon \triangleright i})\) iff \( P \prec_i ([\epsilon \triangleright i](i)) \). Thus, it can be simulated by a deterministic oracle Turing machine in polynomial time with at most \( 2^n \) non-adaptive queries to an oracle for the problem of sequent validity. When the problem of sequent validity in \( \text{LOG} \) is in \( \mathbf{NP} \) it yields a complexity of \( \mathbf{NP} \). When it is in \( \mathbf{PSPACE} \), it yields a complexity of \( \mathbf{PSPACE} \).

### 4.2.2 Hardness

The linear and affine cases both use the same proof strategy which we present at once.

**Proposition 18.** \( \text{RE} \) is as hard as the problem of checking sequent invalidity in \( \text{LOG} \).

**Proof.** Let \( \Gamma \vdash \delta \) be an arbitrary intuitionistic sequent. Let \( \varphi = \Gamma^* \rightarrow \delta \). (Remember that \( \Gamma^* = \bigotimes_{A \in \Gamma} A \).) Let \( G' = (\{1, 2\}, \varphi, 1, \emptyset, \{\varphi\}) \) be an individual resource game. So, we have \( \epsilon_1 = \emptyset \) and \( \epsilon_2 = \{\varphi\} \). There is only one other distinct redistribution \( \epsilon' \) of \( \epsilon \) where \( \epsilon'_1 = \{\varphi\} \) and \( \epsilon'_2 = \emptyset \). It is the case that \( \text{redis}(\epsilon) = \{\epsilon, \epsilon'\} \). Let \( G'' = (\{1, 2\}, \varphi, 1, \{\varphi\}, \emptyset) \) be the individual resource game resulting from the redistribution \( \epsilon' \). Both games are represented on Figure 3.

![Figure 3](image-url)

**Figure 3:** Games \( G' \) and \( G'' \). The profile \((\emptyset, \emptyset)\) is a Nash equilibrium in \( G' \). The profile \((\emptyset, \emptyset)\) is a Nash equilibrium in \( G'' \) iff \( \Gamma \vdash \delta \). (The profile \((\{\varphi\}, \emptyset)\) is a Nash equilibrium in \( G'' \). Depending on whether \( \Gamma \vdash \delta \) and whether \( \text{LOG} \) is linear or affine, \((\emptyset, \{\varphi\})\) may or may not be Nash equilibria in \( G'' \). This is inconsequential for the reduction in the proof of Prop. 18.)

We show that both in the case of linear and of affine logics, we have \( \Gamma \vdash \delta \) iff \((\emptyset, \emptyset)\) is rationally eliminable in \( G'' \).

We first show that

\[
\Gamma \vdash \delta \iff \emptyset \vdash \varphi .
\]
From left to right, suppose $\Gamma \vdash \delta$. By applying $\otimes L$ enough times we obtain $\Gamma^* \vdash \delta$. Then we obtain $\emptyset \vdash \Gamma^* \supset \delta$ using $\rightarrow R$. From right to left, suppose $\emptyset \vdash \Gamma^* \supset \delta$. With (ax) and $\otimes R$ we can show $\Gamma \vdash \Gamma^*$. Using $\otimes R$ on the sequents $\Gamma \vdash \Gamma^*$ and $\emptyset \vdash \Gamma^* \supset \delta$ we obtain

$$\Gamma \vdash \Gamma^* \otimes \Gamma^* \supset \delta.$$  \hspace{1cm} (2)

Without assumption we can also show

$$\Gamma^* \otimes \Gamma^* \supset \delta \supset \delta,$$  \hspace{1cm} (3)

using the rules (ax), $\rightarrow L$, and $\otimes L$. We conclude that $\Gamma \vdash \delta$ using (cut) on the sequents 2 and 3.

We can proceed. Suppose $\Gamma \not\vdash \delta$. We show that $(\emptyset, \emptyset)$ is not a Nash equilibrium in $G'$. Since $\Gamma \not\vdash \delta$, we also have $\emptyset \not\vdash \varphi$ (by Equation 1). On the other hand, using (ax), we have $\{\varphi\} \vdash \varphi$. So in the profile $(\emptyset, \emptyset)$, Player 1 has an incentive to deviate to the profile $(\{\varphi\}, \emptyset)$. So $(\emptyset, \emptyset)$ is not a Nash Equilibrium in $G'$.

Suppose $\Gamma \vdash \delta$. We show that $(\emptyset, \emptyset)$ is a Nash equilibrium both in $G^*$ and in $G'$. In $G^*$, we have $\emptyset \vdash 1$ from $1R$, so Player 2 has no incentive to deviate from the profile $(\emptyset, \emptyset)$ in $G^*$. Moreover, Player 1 is dummy in $G^*$. So $(\emptyset, \emptyset)$ is a Nash Equilibrium in $G^*$.

In $G'$, since $\Gamma \vdash \delta$, we also have $\emptyset \vdash \varphi$ (by Equation 1), so Player 1 has no incentive to deviate from the profile $(\emptyset, \emptyset)$ in $G'$. Moreover, Player 2 is dummy in $G'$. So $(\emptyset, \emptyset)$ is a Nash Equilibrium in $G'$.

\[\square\]

4.3 Construction

For elimination, Lemma 15 provided a remarkable necessary and sufficient condition for the rational eliminability of a profile. For the rational constructibility of a profile, we can only indicatively provide sufficient conditions. Let $G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)$ be an IRG, and let $P \in \text{ch}(G)$ be a profile in $G$. If there is a player $i \in N$ such that out($P$) $\supset \gamma_i$, then $P$ can be rationally constructed by redistributing all the resources to Player $i$. Also, if there a player $i \in N$ such that $\bigcup_{k \in N} \epsilon_k \not\vdash \gamma_i \otimes \top$, then $P$ can be rationally constructed by redistributing all the resources to Player $i$.

We tackle the complexity of RATIONAL CONSTRUCTION with dichotomous preferences.

4.3.1 Hardness

We prove a lower bound of the problem RC in presence of dichotomous preferences.

**Proposition 19.** RC is as hard as the problem of checking sequent validity in LOG.

**Proof.** Let $\varphi = \Gamma^* \supset \delta$ and $G = (\{1\}, \varphi, \epsilon_1 = \{\varphi\})$. We can see that $(\emptyset) \in \text{NE}(G)$ iff $\emptyset \vdash \varphi$, that is $\Gamma \vdash \delta$. As $\text{redis}(\epsilon) = \{\epsilon\}$, we conclude that: for every sequent $\Gamma \vdash \delta$, $(\emptyset)$ is rationally constructible in $G$ iff $\Gamma \vdash \delta$ is provable. \[\square\]

4.3.2 Algorithms

Let $G^*$ be an individual resource game, and let $P \in \text{ch}(G^*)$. To decide whether the profile $P$ can be rationally constructed we can use Algorithm 5. This algorithm will serve for all cases of rational construction in this paper.

The algorithmic analysis is rather simple: we use the problem NE as a blackbox, for which complexity upper bounds have been established in Prop. 10 and Prop. 13.

**Proposition 20.** When LOG is in NP, RC is in $\Sigma^P_3$. When LOG is in PSPACE, RC is in PSPACE.
Proposition 23. If a profile $P$ is a Nash equilibrium in presence of parsimonious preferences, then $P$ is a Nash equilibrium in presence of dichotomous preferences.

Proof. Let $\prec^d_i$ (resp., $\prec^p_i$) denote Player $i$’s parsimonious (resp., dichotomous) preferences; Let $NE_d(G)$ (resp., $NE_p(G)$) denote the set of Nash equilibria in $G$ when considering dichotomous (resp., parsimonious) preferences. Now suppose that $P \in NE_p(G)$. That is, for every $i \in N$ and for every $C_i \in ch_i(G)$ we have not $P \prec^d_i (C_i, P_{-i})$. So $P \notin NE_d(G)$.

Proposition 23 indicates that every Nash equilibrium in presence of parsimonious preference is also a Nash equilibrium in presence of dichotomous preference. The next proposition, which will help us later to prove some hardness result, says that the other way around holds when the profile is the one where every player plays the empty set of resources.

5 Parsimonious preferences

Weakening ($W$) is sometimes a desirable property of LOG and of our preferences of resources. However, it has the untoward consequence of incentivizing players to spend all their resources in individual resource games with dichotomous preferences. This is well exemplified for instance by Prop. 12.

We can teach our players parsimony by attaching to them finer preferences that take into account the realization of their objective, but also the optimality of their contribution.

In an individual resource game $G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)$, we now say that player $i \in N$ (parsimoniously) strongly prefers $P \in ch(G)$ over $Q \in ch(G)$ (noted $Q \prec_i P$) iff one of the following conditions is satisfied:

1. not $\text{out}(P) \vdash \gamma_i$ and not $\text{out}(Q) \vdash \gamma_i$ and $P_i \subset Q_i$;
2. $\text{out}(P) \vdash \gamma_i$ and not $\text{out}(Q) \vdash \gamma_i$;
3. $\text{out}(P) \vdash \gamma_i$ and $\text{out}(Q) \vdash \gamma_i$ and $P_i \subset Q_i$.

Similar preferences have been called pseudo-dichotomous in the literature.

We recognize that the second condition corresponds to profile $P$ being dichotomously strongly preferred by Player $i$ to profile $Q$. The following proposition is a simple consequence.

Proposition 22. If Player $i$ dichotomously strongly prefers $P$ over $Q$ then Player $i$ parsimoniously strongly prefers $P$ over $Q$.

This has another immediate consequence on Nash equilibria.

Proposition 21. If LOG is affine, when LOG is in NP, RC is in $\Sigma^P_2$. When LOG is in PSPACE, RC is in PSPACE.

Proof. The proof is similar to the one of Prop. 20, using the result of Prop. 13 and the fact that $\text{NP}^{\text{NP[log]}} \subseteq \text{NP}^{\Sigma^P_2} = \Sigma^P_2$.

Proof. When LOG is in NP, from Prop. 10, we know that the test of line 2 is $\Pi^P_2$-easy. So RC is in $\text{NP}^{\Pi^P_2} = \Sigma^P_3$. The case for LOG in PSPACE is similar.

Again, an affine LOG seems to bring some relative algorithmic ease.

