Types of Rights in Two-Party Systems: A Formal Analysis [1]

Gordon J. PACE [a] Fernando SCHAPACHNIK [b]

[a] *gordon.pace@um.edu.mt*
University of Malta, Malta
[b] *fschapachnik@dc.uba.ar*
Departamento de Computación, FCEyN,
Universidad de Buenos Aires, Buenos Aires, Argentina

Abstract. We present a formalization of Kanger's types of rights in the context of interacting two-party systems, such as contracts. We show that in this setting basic rights such as *claim*, *freedom*, *power* and *immunity* can be expressed in terms of (possibly negated) permissions and obligations over presence or absense of actions. Another way of saying this is that, at least in the context of contracts, neither *claim*, nor *power*, nor *freedom* nor *immunity* are foundational modalities, as they can be defined in terms of others. We also show that the set of atomic type rights is different from Kanger's original proposal.

Keywords. Automated Legislative Drafting, Contract Verification, Kanger's Types

1. Introduction

Deontic modalities such as permission and obligation have been debated exhaustively in the literature, and although a final consensus has not been achieved, there is at least agreement over their basic properties. This is not the case with more intricate concepts such as Hohfeld's *claim right*, *power*, *freedom* and *immunity* [2].

Kanger *et al.* [4] attempted to clarify Hohfeld's modalities, but a lack of formal underlying semantics somewhat limits his work. Others, surveyed in Section 6, addressed this aspect, yet not always fully formalising the intricate modalities and other derivative ones, such as intention and causality, which arise in this context. Part of the difficulty, we believe, can be accounted for if the context is limited. We restrict ourselves to a setting that is both specific and interesting: interacting two-party systems, also commonly known as *contracts*. Contracts are prevalent enough so their analysis becomes of practical importance, yet restrictive enough so there is a clear boundary for the analysis. In such systems, the interactive nature of the parties gives rise to potential cooperation and interference, allowing us to reach conclusions separately for both the individual parties and the result of their combination.

In Kanger *et al.* [4] rights are state-based, and dependant on a notion of causality which interacts with directed rights in a non-obvious manner. For instance, it is not im-

[1] Partially supported by UBACyT 20020100200103.

mediately clear how a statement such as *'it shall be that party p causes S'* is to be interpreted in a system where, for instance, the other party can interfere with p's intention to bring about state S. However, in computer science, concurrent and synchronous composition have been studied for a number of decades, and these notions directly address interaction and interference from an action-based perspective.

In [7] we have looked at how synchrony can be applied in a contract setting — using a formal automaton-based model in which two parties synchronise over a set of actions, contracts are given a precise semantics. In this paper we apply the model to be able to study how Kanger's rights apply in such a setting. The synchronous nature of composition adopted, which forces the parties to agree on actions to perform, brings about a setting subtly different from the one originally presented in Kanger *et al.* [4], in particular because rights and obligations affect both parties. E.g., if a party has an obligation to perform a particular action, then the other party must provide her with a way of achieving this. Just as in Kanger *et al.*, we proceed to study the different compound types of rights in this setting. Unsurprisingly, our different, mostly stronger, modalities bring to light further conflicts in contract clauses, and thus induce a different set of possible rights.
The main contributions of this paper are:

- Giving formal semantics to Kanger's types of right in the context of action-based, interacting two-party systems.
- Showing that, at least in this context, neither claim, nor power, nor freedom nor immunity are foundational modalities, as they can be defined in terms of possitive and negative permissions and obligations, over presence or lack of actions.
- Proving that the number of atomic types (maximally consistent sets of rights) is reduced in this context.

The rest of the paper is organised as follows. The next section formalises our notions of automata, deontic operators, contracts and contracts' strength, which allows us to show, in Section 4, that some contracts cannot be satisfied at the same time and thus lead to a conflict. In Section 3 we interpret Kanger's work in the setting of interacting two-party systems and compare Kanger's modalities strength diagram with ours, explaining why they differ, comparing atomic types under both proposals in Section 5. Finally, in Section 6 we discuss related work and conclude in Section 7.