Proposition 23. If a Nash equilibrium in presence of dichotomous preferences. Now suppose that $P \in \text{NE}_d(G)$, and for every $i \in N$ and for every $C_i \in ch_i(G)$ we have not $P \prec^p_i (C_i, P_{-i})$. So $P \in \text{NE}_d(G)$.
Proposition 24. The profile \((\emptyset, \ldots, \emptyset)\) is a Nash equilibrium in presence of parsimonious preferences if it is a Nash equilibrium in presence of dichotomous preferences.

Proof. Left to right is a consequence of Prop. 23. For right to left, assume \((\emptyset, \ldots, \emptyset)\) is in \(NE_d(G)\). With parsimonious preferences, the only incentive to deviate from a Nash equilibrium in presence of dichotomous preference, would be to play a smaller multiset of resources. This is impossible in \((\emptyset, \ldots, \emptyset)\). \(\square\)

We now address the complexity of the decision problem of deciding whether a player parsimoniously strongly prefers a profile over another profile.

Proposition 25. Let \(G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)\) be an individual resource game. Let also \(P \in \text{ch}(G)\) and \(Q \in \text{ch}(G)\) be two profiles, and \(i \in N\) be a player. When sequent validity in \(\text{LOG}\) is in \(\text{NP}\), deciding whether \(Q \prec_i P\) is a problem in \(P^{\text{NP}|\|2}\). When sequent validity in \(\text{LOG}\) is in \(\text{PSPACE}\), deciding whether \(Q \prec_i P\) is a problem in \(\text{PSPACE}\).

Proof. First, we can evaluate \(P_1 \subseteq Q\), efficiently. We store the result in the Boolean variable \(v_\subseteq\).

We can then perform two non-adaptive queries to an oracle to solve sequent validity in \(\text{LOG}\) on \(\text{out}(P) \vdash \gamma_i\) and on \(\text{out}(Q) \vdash \gamma_i\), and store the results in the Boolean variables \(v_P\) and \(v_Q\) respectively. The formula \((\neg v_P \wedge \neg v_Q \wedge v_\subseteq) \lor (v_P \wedge \neg v_Q) \lor (v_P \wedge v_Q \wedge v_\subseteq)\) is true iff \(Q \prec_i P\).

This yields a correct algorithm for deciding \(Q \prec_i P\) in \(P^{\text{NP}|\|2}\) when \(\text{LOG}\) is in \(\text{NP}\), and in \(\text{PSPACE}\) when \(\text{LOG}\) is in \(\text{PSPACE}\). \(\square\)

To compare the complexity of dichotomous and parsimonious preferences, remember from Prop. 8 that when \(\text{LOG}\) is in \(\text{NP}\), the same problem for dichotomous preferences is in \(\text{BH}_2\). From [11] we know that \(P^{\text{NP}|\|1} \subseteq \text{BH}_2 \subseteq P^{\text{NP}|\|2}\). Is is not known whether these inclusions are strict.

5.1 Illustration of redistribution and parsimony

Consider again the individual resource game of Section 4.2. (Unless stated otherwise, suppose we are in the affine case.) With parsimonious preferences, we have \(NE(G) = \{(\emptyset, \emptyset)\}\). The profile \((\{A\}, \{B\})\) is not a Nash equilibrium as it was with dichotomous preferences. It would be more desirable from a social welfare point of view than any other outcome (it satisfies both players), but the players would nonetheless not be individually rational by choosing it. They have indeed no bearing upon the outcome that satisfies them and thus are rational in withholding their resources.

Nonetheless, like in the case of dichotomous preference, we can effectively eliminate the current Nash equilibrium in \(G'\) and construct the Nash equilibrium yielding \((\{A\}, \{B\})\) by redistributing the resources present in \(G'\) so as to obtain \(G' = (\{1, 2\}, \gamma_1 = B, \gamma_2 = A, \{B\}, \{A\})\). The only Nash equilibrium is now \((\{B\}, \{A\})\).

Unlike dichotomous preferences, parsimonious preferences do not ensure the existence of a Nash equilibrium in the affine case. Consider the individual resource game \(H' = (\{1, 2\}, \gamma_1 = A, \gamma_2 = A \oplus A, \{A\}, \{A\})\). There are two players. The game \(H'\) can be represented as on Figure 4.

The game \(H'\) has no Nash equilibrium: At \((\emptyset, \emptyset)\), Player 1 does not realize her objective, but she can deviate and play \(\{A\}\) to satisfy it. At \((\{1\}, \emptyset)\), Player 2 has an incentive to deviate and play \(\{A\}\) to realize her objective. At \((\{A\}, \emptyset)\) Player 1 has an incentive to deviate and play \(\emptyset\). (In the affine case this is because she can still satisfy her objective by contributing less. In the linear case, this is because she can satisfy her objective while she does not before deviating.) At \((\emptyset, \{A\})\), Player 2 does not satisfy her objective and thus has an incentive to deviate to play \(\emptyset\).

However, we can construct the Nash equilibrium yielding \((\{A\}, \{A\})\). Let \(\epsilon'\) be the redistribution of \(\epsilon\) such that \(\epsilon'(2) = \{A, A\}\) and \(\epsilon'(1) = \emptyset\). We obtain the game depicted on Figure 5.
In $H^\prime$, by assigning all the resources to Player 2, the profile $(\emptyset, \{A, A\})$ is a Nash equilibrium and the only one. In affine logics, both players satisfy their objectives, but only Player 2 does when the logic is linear.

### 5.2 Finding Nash equilibria

We study the complexity of NASH EQUILIBRIUM with parsimonious preferences.

#### 5.2.1 Hardness

We are not getting used to many-to-one reductions from sequent (in)validity. It was a fruitful problem in presence of dichotomous preference, and it will remain one in presence of parsimonious preferences. We prove a complexity lower bound for the problem of NE in presence of parsimonious preferences.

**Proposition 26.** The problem NE is as hard as the problem of checking sequent invalidity in LOG, even when there is only one player.

**Proof.** As before, we consider w.l.o.g. only the intuitionistic sequents of LOG in the following reduction.

Let $\Gamma \vdash \delta$ be an intuitionistic sequent of LOG. We define $\varphi = \Gamma^* \nrightarrow \delta$. We can construct the individual resource game $G$ such that $G = (\{1\}, \varphi, \{\varphi\})$. In $G$, Player 1 has exactly two choices: $\text{ch}_1(G) = (\emptyset, \{\varphi\})$.

We show that $\Gamma \vdash \delta$ iff $\varphi \not\in NE(G)$.

Suppose $(\varphi) \not\in NE(G)$. So $(\{\varphi\}) \prec_1 (\emptyset)$. Since by (ax) $\varphi \nrightarrow (\{\varphi\})$ satisfies Player 1’s objectives and $\emptyset \subset \{\varphi\}$ (Player 1’s contribution is strictly less in the profile $(\emptyset)$ than it is in $(\varphi)$), it must be that $\emptyset \nrightarrow \varphi$. We infer $\Gamma \vdash \varphi$, as we did in part of the proof of Prop. 18.

Suppose $\Gamma \vdash \delta$. We obtain $\Gamma^* \vdash \delta$ by using $\otimes L$ enough times, and we deduce $\nrightarrow \varphi$ with $\nrightarrow R$. We thus have $\emptyset \nrightarrow \varphi$ and $\emptyset \subset \{\varphi\}$. So $(\{\varphi\}) \prec_1 (\emptyset)$ and $(\varphi) \not\in NE(G)$.

#### 5.2.2 Algorithms

In the individual resource game $G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)$, we can use Algorithm 1 to check whether a profile $P \not\in NE(G)$, even for parsimonious preferences. We have a result analogous to Prop. 10 for parsimonious preferences.
Proposition 27. If the problem of sequent validity checking of LOG is in NP then NE is in $\Pi_2^p$. If the problem of sequent validity checking of LOG is in PSPACE then NE is in PSPACE.

Proof. We use Prop. 25 and the fact that $\text{coNP}^{\text{PSPACE}} \subseteq \text{coNP}^\Delta_p = \text{co}\Sigma_p^p = \Pi_2^p$. \hfill $\square$

When LOG is affine, we can do better than using Algorithm 1. We first state a technical lemma which is analogous to Lemma 11.

Lemma 28. Let $G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)$ be an individual resource game. When LOG is affine, $P \notin NE(G)$ iff there exists $i \in N$ such that $\text{out}(P) \not\prec_i \text{out}((P_{i}, C_i))$. There are three cases to consider:

1. out$(P) \not\prec_i \gamma_i$ and $P_i \neq \emptyset$;
2. out$(P) \not\prec_i \gamma_i$ and out$(\{P_{i}, \epsilon_i\}) \vdash \gamma_i$;
3. out$(P) \vdash \gamma_i$ and $\exists A \in P_i$ out$(\{P_{i}, \{A\}, C_i\}) \vdash \gamma_i$.

Proof. Right to left is immediate. From left to right, suppose $P \notin NE(G)$. So there exists $i \in N$ and $C_i \in \text{ch}_i(G)$ such that out$(P) \not\prec_i \text{out}((P_{i}, C_i))$. There are three cases to consider:

1. not out$(\{P_{i}, C_i\}) \vdash \gamma_i$ and not out$(P) \vdash \gamma_i$ and $C_i \subset P_i$;
2. out$(\{P_{i}, C_i\}) \vdash \gamma_i$ and not out$(P) \vdash \gamma_i$;
3. out$(\{P_{i}, C_i\}) \vdash \gamma_i$ and out$(P) \vdash \gamma_i$ and $C_i \subset P_i$.

Suppose (1) is the case. It implies that there is $C_i \subset P_i$ and thus that $P_i \neq \emptyset$. Suppose (2) is the case. We essentially use the same argument as the one used in the proof of Lemma 11. We have out$(\{P_{i}, C_i\}) \vdash \gamma_i$. By applying weakening $(|\epsilon_i| - |C_i|)$ times, we easily obtain that out$(\{P_{i}, \epsilon_i\}) \vdash \gamma_i$. Suppose (3) is the case. We thus have out$(\{P_{i}, C_i\}) \vdash \gamma_i$ with $C_i \subset P_i$. Take a formula $A \in P_i \setminus C_i$. Then, by applying weakening $(|P_i| - |C_i| - 1)$ times, we easily obtain that out$(\{P_{i}, P_i \setminus \{A\}\}) \vdash \gamma_i$. \hfill $\square$

Algorithm 6 can then be used to check whether $P \in NE(G)$.\footnote{Algorithm 6 corrects an omission in [26, Algo. 5] by adding “if ($P_i \neq \emptyset$): return false” lines 9 and 10.}

Algorithm 6 Algorithm for NE with parsimonious preferences and affine LOG

1: for each $i \in N$ do:
2: if out$(P) \not\prec_i \gamma_i$ : {
3: for each $A \in P_i$ do:
4: if out$(\{P_{i}, A\}) \vdash \gamma_i$:
5: return false.
6: else {  
7: if out$(\{P_{i}, \epsilon_i\}) \vdash \gamma_i$:
8: return false.
9: if ($P_i \neq \emptyset$):
10: return false.
11: }
12: return true.

Proposition 29. When LOG is affine, if the problem of sequent validity checking of LOG is in NP and we adopt parsimonious preferences, then NE is in $\text{P}^{\text{NP}}$. If the problem of sequent validity checking of LOG is in PSPACE, then NE is in PSPACE.

Proof. Lemma 28 justifies the correctness of Algorithm 6. The algorithm can be simulated by a deterministic oracle Turing machine in polynomial time with less than $\Sigma_{i \in N} (1 + |P_i|)$ non-adaptive queries to an oracle for sequent validity in LOG. When the complexity of sequent validity in LOG is in NP it yields a complexity of $\text{P}^{\text{NP}}$. \hfill $\square$
5.3 Elimination

We study the complexity of Rational Elimination with parsimonious preferences.

5.3.1 Algorithms

Lemma 15 also holds for parsimonious preferences. It is easy to see that the proof carries over.

Algorithm 3 can still be used in the case of parsimonious preferences because Lemma 15 is still granted. We thus have the analog to Prop. 16 for parsimonious preferences.

Proposition 30. When \( \text{LOG} \) is linear, \( \text{RE} \) is in \( \Sigma^p_2 \) when \( \text{LOG} \) is in \( \text{NP} \), and in \( \text{PSPACE} \) when \( \text{LOG} \) is in \( \text{PSPACE} \).

Proof. We use Prop. 25 and the fact that \( \text{NP} \parallel \text{P} \subseteq \text{NP} \parallel \text{NP} = \text{NP} \Delta^p_2 = \Sigma^p_2 \).

Let \( G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n) \) be an individual resource game and let \( P \in \text{ch}(G) \) be a profile. We can use Algorithm 7 to check whether a profile \( P \in \text{ch}(G) \) is rationally eliminable.

**Algorithm 7** Algorithm for \( \text{RE} \) with parsimonious preferences and affine \( \text{LOG} \)

1: for each \( i \in N \) do:
2: if \( ([\epsilon \triangleright \hat{i}(i)]_i) \prec_i ([\epsilon \triangleright \hat{i}(i)])_i)\):
3: return true.
4: for each \( A \in \text{out}(P)\):
5: if \( ([\epsilon \triangleright \hat{i}(i)]_i) \prec_i (\text{out}(P) \setminus \{A\})_i)\):
6: return true.
7: return false.

Proposition 31. When \( \text{LOG} \) is affine, \( \text{RE} \) is in \( \text{P} \parallel \text{NP} \) when \( \text{LOG} \) is in \( \text{NP} \). It is in \( \text{PSPACE} \) when \( \text{LOG} \) is in \( \text{PSPACE} \).

Proof. Lemma 15 which still holds with parsimonious preferences ensures that it is enough to consider the redistributions \( [\epsilon \triangleright \hat{i}]_i \) for some player \( i \). Algorithm 7, then checks for each of these redistributions whether Player \( i \) has an incentive to deviate in the game \( G[\epsilon \triangleright \hat{i}]_i \) from the profile \( (\text{out}(P))_i \in \text{ch}(G[\epsilon \triangleright \hat{i}]_i) \) to any one of \( (\text{out}(P))_i \in \text{ch}(G[\epsilon \triangleright \hat{i}]_i) \) and \( (\text{out}(P) \setminus \{A\})_i \in \text{ch}(G[\epsilon \triangleright \hat{i}]_i) \) for some \( A \in \text{out}(P) \). It is weakening \( (W) \) that justifies that it is enough to consider these profiles, because \( X \not\models \gamma_i \) implies \( Y \not\models \gamma_i \) for any couple of multisets \( Y \subseteq X \). The correctness of Algorithm 7 follows.

The tests of line 2 and line 5 only involve the following instances of the sequent validity decision problem: \( ([\epsilon \triangleright \hat{i}]_i) \vdash \gamma_i \) and \( ([\epsilon \triangleright \hat{i}]_i) \vdash \gamma_i \) for every Player \( i \in N \), and \( (\text{out}(P) \setminus \{A\}) \vdash \gamma_i \), for every Player \( i \in N \) and every formula \( A \in \text{out}(P) \). The algorithm can thus be simulated by a deterministic oracle Turing machine in polynomial time with at most \( |N|(|\text{out}(P)| + 2) \) non-adaptive calls to an oracle for sequent validity.

5.3.2 Hardness

After Prop. 24 and the proof of Prop. 18, the following proposition does not come as a surprise.

Proposition 32. \( \text{RE} \) is as hard as the problem of checking sequent invalidity in \( \text{LOG} \).

Proof. Let \( \Gamma \vdash \delta \) be an arbitrary intuitionistic sequent. We construct the same game as in the proof of Prop. 18. Let \( \varphi = \Gamma^* \rightarrow \delta \). Let \( G^\varphi = ([1, 2], \varphi, 1, 0, \{\varphi\}) \).
In the proof of Prop. 18, we showed that, in presence of dichotomous preferences, both in the case of linear and of affine logics, we have \( \Gamma \not\vdash \delta \) iff \((\emptyset, \emptyset)\) is rationally eliminable in \( G' \).

Now with Prop. 24, we know that \((\emptyset, \emptyset)\) is a Nash equilibrium in presence of dichotomous preferences iff it is a Nash equilibrium in presence of parsimonious preferences (both in \( G' \) and \( G'' \), and no matter if \( \text{LOG} \) is linear or affine, or if \( \Gamma \vdash \delta \) or \( \Gamma \not\vdash \delta \)).

Hence, we have \( \Gamma \not\vdash \delta \) iff \((\emptyset, \emptyset)\) is rationally eliminable in \( G' \), also in presence of parsimonious preferences.

5.4 Construction

Finally, we tackle the complexity of RATIONAL CONSTRUCTION with parsimonious preferences.

5.4.1 Hardness

We establish a complexity lower bound for the problem of \( \text{RC} \) in presence of parsimonious preferences.

**Proposition 33.** \( \text{RC} \) is as hard as the problem of checking sequent invalidity in \( \text{LOG} \).

**Proof.** Consider the games in the proof of Prop. 32. We can see that both for linear and affine logics we have that \( \Gamma \not\vdash \delta \) iff \((\{\varphi\}, \emptyset)\) can be rationally constructed in \( G'' \).

5.4.2 Algorithms

Our algorithmic analysis is very similar to the analysis we made when the preferences are dichotomous in Section 4.3.2. Let \( G' \) be an individual resource game and \( P \in \text{ch}(G') \). To decide whether \( P \) can be rationally constructed we can reuse Algorithm 5.

Again, we use the problem \( \text{NE} \) as a blackbox, for which complexity upper bounds have been established in Prop. 27 and Prop. 29.

**Proposition 34.** When \( \text{LOG} \) is in \( \text{NP} \), \( \text{RC} \) is in \( \Sigma_3^p \). When \( \text{LOG} \) is in \( \text{PSPACE} \), \( \text{RC} \) is in \( \text{PSPACE} \).

**Proof.** The proof is similar to the one of Prop. 20, using the result of Prop. 27.

The next proposition also comes without surprise.

**Proposition 35.** If \( \text{LOG} \) is affine, when \( \text{LOG} \) is in \( \text{NP} \), \( \text{RC} \) is in \( \Sigma_2^p \). When \( \text{LOG} \) is in \( \text{PSPACE} \), \( \text{RC} \) is in \( \text{PSPACE} \).

**Proof.** The proof is similar to the one of Prop. 21, using the result of Prop. 29.

6 Examples

We present more thorough examples. They involve several resources and objectives that are modelled with a variety of logical operands. We use the opportunity to present fully the important formal proofs of the realized objectives.
6.1 Alan and the fish

We first introduce the resources involved and how they are built in the logical language.

- Basic resources:
  - one mole of dioxygen: \(O_2\)
  - one mole of dihydrogen: \(H_2\)
  - one mole of water: \(H_2O\)
  - one ‘token’ of thirst: \(T\)

- Anti-resources can be captured via the linear negation:
  - one thirst quencher: \(\sim T\)

- Resource transformation processes:
  - one process of electrolysis: \(E = H_2O \otimes H_2O \rightarrow H_2 \otimes H_2 \otimes O_2\)
  - one process of drinking water: \(D = H_2O \rightarrow \sim T\)

**Game definition.** Let \(G'_{af} = (\{a, f\}, \gamma_a, \gamma_f, \epsilon_a, \epsilon_f)\) be the individual resource game with two players, Alan \(a\) and the Fish \(f\). The fish wants one mole of dioxygen: \(\gamma_f = O_2\). Alan wants one mole of dioxygen for his fish and wants to quench his thirst: \(\gamma_a = O_2 \otimes \sim T\).

In the game \(G'_{af}\), Alan is endowed with \(\epsilon_a = \{D, E\}\). He can drink once and can electrolysis water once. The fish is endowed with three tokens of water \(\epsilon_f = \{H_2O, H_2O, H_2O\}\).

We suppose that LOG is affine. For this example, we will consider both cases of dichotomous and parsimonious preferences.