2. Background

To enable direct reasoning about contracts, one requires a model in which the two parties somehow interact to agree on which actions to perform. In [7] we presented such a model, based on the notion of synchronous composition [1] and multi-action labels on transitions, since otherwise it would be impossible not to violate a contract in which both parties have different obligations at the same time. This section summarises the key aspects of our model.

Definition 1 *A multi-action automaton S is a tuple $\langle \Sigma,\ Q,\ q0,\ \rightarrow \rangle$, where Σ is the alphabet of actions, Q is the set of states, $q0 \in Q$ is the initial state and $\rightarrow \subseteq Q \times 2^{\Sigma} \times Q$ is the transition relation. We will write acts(q) to be the set of all action sets on the outgoing transitions from q (defined to be $\{A \mid \exists q' \cdot q \xrightarrow{A} q'\}$).*

The synchronous composition of two automata $S_i = \langle Q_i,\ q0_i,\ \rightarrow_i \rangle$ for $i \in \{1, 2\}$ (both with alphabet Σ) synchronising over alphabet G, written as $S_1 \|_G S_2$, is defined to *be* $\langle Q_1 \times Q_2,\ (q0_1, q0_2), \rightarrow\rangle$*, where* $\rightarrow$ *is the classical synchronous composition relation defined in [1]. We will assume that all systems are well-formed, i.e., do not deadlock.*

We define contracts to be automata with each state tagged with the clauses which will be in force at that point. Contract clauses are either (i) obligation clauses of the form $\mathcal{O}_p(a)$ or $\mathcal{O}_p(!a)$, which say that party p is obliged to perform or not perform action a respectively; or (ii) permission clauses which can be either of the form of $\mathcal{P}_p(a)$ or $\mathcal{P}_p(!a)$ (party p is permitted to perform, or not perform action a respectively).

Definition 2 *A* contract automaton *is a total and deterministic multi-action automaton* $S = \langle Q,\ q0,\ \rightarrow\rangle$*, together with a total function contract* $\in Q \rightarrow 2^{\mathit{Clause}}$ *assigning a set of clauses to each state.*

2.1. Contract Satisfaction

Given a two-party system (S_1, S_2), and a contract automaton $\mathcal{A}$, we define whether or not a party violates the contract when a particular state is reached or a transition is taken.

Definition 3 *Functions* $O_p(q_{\mathcal{A}})$ *and* $F_p(q_{\mathcal{A}})$ *give the set of actions respectively obliged to be performed and obliged not to be performed by party p. They are defined in terms of the contract clauses in the state. Action set A is said to be viable for party p in a contract automaton state* $q_{\mathcal{A}}$*, written as viable*$_p(q_{\mathcal{A}}, A)$*, if (i) all her obliged actions are included in A but; (ii) no actions which the party is obliged not to perform are included in A:* $\mathit{viable}_p(q_{\mathcal{A}}, A) \stackrel{df}{=} O_p(q_{\mathcal{A}}) \subseteq A \wedge F_p(q_{\mathcal{A}}) \cap A = \emptyset$

To be able to place blame in case of a violation, we parametrise contract satisfaction by party. It is also worth noting that while obligation to perform an action, for instance, is violated in a transition which does not include the action, permission is violated by a state in which the opportunity to perform the permitted action is not present. The satisfaction predicate will thus be overloaded to be applicable to both states and transitions. The predicate $\mathit{sat}_p^{\mathcal{A}}(X)$ will denote that the contract automaton $\mathcal{A}$, reaching state X or traversing transition X, does not constitute a violation for party p.