As we did before, we will represent a Nash equilibrium under dichotomous preferences with the symbol □, and under parsimonious preferences with the symbol ■. By Prop 23, the latter implies the former. Then, when a profile is a Nash equilibrium under both dichotomous and parsimonious preferences we will use the symbol ■. The game \(G'_{af}\) and the realized objectives can be depicted as on Figure 6.

| \(a\) | \(f\) | \(\emptyset\) | \(\{H_2O\}\) | \(\{H_2O, H_2O\}\) | \(\{H_2O, H_2O, H_2O\}\) |
|------|------|-------------|-------------|-----------------|-----------------|
| \(\emptyset\) | \(\{D\}\) | □ | \(\{H_2O\}\) | \(\{H_2O, H_2O\}\) | \(\{H_2O, H_2O, H_2O\}\) |
| \(\{D\}\) | \(\{D\}\) | □ | \(\{H_2O, H_2O\}\) | \(\{H_2O, H_2O, H_2O\}\) | \(\{H_2O, H_2O, H_2O\}\) |
| \(\{E\}\) | \(\{E\}\) | □ | \(\{H_2O, H_2O\}\) | \(\{H_2O, H_2O, H_2O\}\) | \(\{H_2O, H_2O, H_2O\}\) |
| \(\{D, E\}\) | \(\{D, E\}\) | □ | \(\{H_2O, H_2O\}\) | \(\{H_2O, H_2O, H_2O\}\) | \(\{H_2O, H_2O, H_2O\}\) |

Figure 6: The game \(G'_{af}\). LOG is affine. The symbol □ marks the Nash equilibria under dichotomous. The symbol ■ marks the profiles that are also Nash equilibria under both dichotomous and parsimonious preferences.

We show next how the objectives are realized.
**Formal proofs of the realized objectives.** The proof of $H_2O, H_2O, E \vdash \gamma_f$ will be instrumental for the others. We label it Proof $\star$ for reuse.

\[
\begin{array}{c}
\text{Proof }\star \quad \frac{O_2 \vdash O_2}{W} \\
\text{Proof }\star \quad \frac{O_2 \otimes H_2 \otimes O_2 \vdash O_2}{E} \\
\text{Proof }\star \quad \frac{O_2 \otimes H_2 \otimes O_2 \vdash O_2}{\otimes L}
\end{array}
\]

The other realized objectives of the fish are immediate using Proof $\star$ and the weakening rule. We prove that $H_2O, H_2O, H_2O, E \vdash \gamma_f$, $D, H_2O, H_2O, E \vdash \gamma_f$, and $D, H_2O, H_2O, H_2O, E \vdash \gamma_f$.

\[
\begin{array}{c}
\text{Proof }\star \quad \frac{H_2O, H_2O, E, H_2O \vdash \gamma_f}{H_2O, H_2O, E, H_2O \vdash \gamma_f} \\
\text{Proof }\star \quad \frac{H_2O, H_2O, E, D \vdash \gamma_f}{D, H_2O, H_2O, E, D \vdash \gamma_f} \\
\text{Proof }\star \quad \frac{H_2O, H_2O, E, H_2O \vdash \gamma_f}{D, H_2O, H_2O, E, H_2O \vdash \gamma_f}
\end{array}
\]

Finally, we prove $D, E, H_2O, H_2O, H_2O \vdash \gamma_a$. The proof also uses Proof $\star$.

\[
\begin{array}{c}
\text{Proof }\star \quad \frac{H_2O, H_2O, E \vdash \gamma_f}{H_2O, H_2O, E \vdash \gamma_f} \\
\text{Definition} \quad \frac{H_2O \rightarrow \sim T, H_2O \vdash \sim T}{\otimes R}
\end{array}
\]

**Dichotomous preferences: eliminations of bad equilibria.** If the preferences are dichotomous, there are plenty Nash equilibria in $G'_{af}$. They are: $(\emptyset, \emptyset)$, $(\emptyset, \{H_2O\})$, $(\emptyset, \{H_2O, H_2O\})$, $(\{D\}, \emptyset)$, $(\{D\}, \{H_2O, H_2O\})$, $(\{E\}, \{H_2O, H_2O\})$, $(\{D, E\}, \{H_2O, H_2O\})$, and $(\{D, E\}, \{H_2O, H_2O, H_2O\})$.

However, only the profile $(\{D, E\}, \{H_2O, H_2O, H_2O\})$, whose outcome is $\{D, H_2O, H_2O, H_2O, E\}$, satisfies the objectives of both players. It would thus be desirable to eliminate the other profiles. To do so, let $\epsilon'$ be the endowment such that $\epsilon'_a = \{D, E, H_2O, H_2O, H_2O\}$ and $\epsilon'_f = \emptyset$. The game $G'_{af}$ and the realized objectives can be (partially) depicted as on Figure 7.

It is readily seen that in $G'_{af}$, when preferences are dichotomous, only the profile $(\{D, E, H_2O, H_2O, H_2O\}, \emptyset)$ whose outcome is $\{D, H_2O, H_2O, H_2O, E\}$, is a Nash equilibrium.

**Parsimonious preferences: construction of a good equilibrium.** If the preferences are parsimonious, the profile $(\emptyset, \emptyset)$ is a Nash equilibrium, and is the only one. One can nonetheless redistribute the resources so as to construct an equilibrium where Alan and the fish both realize their objectives. That is, one can construct the profile $(\{D, E\}, \{H_2O, H_2O, H_2O\})$. To do so, let $\epsilon''$ be the endowment such that $\epsilon''_a = \{D, H_2O, H_2O, H_2O\}$ and $\epsilon''_f = \{E\}$. The game $G''_{af}$ and the realized objectives can be depicted as on Figure 8.
When preferences are parsimonious, the profiles \((\emptyset, \emptyset)\) and \(\{(D, H_2O, H_2O, H_2O), \{E\}\}\) are Nash equilibria in \(G'_{af}\) and are the only ones.

Notice that, the redistribution \(\epsilon'\) would also effectively construct the profile \(\{(D, H_2O, H_2O, H_2O), \{E\}\}\). At the price of a more draconian redistribution, it would also eliminate \((\emptyset, \emptyset)\).

6.2 Ann and Bernard get a divorce

We formalize Example 1. We will only consider parsimonious preferences. We also assume that \(\text{LOG}\) is Affine MLL. We introduce the resources involved in the example.

- the alarm clock: \(\text{aclock}\)
- the resource of flour for a year: \(\text{flour}\)
- the resource of one year worth of bread: \(\text{bread}\)
- the breadmaker is the resource transformation process: \(\text{flour} \rightarrow \text{bread}\)

Using these as basic resources, we formalize Example 1 as the game \(G_{ab}\).

**Game definition.** Let \(G_{ab}^\epsilon = (\{a,b\}, \gamma_a, \gamma_b, \epsilon_a, \epsilon_b)\) be the individual resource game with two players, Ann \(a\) and Bernard \(b\). Ann wants enough bread for a year: \(\gamma_a = \text{bread}\). Bernard wants the alarm clock: \(\gamma_b = \text{aclock}\). In the game \(G_{ab}^\epsilon\), Ann is endowed with the alarm clock: \(\epsilon_a = \{\text{aclock}\}\). Bernard is endowed with enough flour to make bread for two years, and with the breadmaker: \(\epsilon_b = \{\text{flour}, \text{flour}, \text{breadmaker}\}\).

The game \(G_{ab}^\epsilon\) and the realized objectives can be depicted as on Figure 9. (For the convenience representation, Bernard plays rows, and Ann plays columns.)

**Formal proofs of the realized objectives.** All the formal proofs of the realized objectives are trivial. The proof that \(\text{aclock} \vdash \gamma_b\) indicating that Bernard can realize his objective with the only resource of an alarm clock is simply:

\[
\text{aclock} \vdash \text{aclock} \quad \text{ax}
\]
The proof that Ann can realize her objective with resources of enough flour for a year and a breadmaker, viz., that flour, flour → bread ⊨ γₐ, is also very simple.

\[
\text{flour} \vdash \text{flour} \quad \text{ax} \quad \text{flour, flour} \vdash \text{bread} \quad \text{ax} \quad \text{flour, flour} \vdash \text{bread} \quad \text{ax}
\]

These two elementary proofs can easily be extended to a proof for every other realized objective by using the weakening rule (W).

An undesirable equilibrium. One can see on Figure 9, that the profiles (\{flour, flour → bread\}, \{aclock\}) and (\{flour, flour, flour → bread\}, \{aclock\}) in chₐ × chₐ would satisfy both Ann and Bernard. However, in both them, Bernard has an incentive to provide less resources from his endowment, and to deviate to \(\emptyset \in chₐ\). In turn, in (\(\emptyset, \{aclock\}\)) in chₐ × chₐ, Ann is not satisfied, and so has an incentive to retain her resources as well, deviating to her choice \(\emptyset \in chₐ\). The profiles (\{flour, flour → bread\}, \(\emptyset\)) and (\{flour, flour, flour → bread\}, \(\emptyset\}) in chₐ × chₐ satisfy Ann’s objective but do not satisfy Bernard’s. Hence, Bernard has an incentive to deviate to \(\emptyset \in chₐ\).
The profile \((\emptyset, \emptyset)\) is the only Nash equilibrium of \(G'_{ab}\), but it satisfy neither Ann’s objective, nor Bernard’s. On the other hand, the outcome of the profile \((\{\text{flour}, \text{flour} \rightarrow \text{bread}\}, \{\text{aclock}\}) \in ch_b \times ch_a\) would satisfy them both.

A desirable redistribution. So the arbitrator redistributes the resources that are available. He assigns the breadmaker and half the flour to Ann. He assigns the alarm clock and half the flour to Bernard. That is, \(\epsilon'_a = \{\text{flour}, \text{flour} \rightarrow \text{bread}\}\) and \(\epsilon'_b = \{\text{flour}, \text{aclock}\}\). This redistribution yields the game \(G'_{ab}\). It can be depicted as on Figure 10. (As precedently, Bernard plays rows, and Ann plays columns.) In \(G'_{ab}\), the profile \((\emptyset, \emptyset)\) is not a Nash equilibrium, and so has been eliminated. Indeed, it does not satisfy Bernard, and he has an incentive to deviate to the profile \((\{\text{aclock}\}, \emptyset) \in ch_b \times ch_a\) in which his objective is satisfied. But \((\{\text{aclock}\}, \emptyset)\) is not a Nash equilibrium either. Indeed, it does not satisfy Ann, and she has an incentive to deviate to the profile \((\{\text{aclock}\}, \{\text{flour, flour} \rightarrow \text{bread}\}) \in ch_b \times ch_a\). From here, nobody has an incentive to deviate, and it is a Nash equilibrium. It is in fact the only Nash equilibrium in \(G'_{ab}\).

One can readily see that the profile \((\{\text{flour, aclock}\}, \{\text{flour, flour} \rightarrow \text{bread}\}) \in ch_b \times ch_a\), even though it satisfies both Ann and Bernard, is not a Nash equilibrium. Bernard has an incentive to provide less resources. The same can be said about the profile \((\{\text{flour, aclock}\}, \{\text{flour} \rightarrow \text{bread}\}) \in ch_b \times ch_a\).

### 7 Cooperative games with individual goals

We have used the individual resource games as models of strategic individual games. The players have to seek after resource objectives and compete for those resources. When unable to reach a resource alone, they might have to form coalitions. (See, e.g., [17].) In this section, we show how IRGs can also be used as models of cooperative games.

In their abstract definition, coalitional games are presented as a tuple \((N, v)\), where \(N\) is a set of agents, and \(v : 2^N \rightarrow \mathbb{R}\) is a coalition collective payoff, or valuation function. Typically, we assume that \(v(\emptyset) = 0\). We call simple game a coalitional game such that for every coalition \(C \subseteq N\), we have \(v(C) = 0\) or \(v(C) = 1\). Where these utilities come from however is not part of the description. Here we build our models of coalitional games on top of the individual resource games. Each player \(i\) of a game is endowed with a multiset of resources \(\epsilon_i\). An action for Player \(i\) consists in contributing a subset of \(\epsilon_i\). Then, each player \(i\) has a goal \(\gamma_i\), which is a resource, represented by one formula of LOG. In the resulting coalition games, the valuation function will depend of these individual endowments and goals. Analogous models can be found in [31] and [3].

A coalition \(C\) is a subset of \(N\). We denote the goal of coalition \(C\) with the multiset \(\gamma_C = \{\gamma_i \mid i \in C\}\). We denote the endowment of coalition \(C\) with \(\epsilon_C = \bigcup_{i \in C} \epsilon_i\).
We say a coalition of agents \( C = \{c_1, \ldots, c_p\} \), with \( c_i \neq c_j \) for every \( i \neq j \), can perform a multiset of resources \( P \subseteq \text{LOG} \) if \(^7\)
\[
\exists E_1 \subseteq \epsilon_{c_1}, \ldots, \exists E_p \subseteq \epsilon_{c_p}
\]
such that\(^8\)
\[
E_1, \ldots, E_p \vdash \bigotimes P
\]
When it is the case, we write canperform\((C, P)\). We then denote the set of multisets of individual goals a coalition \( C \) can perform
\[
\Gamma(C) = \{\Gamma \subseteq \gamma_C \mid \text{canperform}(C, \Gamma)\}
\]
We state a few properties of the predicate canperform.

**Proposition 36.** The following properties hold:

1. If \( C_1 \subseteq C_2 \) and canperform\((C_1, P)\) then canperform\((C_2, P)\).
2. If \( C_1 \cap C_2 = \emptyset \), canperform\((C_1, P_1)\), and canperform\((C_2, P_2)\) then canperform\((C_1 \cup C_2, P_1 \uplus P_2)\).
3. If \( \text{LOG} \) is affine, \( C_1 \cap C_2 = \emptyset \), canperform\((C_1, P_1)\), and canperform\((C_2, P_2)\) then canperform\((C_1 \cup C_2, P_1 \cup P_2)\).

**Proof.** We prove each item:

**Proof of (1).** Let \( C_1 = \{c_1, \ldots, c_p\} \) and \( C_2 = C_1 \cup \{c_{p+1}, \ldots, c_q\} \). Suppose canperform\((C_1, P)\).
So, \( \exists E_1 \subseteq \epsilon_{c_1}, \ldots, \exists E_p \subseteq \epsilon_{c_p} \) such that \( E_1, \ldots, E_p \vdash \bigotimes P \). Also, \( E_1, \ldots, E_p, \emptyset, \ldots, \emptyset \vdash \bigotimes P \). So, \( \exists E_1 \subseteq \epsilon_{c_1}, \ldots, \exists E_p \subseteq \epsilon_{c_p} \) such that \( E_1, \ldots, E_p \vdash \bigotimes P \), and we conclude.

**Proof of (2).** Let \( C_1 = \{c_1, \ldots, c_p\} \) and \( C_2 = \{c_{p+1}, \ldots, c_q\} \). The fact canperform\((C_1, P_1)\) implies \( \exists E_1 \subseteq \epsilon_{c_1}, \ldots, \exists E_p \subseteq \epsilon_{c_p} \) such that \( E_1, \ldots, E_p \vdash \bigotimes P_1 \). The fact canperform\((C_2, P_2)\) implies \( \exists E_{p+1} \subseteq \epsilon_{c_{p+1}}, \ldots, \exists E_q \subseteq \epsilon_{c_q} \) such that \( E_{p+1}, \ldots, E_q \vdash \bigotimes P_2 \).
So, \( \exists E_1 \subseteq \epsilon_{c_1}, \ldots, \exists E_q \subseteq \epsilon_{c_q} \) such that \( E_1, \ldots, E_p \vdash \bigotimes P_1 \) and \( E_{p+1}, \ldots, E_q \vdash \bigotimes P_2 \). With, \( \ominus \text{R} \), we thus obtain \( E_1, \ldots, E_p, E_{p+1}, \ldots, E_q \vdash \bigotimes P_1 \otimes \bigotimes P_2 \), which is equivalent to \( E_1, \ldots, E_q \vdash \bigotimes (P_1 \uplus P_2) \).

Now suppose that \( C_1 \cap C_2 = \emptyset \). So all players from \( c_1 \) through \( c_q \) are unique, and we can conclude.

**Proof of (3).** Starting with the same premises as the ones for the proof of (2), we obtain that \( \exists E_1 \subseteq \epsilon_{c_1}, \ldots, \exists E_q \subseteq \epsilon_{c_q} \) such that \( E_1, \ldots, E_q \vdash \bigotimes (P_1 \uplus P_2) \). Since \( \text{LOG} \) is affine, we can show that \( E_1, \ldots, E_q \vdash \bigotimes (P_1 \uplus P_2) \). It immediately follows from two simple facts: (i) \( P_1 \uplus P_2 \subseteq P_1 \uplus P_2 \), and (ii) for every three multisets of formulas \( X, Y, \Gamma \): if \( X \subseteq Y \), \( \Gamma \vdash \bigotimes Y \), and \( \text{LOG} \) is affine, then \( \Gamma \vdash \bigotimes X \).
(i) is trivial. (ii) is rather banal to a reader familiar with formal proofs: take \( X = \{A_1, \ldots, A_{k_Y}\} \) and \( Y = \{A_1, \ldots, A_{k_Y}\} \) with \( k_Y > k_X \). We start from the axiom (rule (ax)) \( A_1 \otimes \ldots \otimes A_{k_Y} \vdash A_1 \otimes \ldots \otimes A_{k_Y} \) and, with every formulas \( A_i, k_X < i \leq k_Y \), successively apply (W) to add \( A_i \) to the left part of the sequent, and apply (L). We obtain \( A_1 \otimes \ldots \otimes A_{k_X} \otimes \ldots \otimes A_{k_Y} \vdash A_1 \otimes \ldots \otimes A_{k_X} \). That is \( \bigotimes X \vdash \bigotimes X \). Together with \( \Gamma \vdash \bigotimes Y \), we conclude that \( \Gamma \vdash \bigotimes Y \) by applying (cut).

**Remark 37.** Notice that in general, if \( \text{LOG} \) is linear, Prop. 36.3 does not hold. To see this, consider the \( \text{IRG} \) \( \{(1, 2), \gamma_1 = D, \gamma_2 = D, \epsilon_1 = \{A \otimes B\}, \epsilon_2 = \{A \otimes C\}\} \), and assume that \( A \) is not a vacuous resource (that is, not provably equivalent to the constant 1). We have canperform\({\{1\}, \{A, B\}\}) and canperform\({\{2\}, \{A, C\}\}), but we don’t have canperform\({\{1, 2\}, \{A, B, C\}\}). Indeed, to obtain the resources \( B \) and \( C \), both agents need to play their full endowment. But then, we will be left with an extra \( A \) that we cannot dispose of.

We first give a general definition of our coalitional games where the valuation function of the coalitions is kept abstract.

---

\(^7\)A formula of \( \text{LOG} \) can occur more than once in \( P \); the multiplicity of a formula in \( \epsilon_i \) cannot exceed its multiplicity in \( E_i \).

\(^8\)\( \bigotimes \{\epsilon_1, \ldots, \epsilon_n\} = \epsilon_1 \otimes \ldots \otimes \epsilon_n \), with the convention \( \bigotimes \emptyset = 1 \)—the neutral element for \( \otimes \).
Definition 38. An Individual Goal Coalition Resource Game (IGCRG) is a tuple \((G, v)\) where:

- \(G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)\) is an individual resource game;
- \(v: 2^N \rightarrow \mathbb{R}\) is a coalition valuation function.

The function \(v\) assigns a utility to every coalition. In the remainder, we exemplify IGCRGs with two instantiations of \(v\). In Section 7.1, we present a model of simple games where a coalition has utility 1 when its members can act together and achieve all their individual goals. In Section 7.2, we present a model where the value of a coalition is the maximal number of its members’ goals it can satisfy by acting together.

7.1 All Individual Goal Coalition Resource Games

We define a first concrete instantiation of IGCRG. The following class of models is equivalent to one-goal Rich Coalitional Resource Games introduced in [27].

Definition 39. An All Individual Goal Coalition Resource Game (AIGCRG) is a tuple \((G, v)\) where:

- \(G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)\) is an individual resource game;
- For every coalition \(C \subseteq N \setminus \{\emptyset\}\), \(v(C) = 1\) if canperform\((C, \gamma_C)\) and \(v(C) = 0\) otherwise. Also, \(v(\emptyset) = 0\).

An AIGCRG is thus a simple game. A coalition \(C\) is winning when it can perform \(\gamma_C\), that is, all the individual goals of its members simultaneously. The coalition is losing otherwise.

We illustrate AIGCRG with a very simple example.