Permission. If party p is permitted to perform shared action a, then the other party $\overline{p}$ must provide p with at least one viable outgoing transition which contains a but does not include any forbidden actions. Permission to perform local actions cannot be violated. In the case of a single permission, this can be expressed as follows:

$$(q_1, q_2)_{q_{\mathcal{A}}} \vdash_{\overline{p}} \mathcal{P}_p(a) \stackrel{df}{=} a \in G \implies \exists A \in \mathit{acts}(q_{\overline{p}}),\ A' \subseteq G^c \cdot a \in A \wedge \mathit{viable}_p(q_{\mathcal{A}}, A \cup A')$$

Similarly, if party p is permitted to not perform action a (i.e. permitted to perform some action set which does not include action a), then the other party $\overline{p}$ must provide p with at least one viable outgoing transition which does not include a nor any forbidden action. Permission to perform local actions can never be violated. In the case of a single permission, this can be expressed as follows:

$$(q_1, q_2)_{q_{\mathcal{A}}} \vdash_{\overline{p}} \mathcal{P}_p(!a) \stackrel{df}{=} a \in G \implies \exists A \in \mathit{acts}(q_{\overline{p}}),\ A' \subseteq G^c \cdot a \notin A \wedge \mathit{viable}_p(q_{\mathcal{A}}, A \cup A')$$

While actual obligation violations occur when an action is not performed, violations of a permission occur when no appropriate action is possible.
To combine all permissions in a state, we simply take the conjunction of all conditions:

$$\mathit{sat}^P_{\overline{p}}((q_1, q_2)_{q_\mathcal{A}}) \stackrel{df}{=} \forall \mathcal{P}_p(x) \in q_\mathcal{A} \cdot (q_1, q_2)_{q_\mathcal{A}} \vdash_{\overline{p}} \mathcal{P}_p(x)$$

Obligation. Obligation brings in constraints on both parties. Given that party p is obliged to perform action a in a state means that (i) party p must include the action in any outgoing transition in the composed system in which it participates; and (ii) the other party $\overline{p}$ must provide a viable synchronisation action set which, together with other asynchronous actions performed by p, allows p to perform *all* its obligations, positive and negative. Obligation to not perform action a ($\mathcal{O}_p(!a)$) can be similarly expressed. We combine all positive and negative obligations in the following definition:

$$\mathit{sat}^O_p((q_1, q_2)_{q_\mathcal{A}} \xrightarrow{A} (q_1', q_2')_{q_\mathcal{A}'}) \stackrel{df}{=} \mathit{viable}_p(q_\mathcal{A}, A)$$
$$\mathit{sat}^O_{\overline{p}}((q_1, q_2)_{q_\mathcal{A}}) \stackrel{df}{=} \exists A \in \mathit{acts}(q_{\overline{p}}),\ A' \subseteq G^c \cdot \mathit{viable}_p(q_\mathcal{A}, A \cup A')$$

We can now define the rest of the deontic modalities:

- Party p not being permitted to perform an action is equivalent to p being obliged not to perform the action: $!\mathcal{P}_p(a) \stackrel{df}{=} \mathcal{O}_p(!a) \qquad !\mathcal{P}_p(!a) \stackrel{df}{=} \mathcal{O}_p(a)$
- Party p not being obliged to perform an action is equivalent to p being permitted not to perform the action: $!\mathcal{O}_p(a) \stackrel{df}{=} \mathcal{P}_p(!a) \qquad !\mathcal{O}_p(!a) \stackrel{df}{=} \mathcal{P}_p(a)$
- Prohibition contract clauses $\mathcal{F}_p(a)$ and $\mathcal{F}_p(!a)$, prohibiting party p from performing and not performing a respectively, can be expressed in terms of permission:
$$\mathcal{F}_p(a) \stackrel{df}{=} !\mathcal{P}_p(a) \qquad \mathcal{F}_p(!a) \stackrel{df}{=} !\mathcal{P}_p(!a)$$

From these definitions, it follows that prohibition to perform an action is equivalent to obligation not to perform the action: $\mathcal{F}_p(x) = \mathcal{O}_p(!x)$.

2.2. Contract Strength

Definition 4 *A party p is said to be incapable of breaching a contract in a regulated two-party system, if p cannot be in violation in any of the reachable states and transitions of the system.*

A contract automaton $\mathcal{A}'$ is said to be stricter than *contract automaton $\mathcal{A}$ for party p (or $\mathcal{A}$ said to be* more lenient *than $\mathcal{A}'$ for party p), written as $\mathcal{A} \sqsubseteq_p \mathcal{A}'$, if for any systems S_1 and S_2, p being incapable of breaching $\mathcal{A}'$ implies that p is incapable of breaching $\mathcal{A}$. We say that two contract automata $\mathcal{A}$ and $\mathcal{A}'$ are equivalent for party p, written as $\mathcal{A} =_p \mathcal{A}'$, if $\mathcal{A} \sqsubseteq_p \mathcal{A}'$ and $\mathcal{A}' \sqsubseteq_p \mathcal{A}$. We define global contract strictness $\mathcal{A} \sqsubseteq \mathcal{A}'$ to hold if $\mathcal{A} \sqsubseteq_p \mathcal{A}'$ holds for all parties p, and similarly global contract equivalence $\mathcal{A} = \mathcal{A}'$.*

Definition 5 *Given two contract clauses C and C', the relation over contract automata $[C \to C'] \subseteq \mathcal{CA} \times \mathcal{CA}$ relates two contract automata $\mathcal{A}$ and $\mathcal{A}'$ if $\mathcal{A}$ is equivalent to $\mathcal{A}'$ except possibly for a number of instances of clause C replaced by C'.*

We extend the notion of strictness to contract clauses. We say that clause C' is stricter than clause C for party p, written as $C \sqsubseteq_p C'$, if for any contract automata $\mathcal{A}$ and $\mathcal{A}'$ such that $(\mathcal{A}, \mathcal{A}') \in [C \to C']$, it follows that $\mathcal{A} \sqsubseteq_p \mathcal{A}'$.

3. Kanger Rights in a Two-Party Setting

Kanger's paper [4] investigated the notion of rights in a general setting. Although the rights are directed between parties (e.g. party p has versus party $\overline{p}$ a claim that $S(p, \overline{p})$), the interaction between the parties and directionality of the rights depends on various other notions such as causality, interference and intention. The synchronous two-party approach we presented in the previous section gives a closed-world view for rights, that allows these notions to be formalised in a straightforward manner. In this section, we explore how Kanger's rights translate into this setting.

3.1. Actions and States

Kanger *et al.* [4] presents rights to be over a state of affairs, which is clearly a state-based look, but also identifies whether or not a party is responsible for causing a state to hold — indicating that there is an underlying notion of a party performing an action which leads to the state predicate holding. In synchronous systems, the parties involved synchronise over actions, making the approach inherently action-based. There are various standard ways in which one can encode state using actions and vice-versa.

One possible encoding from states to actions is to identify two special mutually exclusive actions $S^{\uparrow}$ and $S^{\downarrow}$ which cause S to start holding (become true) and to stop holding (become false) respectively. This approach is further elaborated upon in [8].

Two important properties of these actions are that: (i) the actions are mutually exclusive — the system may never perform $S^{\uparrow}$ and $S^{\downarrow}$ together; and (ii) the causality actions for the negation of a state $\neg S$ are the opposite of those of S i.e. $(\neg S)^{\uparrow} = S^{\downarrow}$ and $(\neg S)^{\downarrow} = S^{\uparrow}$.

3.2. Kanger et al.

Kanger *et al.* [4] identify eight simple types of rights:

(a) Party p has versus party $\overline{p}$ a *claim* that $S(p, \overline{p})$.
(b) Party p has versus party $\overline{p}$ a *freedom* that $S(p, \overline{p})$.
(c) Party p has versus party $\overline{p}$ a *power* that $S(p, \overline{p})$.
(d) Party p has versus party $\overline{p}$ a *immunity* that $S(p, \overline{p})$.
(a′) Party p has versus party $\overline{p}$ a *counter-claim* that $S(p, \overline{p})$.
(b′) Party p has versus party $\overline{p}$ a *counter-freedom* that $S(p, \overline{p})$.
(c′) Party p has versus party $\overline{p}$ a *counter-power* that $S(p, \overline{p})$.
(d′) Party p has versus party $\overline{p}$ a *counter-immunity* that $S(p, \overline{p})$.