Example 40. Let \((G, v)\) be the AIGCRG where \(G = (\{1, 2, 3\}, \gamma_1 = A, \gamma_2 = A \otimes A, \gamma_3 = A, \epsilon_1 = \{A\}, \epsilon_2 = \{A\}, \epsilon_3 = \{A, A\})\). We have:

| \(C \subseteq \{1, 2, 3\}\) | \(v(C)\) |
|--------------------------|--------|
| \(\emptyset\)            | 0      |
| \(\{1\}\)               | 1      |
| \(\{2\}\)               | 0      |
| \(\{3\}\)               | 1      |
| \(\{1, 2\}\)            | 0      |
| \(\{1, 3\}\)            | 1      |
| \(\{2, 3\}\)            | 1      |
| \(\{1, 2, 3\}\)         | 1      |

It is indeed a simple example, and it admits simple proofs involving only the rules (ax) and \((\otimes R)\). We see that \(v(\{2, 3\}) = 1\). So \(\{2, 3\}\) is a winning coalition. To see it, take \(E_2 = \epsilon_1 = \{A\}\) and \(E_3 = \epsilon_3 = \{A, A\}\). We can prove that \(E_2, E_3 \vdash \gamma_2 \otimes \gamma_3\), as follows:

\[
\frac{A \vdash A}{A, A \vdash A \otimes A}^{\text{ax}} \quad \frac{A \vdash A}{A, A \vdash (A \otimes A) \otimes A}^{\text{ax}} \quad \frac{A \vdash A}{A \vdash A}^{\text{ax}}
\]

Remark 41. AIGCRGs are in general neither monotonic nor superadditive. The former is rather unusual. The latter is particularly expected for a class of simple games. In general, AIGCRGs are not monotonic. In Example 40 we can see that \(v(\{1\}) = 1\), but \(v(\{1, 2\}) = 0\). In general, AIGCRGs are not superadditive. In Example 40 we can see that \(v(\{1\}) = 1\) and \(v(\{3\}) = 1\), but \(v(\{1, 3\}) = 1 < v(\{1\}) + v(\{3\})\).
As mentioned, AIGCRGs are equivalent to one-goal Rich Coalitional Resource Games (one-goal RCRGs) studied in [27]; the following example is adapted from there.

**Example 42.** Player 1 is happy with bacon, Player 2 is happy with either bacon or an egg, and Player 3 is happy with an omelet. Player 1 is endowed with one egg and the capacity of using an egg to make an omelet. Player 2 is endowed with bacon. Player 3 is endowed with one egg.

Formally, b stands for bacon, e for one egg, and o for an omelet. Player 1 is happy with b, Player 2 is happy with either b or e and does care about choosing (i.e., b $\oplus$ e), and Player 3 is happy with o. Player 1 is endowed with one token of e and the consumable capacity of transforming an e into an o (i.e., e $\rightarrow$ o).

Player 2 is endowed with one token of b. Player 3 is endowed with one token of e. To formalise it, let $(G, v)$ be the the AIGCRG where $G = (N, \gamma_1, \gamma_2, \gamma_3, \epsilon_1, \epsilon_2, \epsilon_3)$, where:

- $N = \{1, 2, 3\}$
- $\gamma_1 = b$
- $\epsilon_1 = \{e, e \rightarrow o\}$
- $\gamma_2 = b \oplus e$
- $\epsilon_2 = \{b\}$
- $\gamma_3 = o$
- $\epsilon_3 = \{e\}$

Clearly, Player 3 needs Player 1 to be happy. Player 1 needs Player 2 to be happy. Player 2 can rely on herself by using her endowed b. However, if she forms a winning coalition with Player 1, this b must be used towards the happiness of Player 1. In this case, it is Player 1’s endowed e that will be used towards Player 2’s happiness. To add Player 3 into the winning coalition, Player 3 can provide his endowed e, which can be transformed into an o using Player 1’s capacity e $\rightarrow$ o.

The winning coalitions are $\{2\}$, $\{1, 2\}$, and $\{1, 2, 3\}$. The coalition $\{2\}$ is winning because $b \vdash b \oplus e$ and $\{b\} \subseteq \epsilon_2$. The coalition $\{1, 2\}$ is winning because $e, b \vdash b \otimes (b \oplus e)$, $\{e\} \subseteq \epsilon_1$, and $\{b\} \subseteq \epsilon_2$.

We show in more details that $\{1, 2, 3\}$ is a winning coalition, and that they can win by using all their endowed resources.

\[
\epsilon_1, \epsilon_2, \epsilon_3 \vdash b \otimes (b \oplus e) \otimes o \\
\]

A player is a veto player when there is no winning coalition without the player’s contribution. Since we have identified all the winning coalitions in $(G, v)$, we can easily determine the veto players. Player 2 is the only veto player of the game. Player 1 and Player 3 are not, as witnessed by $\{2\}$ being a winning coalition.

A player is a dummy player when its presence or absence in a coalition does not change the value; it has neither a positive nor a negative impact. Player 3 is the only dummy player of the game $(G, v)$. Player 1 is not a dummy because $v(\{1, 2, 3\}) = 1$ and $v(\{2, 3\}) = 0$. Player 2 is not a dummy because $v(\emptyset) = 0$ and $v(\{2\}) = 1$.

A payoff vector is a (real-valued) distribution of the value of the grand coalition, that is $v(N) = v(\{1, 2, 3\}) = 1$, here. A payoff vector is stable, in the core, when no coalition could get more that it gets. Let $p = (0, 1, 0)$ be a payoff vector. Since Player 2 is a veto player it is in the core of the game. It is the only one, because Player 2 is the only veto player.

### 7.2 Max Number Individual Goal Coalition Resource Games

Definition 38 leaves space to more and richer specializations, which enjoy some properties that AIGCRGs do not.
Definition 43. A Max Number Individual Goal Coalition Resource Game (MNIGCRG) is a tuple \((G, v)\) where:

- \(G = (N, \gamma_1, \ldots, \gamma_n, \epsilon_1, \ldots, \epsilon_n)\) is an individual resource game;
- For every coalition \(C \subseteq N\), \(v(C) = \max_{\Gamma \in \Gamma(C)} |\Gamma|\).

We illustrate MNIGCRG with a very simple example.

Example 44. Let \((G, v)\) be the MNIGCRG where \(G = (\{1, 2, 3\}, \gamma_1 = A, \gamma_2 = A \otimes A, \gamma_3 = A, \epsilon_1 = \{A\}, \epsilon_2 = \{A\}, \epsilon_3 = \{A, A\})\). We have:

| \(C \subseteq \{1, 2, 3\}\) | \(v(C)\) |
|-----------------|-----|
| \(\emptyset\)   | 0   |
| \(\{1\}\)      | 1   |
| \(\{2\}\)      | 0   |
| \(\{3\}\)      | 1   |
| \(\{1, 2\}\)   | 1   |
| \(\{1, 3\}\)   | 2   |
| \(\{2, 3\}\)   | 2   |
| \(\{1, 2, 3\}\)| 3   |

MNIGCRGs are monotonic games.

Proposition 45. If \(C_1 \subseteq C_2\) then \(v(C_1) \leq v(C_2)\).

Proof. Suppose \(C_1 \subseteq C_2\). By Prop. 36.1, we have that whenever \(\text{canperform}(C_1, \Gamma)\), we also have \(\text{canperform}(C_2, \Gamma)\). This means that \(\Gamma(C_1) \subseteq \Gamma(C_2)\). Thus \(\max_{\Gamma \in \Gamma(C_1)} |\Gamma| \leq \max_{\Gamma \in \Gamma(C_2)} |\Gamma|\), and we can conclude. \(\square\)

MNIGCRGs are superadditive games.

Proposition 46. If \(C_1 \cap C_2\) then \(v(C_1) + v(C_2) \leq v(C_1 \cup C_2)\).

Proof. By definition of MNIGCRG, there is \(\Gamma_1 \subseteq \gamma C_1\) such that \(\text{canperform}(C_1, \Gamma_1)\) and \(|\Gamma_1| = v(C_1)\). Similarly, there is \(\Gamma_2 \subseteq \gamma C_2\) such that \(\text{canperform}(C_2, \Gamma_2)\) and \(|\Gamma_2| = v(C_2)\). Now suppose \(C_1 \cap C_2\). By Prop. 36.2, we know that \(\text{canperform}(C_1 \cup C_2, \Gamma_1 \uplus \Gamma_2)\). Clearly \(|\Gamma_1 \uplus \Gamma_2| = |\Gamma_1| + |\Gamma_2|\), and, by definition of MNIGCRG, \(v(C_1 \cup C_2) \geq |\Gamma_1 \uplus \Gamma_2|\). Thus, we have \(v(C_1 \cup C_2) \geq v(C_1) + v(C_2)\). \(\square\)

Still, there are properties that MNIGCRGs do not enjoy.

Remark 47. MNIGCRGs are not subadditive (and thus not additive, and not submodular): In Example 44, \(v(\{1, 3\}) + v(\{2\}) < v(\{1, 2, 3\})\). They are not supermodular: In Example 44, \(v(\{1, 3\}) + v(\{2, 3\}) \neq v(\{1, 2, 3\}) + v(\{2\})\).

8 Conclusions

We presented a class of games of resources that exploits the formalisms and reasoning methods for resource-sensitive logics. The language of Linear Logic allows us to represent in an harmonious way simultaneous resources, deterministic and non-deterministic choice, and crucially, resource-transforming capacities. A resource is a formula of Linear Logic.

In individual resource games, each player of a game is endowed with a multiset of resources and has an objective represented by a resource. In this context, we studied studied three decision problems, the
first of which is to decide whether a profile is a Nash equilibrium. Some profiles that are not equilibria can have desirable outcomes from the point of view of an external authority. Some equilibria can have outcomes that are undesirable. We thus studied redistribution schemes which can be used by a central authority to enforce some behavior, either by disincentivizing a behavior or incentivizing a behavior. This yielded two related decision problems: rational elimination and rational construction of profiles. We illustrated the models and the decision problems with two examples.

We considered dichotomous or parsimonious preferences, and showed striking algorithmic differences when the logic employed admits or not the weakening rule.