The first four can be considered as the fundamental rights, with the other four (the *counter* rights) being identical except that they refer to the negation of state predicate S.[2] Thus, for example, saying that *'party p has versus party $\overline{p}$ a counter-claim that state $S(p, \overline{p})$ holds'* is identical to saying that *'party p has versus party $\overline{p}$ a claim that not-$S(p, \overline{p})$'*.

[2]It is worth noting that since, in our context, we have only two parties interacting (p and $\overline{p}$), we need not make explicit (i) the party versus whom the right is; and (ii) the parameters of state predicate S. We can thus write statements just as *'party p has a claim that S'*.

3.3. Semantics

A discussion of the intuitive meaning of these different types of rights can be found in the original paper [4] or any of many papers discussing and extending these notions (see Section 6). However, Kanger *et al.* identifies the interpretation of rights (a) to (d) as:

(1a) It shall be that $\overline{p}$ causes that $S(p,\overline{p})$.
(1b) Not: it shall be that p causes that not-$S(p,\overline{p})$.
(1c) Not: it shall be that not: p causes that $S(p,\overline{p})$.
(1d) It shall be that not: $\overline{p}$ causes that not-$S(p,\overline{p})$.

Furthermore, Kanger *et al.* note that the statement *'Not: it shall be that not: ...'* is synonymous to *'It may be that ...'*. This allows us to rewrite the formulae without top-level negations. Using the relationship between states and actions as identified in Section 3.1, these are equivalent to:

(2a) It shall be that $\overline{p}$ performs $S^{\uparrow}$.
(2b) It may be that not: p performs $S^{\downarrow}$.
(2c) It may be that p performs $S^{\uparrow}$.
(2d) It shall be that not: $\overline{p}$ performs $S^{\downarrow}$.

The *shall be* and *may be* modalities correspond to our notions of obligation and permission, enabling us to define the different forms of rights in our formal model[3]:

$$\text{Cl}(p,\overline{p},S) \stackrel{df}{=} \mathcal{O}_{\overline{p}}(S^{\uparrow}) \qquad \text{Po}(p,\overline{p},S) \stackrel{df}{=} \mathcal{P}_{p}(S^{\uparrow})$$

$$\text{Fr}(p,\overline{p},S) \stackrel{df}{=} \mathcal{P}_{p}(!S^{\downarrow}) \qquad \text{Im}(p,\overline{p},S) \stackrel{df}{=} \mathcal{O}_{\overline{p}}(!S^{\downarrow})$$

Counter rights replace $S^{\uparrow}$ for $S^{\downarrow}$ and vice-versa.

One interesting observation emerging from this formalisation is that in a two party system some of the rights place constraints on both parties. For instance, if p has versus $\overline{p}$ a claim that S, $\overline{p}$ has an obligation to perform $S^{\uparrow}$. If $S^{\uparrow}$ is an action local to $\overline{p}$, then no constraint is placed on p, but if it is a common action, the semantics of obligation insist that p allows $\overline{p}$ to perform $S^{\uparrow}$. For example, consider S to be *'p has access to the webservice'*. Now, to make the predicate hold, $S^{\uparrow}$ may be the action *openPort* which opens a particular port. If this action is local to $\overline{p}$ (i.e., $\overline{p}$ can perform the action independently of p), then the constraint lies solely on $\overline{p}$. However, if *openPort* is a shared action, then the semantics of obligation place a restriction on party p to provide a feasible action set through which $\overline{p}$ may use to satisfy its obligation to open the port. In other words, although p does not necessarily have to use the webservice, it must support $\overline{p}$ in opening the port.

In fact, the moment we are giving semantics to Kanger's types using the interactive two-party systems, the two models diverge. For example, in Kanger *et al.* the following two types are compatible (not in conflict):

$$\text{Cl}(p,\overline{p},S) \text{ and } !\text{Po}(p,\overline{p},S)$$

Their informal meaning, when transformed to reason about actions becomes:

it shall be that $\overline{p}$ performs $S^{\uparrow}$
it shall be that p does not perform $S^{\uparrow}$

---

[3]Using the notation used in Kanger *et al.*, we write Cl for Claim, Po for Power, Fr for Freedom and Im for Immunity.