**Summary of complexity results.** For all decision problems, for both types of preferences, we have studied four cases where proof-search in \( \text{LOG} \) can have the following properties: affine vs. linear, and NP-complete vs. PSPACE-complete.

When \( \text{LOG} \) is NP-complete, we sum up precisely the results in Figure 4. For instance, one can quickly gather that when \( \text{LOG} \) is Affine MLL (whose sequent validity checking is NP-complete) and we consider parsimonious preferences, RATIONAL ELIMINATION is in \( \text{P}^{\text{NP}} \). We proved the same problem to be in \( \Sigma_2^p \) when \( \text{LOG} \) is Linear MLL. It is interesting to note that, although weakening usually does not change the complexity of the problem of sequent validity checking of the logics we considered,\(^9\) one have always been able to capitalize on its presence to simplify our solutions to the problems we studied here.

Putting the results of this paper together, it is also easy to see that we have this theorem.

**Theorem 48.** When \( \text{LOG} \) is PSPACE-complete, linear or affine, with dichotomous or with parsimonious preferences, all three decision problems are PSPACE-complete.

\(^9\)We did not consider full propositional Linear Logic, which also contains so-called ‘exponentials’. Weakening does make a difference: sequent validity in full propositional Linear Logic is undecidable [14], while sequent validity in full propositional Affine Logic is decidable [12].

|                | linear          | affine          |
|----------------|-----------------|-----------------|
| **NE**         | NP-hard (Prop. 9 in \( \Pi_2^p \) (Prop. 10) | NP-hard (Prop. 9 in \( \Pi_2^p \) (Prop. 13) |
| **RE**         | coNP-hard (Prop. 18 in \( \Sigma_2^p \) (Prop. 16) | coNP-hard (Prop. 18 in \( \Pi_2^p \) (Prop. 17) |
| **RC**         | NP-hard (Prop. 19 in \( \Sigma_2^p \) (Prop. 20) | NP-hard (Prop. 19 in \( \Sigma_2^p \) (Prop. 21) |
| **NE**         | coNP-hard (Prop. 26 in \( \Pi_2^p \) (Prop. 27) | coNP-hard (Prop. 26 in \( \Pi_2^p \) (Prop. 29) |
| **RE**         | coNP-hard (Prop. 32 in \( \Sigma_2^p \) (Prop. 30) | coNP-hard (Prop. 32 in \( \Sigma_2^p \) (Prop. 31) |
| **RC**         | coNP-hard (Prop. 33 in \( \Sigma_2^p \) (Prop. 34) | coNP-hard (Prop. 33 in \( \Sigma_2^p \) (Prop. 35) |

Table 4: Complexity results when the problem of validity checking in \( \text{LOG} \) is in NP.
First-Order MLL is one of these logics whose complexity of sequent validity is in \( \text{NP} \). On the other hand, sequent validity for First-Order MALL is \( \text{NEXPTIME} \)-complete. It is routine to adapt our proofs to show this theorem.

**Theorem 49.** When \( \text{LOG} \) is First-Order MALL, linear or affine, with dichotomous or with parsimonious preferences, all three decision problems are \( \text{NEXPTIME} \)-complete.

**Comparison with the related literature.** Electric Boolean Games [9] are an extension of Boolean games where playing a certain action has a numeric cost, and agents are endowed with a certain amount of ‘energy’. Deciding whether a profile is a Nash equilibrium in a Boolean game is \( \text{coNP} \)-complete [4]. In Electric Boolean Games, deciding whether a profile is rationally eliminable is \( \text{NP} \)-complete, while deciding whether a profile is rationally constructible is \( \text{coNP} \)-hard and in \( \Delta^p_2 \).

The trend is that the complexity of decision problems in individual resource games is higher than for their counterparts in Electric Boolean Games. An obvious exception is the problem to decide whether an individual resource game admits a Nash equilibrium when \( \text{LOG} \) is affine and we consider dichotomous preferences. The problem is trivial by Prop. 12, while it is \( \Sigma^p_2 \)-complete in Boolean games [4].

In Boolean games, goals of players are expressed as classical propositional formulas. Moreover, game outcomes or profiles are in fact models of classical propositional logic, i.e., valuations. Checking whether the goal of a player is satisfied in a game profile is an easy problem in Boolean games. This is also true in Electric Boolean Games. In contrast in resource games, checking whether the goal of a player is satisfied in a game profile is as hard as proof search in \( \text{LOG} \).

In individual resource games, there is no one-to-one correspondence between profiles and outcomes. This is another difference with Electric Boolean Game. As a consequence, the notions of elimination and construction in individual resource games add a bit of complexity by having to consider a set of profiles with the same outcomes.

With the decision problems of rational elimination and rational construction, there is a dimension of social choice theory and mechanism design. Formal frameworks concerned with redistribution schemes and economic policies can be found for instance in [9] again, or [6, 13, 15].

Finally, we mostly focused on individual games and looked at Nash equilibria. Nonetheless, the setting allows one to easily build classes of coalition games, as we have shown in Section 7. These classes of games are reminiscent of Coalitional Resource Games [31, 5] and of Coalition Skill Games [2]. In [27], we have started their study, with what we called Rich Coalitional Resource Games (RCRGs). All Individual Goal Coalition Resource Games, introduced in Section 7.1, correspond to one-goal RCRGs. Moreover, RCRGs are effectively an extension of Coalitional Resource Games.

**Perspectives.** We have obtained tight complexity results when \( \text{LOG} \) is \( \text{PSPACE} \)-complete. However, this is lacking when \( \text{LOG} \) is in \( \text{NP} \). We suspect that the complexity of the diverse decision problems generally lie above the lower bounds we have obtained. It is more likely that some proposed upper bounds are tight. One perspective will thus be to investigate whether some decision problems could be proven hard for some complexity class in the polynomial or Boolean hierarchy, for instance using the techniques from [29] of raising \( \text{NP} \) lower bounds to lower bounds for classes above \( \text{NP} \).

Resource games based on resource-sensitive logics become all the more significant when the resources are subject to transforming activities. We can exploit the existing research on these resource-sensitive logics about their proof theory. In particular, through the Curry-Howard correspondence between proofs and programs (see, e.g., [7]), an exciting perspective is the possibility to interpret the logical proofs as rigorous programs to be executed by the players. We can expect to obtain some results for the automated generation of plans, where the resources can be subjected to a series of transforming activities by the agents.
Moving from the models of resource games presented here, we have started to investigate a more amenable class of games, where each player $i$ is also assigned a set of “skills” in the shape of $k$ formulas of the form $A \rightarrow B$. The idea is that Player $i$ has $k$ actions, and can contribute $B$ if they can first transform some of their resource endowment into $A$.

Finally, we intend to pursue the work on cooperative resource games introduced in Section 7, with some advances already presented in [27].
A Sequent rules of Affine MALL

We present the sequent rules for Affine MALL. In what follows, \( A, B, A_0, \) and \( A_1 \) are formulas. \( \Gamma, \Gamma', \Delta, \) and \( \Delta' \) are sequences of zero or more formulas. A sequent rule has an upper and a lower part. The upper part is composed of zero, one, or two sequents. The lower part is composed of one sequent. If there is a proof of all the sequents of the upper part, then the rule can be used to obtain a proof of the sequent of the lower part.

Identities

\[
\frac{\quad}{A \vdash A} \quad \text{ax} \quad \frac{\Gamma, A \vdash \Delta \quad \Gamma', A \vdash \Delta'}{\Gamma, \Gamma' \vdash A, \Delta, \Delta'} \quad \text{cut}
\]

Structural Rules

\[
\frac{\Gamma, A, B, \Gamma' \vdash \Delta}{\Gamma, B, A, \Gamma' \vdash \Delta} \quad E \quad \frac{\Gamma \vdash \Delta, A, B, \Delta'}{\Gamma \vdash \Delta, B, A, \Delta'} \quad E
\]

\[
\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta} \quad W \quad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A} \quad W
\]

Negation

\[
\frac{\Gamma \vdash A, \Delta}{\Gamma, \lnot A \vdash \Delta} \quad \text{L} \quad \frac{\Gamma \vdash A, \Delta}{\Gamma \vdash \lnot A, \Delta} \quad \text{R}
\]

Multiplicatives

\[
\frac{\Gamma \vdash A, \Delta}{\Gamma, \Gamma' \vdash A \otimes B, \Delta, \Delta'} \quad \otimes \text{R} \quad \frac{\Gamma' \vdash A, \Delta'}{\Gamma, \Gamma' \vdash A \otimes B, \Delta} \quad \otimes \text{L}
\]

\[
\frac{\Gamma \vdash A, \Delta}{\Gamma, \Gamma' \vdash A \land B, \Delta, \Delta'} \quad \land \text{R} \quad \frac{\Gamma' \vdash A, \Delta'}{\Gamma, \Gamma' \vdash A \land B, \Delta} \quad \land \text{L}
\]

\[
\frac{\Gamma \vdash \Delta}{\Gamma, \lnot A \vdash \Delta} \quad \lnot \text{L} \quad \frac{\Gamma \vdash \Delta}{\Gamma, \lnot A \vdash \Delta} \quad \lnot \text{R}
\]

Additives (In \( \oplus \text{R}, \) and \( \& \text{L}, \) \( i \) stands for either 0 or 1.)

\[
\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \& B, \Delta} \quad \& \text{R} \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \& B, \Delta} \quad \& \text{L}
\]

\[
\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \oplus B, \Delta} \quad \oplus \text{L} \quad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash A \oplus B, \Delta} \quad \oplus \text{R}
\]

\[
\frac{\Gamma \vdash \Delta}{\Gamma \vdash \top, \Delta} \quad \top \text{R} \quad \frac{\Gamma \vdash \Delta}{\Gamma \vdash 0, \Delta} \quad 0 \text{L}
\]
B Elements of computational complexity

We need to assume some familiarity with computational complexity. This appendix only introduces the some elements of terminology and some definitions about complexity theory. The reader familiar with these notions can use this section for quick reference. Another reader can use it as a starting point and move to a more complete introduction. A classic introduction to computational complexity is [19]. All elementary complexity classes used in this paper are presented in [24].

A decision problem (or problem for short) is a problem that is posed as ‘yes’/’no’ question of the values of the input.

The class $P$ is the class of decision problems that can be solved in deterministic polynomial time (wrt. the size of the input). The class $NP$ is the class of problems that can be solved in non-deterministic polynomial time. The class $PSPACE$ is the class of problems that can be solved using a polynomial amount of space. The complement of a decision problem is the decision problem resulting from reversing the ‘yes’ and ‘no’ answers. For every class of complexity $C$, we note $coC$ the class populated with the complements of the problems in $C$. Given two classes of complexity $C_1$ and $C_2$, the class $C_1^{C_2}$ is the class of problems that are in $C_1$ if we assume the availability of an oracle to solve the problems in $C_2$. An oracle for $C_2$ is a black box capable to solve every problem in $C_2$ in a single operation. Queries to an oracle can be adaptive (also called serial), or non-adaptive (also called parallel). A query is adaptive when it depends on the answer of a previous query. Non-adaptive queries on the other hand, can be chosen in advance and computed from the start and are asked in parallel.

For every class of complexity $C$, we note $P^C$ (resp. $NP^C$) the class of problems solvable on a deterministic (resp. non-deterministic) polynomial-time bounded oracle Turing machine using an oracle set $C$. We note $P^{C[k]}$ and $NP^{C[k]}$ when at most $k$ adaptive queries to $C$ can be used. We note $P^{C||k}$ and $NP^{C||k}$ when at most $k$ non-adaptive queries to $C$ can be used.

We note $P^{C||}$ (resp. $NP^{C||}$) the class of problems solvable on a deterministic (resp. non-deterministic) polynomial-time bounded oracle Turing machine with non-adaptive queries to $C$. The class $P^{NP||}$ is also referred to as $\Theta^p_2$.

The polynomial hierarchy. The polynomial hierarchy contains a family of complexity classes that are smaller than $PSPACE$. The class $P$ lies at the bottom of the polynomial hierarchy. Then, for every positive integer $i$, we can define $\Delta^p_i$, $\Sigma^p_i$, and $\Pi^p_i$ recursively as follows:

- $\Delta^p_0 = \Sigma^p_0 = \Pi^p_0 = P$
- $\Delta^p_{i+1} = \Pi^p_i$
- $\Sigma^p_{i+1} = NP^{\Delta^p_i}$
- $\Pi^p_{i} = co\Sigma^p_{i}$

The Boolean hierarchy over $NP$. The Boolean hierarchy has been studied in [30, 11, 29]. The Boolean hierarchy over $NP$ contains a family of complexity classes that are smaller than $\Delta^p_2$. The class $NP$ lies at the bottom of the Boolean hierarchy over $NP$. Here, we are better off looking at complexity classes not as classes of decision problems, but as classes of languages. A language is the formal realization of a decision problem. Let $p$ be a decision problem with $k$ inputs. A language of $p$ is the language $L_p = \{ (a_1, \ldots, a_k) \mid p \text{ answers ‘yes’ of the input } (a_1, \ldots, a_k) \}$. Given a class of complexity $C$, we say that $L_p \in C$ iff $p \in C$. Then, given two classes of complexity $C_1$ and $C_2$, each representing a set of languages and the decision problems they formalize, we define $C_1 \wedge C_2 = \{ L_1 \cap L_2 \mid L_1 \in C_1 \text{ and } L_2 \in C_2 \}$ and $C_1 \lor C_2 = \{ L_1 \cup L_2 \mid L_1 \in C_1 \text{ and } L_2 \in C_2 \}$. In this context, the class $NP$ is the class of languages that can be recognised in non-deterministic polynomial time. Then, for every positive integer $i$, we can define $BH_i$ recursively as follows:
The class $\text{BH}_2 = \text{NP} \land \text{coNP}$ is the “difference class” $D^P$ presented in [18].

Useful properties. Besides the definitions, the following properties are useful:

- $C_1^{\text{co}C_2} = C_2^C$ (for all two classes $C_1$ and $C_2$);
- $\text{NP}^\Sigma = \Sigma_1^{p+1}$;
- $\text{co} \Sigma = \Pi_1^p$;
- $\text{NP} \Delta_p = \Sigma_p^p$;
- $\text{P} \Delta_p = \Delta_p^p$;
- $\Sigma_p^p \subseteq \text{PSPACE}$;
- $\text{PSPACE} = \text{coPSPACE} = \text{PSPACE}^{\text{PSPACE}} = \text{NP}^{\text{PSPACE}}$;
- $\text{BH}_i \subseteq \Delta_2^p$;
- $\text{P}^{\text{NP}||k} \subseteq \text{BH}_{k+1} \subseteq \text{P}^{\text{NP}||k+1}$.
References

[1] Natasha Alechina, Nils Bulling, Brian Logan, and Hoang Nga Nguyen. The virtues of idleness: A decidable fragment of resource agent logic. Artificial Intelligence, 245:56–85, 2017.

[2] Yoram Bachrach, David C. Parkes, and Jeffrey S. Rosenschein. Computing Cooperative Solution Concepts in Coalitional Skill Games. Artificial Intelligence, 204:1–21, 2013.

[3] Yoram Bachrach and Jeffrey S. Rosenschein. Coalitional Skill Games. In 7th International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS’08, pages 1023–1030. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[4] Elise Bonzon, Marie-Christine Lagasquie-Schiex, Jérôme Lang, and Bruno Zanuttini. Boolean Games Revisited. In 17th European Conference on Artificial Intelligence (ECAI’06), volume 141 of Frontiers in Artificial Intelligence and Applications, pages 265–269. IOS Press, 2006.

[5] Paul E. Dunne, Sarit Kraus, Efrat Manisterski, and Michael Wooldridge. Solving coalitional resource games. Artificial Intelligence, 174(1):20–50, 2010.

[6] Ulle Endriss, Sarit Kraus, Jérôme Lang, and Michael Wooldridge. Designing Incentives for Boolean Games. In 10th International Conference on Autonomous Agents and Multiagent Systems, AAMAS’11, pages 79–86. International Foundation for Autonomous Agents and Multiagent Systems, 2011.

[7] Dov M. Gabbay and Ruy J. G. B. de Queiroz. Extending the Curry-Howard Interpretation to Linear, Relevant and Other Resource Logics. Journal of Symbolic Logic, 57(4):1319–1365, 1992.

[8] Jean-Yves Girard. Linear logic. Theoretical Computer Science, 50(1):1–101, 1987.

[9] Paul Harrenstein, Paolo Turrini, and Michael Wooldridge. Electric Boolean Games: Redistribution Schemes for Resource-Bounded Agents. In 14th International Conference on Autonomous Agents and Multi-agent Systems, AAMAS’15, pages 655–663, 2015.

[10] Max I. Kanovich. The complexity of Horn fragments of Linear Logic. Annals of Pure and Applied Logic, 69(2-3):195–241, 1994.

[11] J. Köbler, U Schöning, and K. Wagner. The difference and truth-table hierarchies for NP. Theoretical Informatics and Applications, 21:419–435, 1987.

[12] A. P. Kopylov. Decidability of Linear Affine Logic. Information and Computation, 164(1):173–198, 2001.

[13] Vadim Levit, Tal Grinshpoun, Amnon Meisels, and Ana L.C. Bazzan. Taxation Search in Boolean Games. In 12th International Conference on Autonomous Agents and Multi-agent Systems, AAMAS’13, pages 183–190. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

[14] Patrick Lincoln, John Mitchell, Andre Scedrov, and Natarajan Shankar. Decision problems for propositional linear logic. Annals of Pure and Applied Logic, 56(1-3):239–311, 1992.

[15] Pavel G. Naumov and Jia Tao. A modal logic for reasoning about economic policies. Journal of Logic and Computation, 27(1):395–412, 2017.

[16] Peter W. O’Hearn and David J. Pym. The logic of Bunched Implications. Bulletin of Symbolic Logic, 5(2):215–244, 1999.
[17] Martin J. Osborne and Ariel Rubinstein. A course in game theory. The MIT Press, Cambridge, USA, 1994.

[18] C. H. Papadimitriou and M. Yannakakis. The complexity of facets (and some facets of complexity). Journal of Computer and System Sciences, 28(2):244–259, 1984.

[19] Christos Papadimitriou. Computational Complexity. Addison Wesley, 1994.

[20] Daniele Porello and Ulle Endriss. Modelling Combinatorial Auctions in Linear Logic. In Fangzhen Lin, Ulrike Sattler, and Miroslaw Truszczyński, editors, Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, pages 71–78. AAAI Press, 2010.

[21] Daniele Porello and Ulle Endriss. Modelling Multilateral Negotiation in Linear Logic. In ECAI 2010 - 19th European Conference on Artificial Intelligence, Proceedings, volume 215 of Frontiers in Artificial Intelligence and Applications, pages 381–386. IOS Press, 2010.

[22] Greg Restall. On Logics Without Contraction. PhD thesis, The University of Queensland, 1994.

[23] John C. Reynolds. Separation Logic: A Logic for Shared Mutable Data Structures. In Proceedings of the 17th Annual IEEE Symposium on Logic in Computer Science, LICS’02, pages 55–74, Washington, DC, USA, 2002. IEEE Computer Society.

[24] Jörg Rothe. Complexity Theory and Cryptology. Springer-Verlag Berlin Heidelberg, 2005.

[25] Anne S. Troelstra. Lectures on Linear Logic. CSLI Publications, 1992.

[26] Nicolas Troquard. Nash equilibria and their elimination in resource games. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, pages 503–509, 2016.

[27] Nicolas Troquard. Rich coalitional resource games. In Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018). AAAI Press, 2018.

[28] Yaron Velner, Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger, Alexander Moshe Rabinovich, and Jean-François Raskin. The complexity of multi-mean-payoff and multi-energy games. Inf. Comput., 241:177–196, 2015.

[29] Klaus W. Wagner. More Complicated Questions About Maxima and Minima, and Some Closures of NP. Theor. Comput. Sci., 51(1-2):53–80, March 1987.

[30] Gerd Wechsung. On the boolean closure of NP. In Lothar Budach, editor, Fundamentals of Computation Theory, volume 199 of Lecture Notes in Computer Science, pages 485–493. Springer Berlin Heidelberg, 1985.

[31] Michael Wooldridge and Paul E. Dunne. On the computational complexity of coalitional resource games. Artificial Intelligence, 170(10):835–871, 2006.