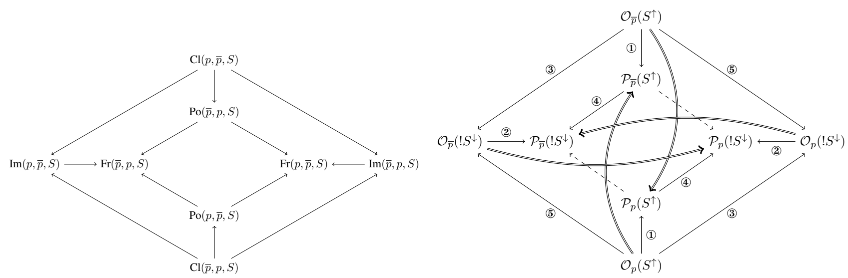

Figure 1. (a) The original Kanger *et al.* strength diagram (left); and (b) the strength diagram for interacting parties (right)

In a non-interactive system, Kanger's view is applicable and the clauses are compatible. However, in an interactive system, where $S^{\uparrow}$ is part of the synchronisation alphabet, these clauses become $\mathcal{O}_{\overline{p}}(S^{\uparrow})$ and $\mathcal{O}_{p}(!S^{\uparrow})$, which can be proved to be conflicting.

Implicit in Kanger *et al.* is that these 8 basic right types are complete — in that through their combination one can express all forms of rights a party may have. However, the formalisation we have given clearly shows that each basic right can be expressed by choosing: (i) the modality — is it a permission or an obligation?; (ii) the party to which the modality applies — is it p or $\overline{p}$?; (iii) the change in the value of S — is it $S^{\uparrow}$ or $S^{\downarrow}$?; and (iv) whether it is the presence or absence of that action that is of interest — e.g. is it $S^{\uparrow}$ or $!S^{\uparrow}$? These four different variables indicate that one can identify 16, not 8 basic right types. The missing ones, however, correspond to obligations on party p, and permissions for party $\overline{p}$, neither of which can be considered to be *rights of* p (in fact, in an interactive context, they are rights of the other party $\overline{p}$). This justifies the argument for completeness of Kanger *et al.*'s basic types.

3.4. *Strengths of Rights*

Given the 8 basic rights, one can construct 2^8 combinations over a particular state of affairs. However, not all these combinations are possible, since (i) some rights are subsumed by others; and (ii) some combinations of rights lead to conflicts (see Section 4). To address the first issue, Kanger provided a partial order on the basic rights in terms of their strength, as can be seen in Figure 1(a). An arrow from a right R to a right R' indicates that R is stronger than R', not unlike our notion of R being stricter than R'. We can apply the formalised versions in a two-party setting to investigate which parts of this strength relation are preserved.

Figure 1(b) is the corresponding strength diagram interpreted for two-party systems. The rights are replaced by their definitions, and the strength arrows revised as required. In fact, for a two party system most of the strictness inequalities still hold:

1. Ones marked ① are of the form $\mathcal{P}_p(a) \sqsubseteq \mathcal{O}_p(a)$, while those marked ② are of the form $\mathcal{P}_p(!a) \sqsubseteq \mathcal{O}_p(!a)$. In both cases they follow from the fact that in our model obligation is stricter than permission ($\mathcal{P}_p(a) \sqsubseteq \mathcal{O}_p(a)$) [7].
2. Those marked ③ are of the form $\mathcal{O}_p(!a) \sqsubseteq \mathcal{O}_p(b)$ while those marked ④ are of the form $\mathcal{P}_p(!a) \sqsubseteq \mathcal{P}_p(b)$, in both cases with mutually exclusive actions a and b. These hold by another theorem of the model: if a and b are mutually exclusive actions, then it holds that $\mathcal{O}_p(!a) \sqsubseteq \mathcal{O}_p(b)$ and $\mathcal{P}_p(!a) \sqsubseteq \mathcal{P}_p(b)$ (see [7]).

3. Ones marked ⑤ are of the form $\mathcal{O}_p(!a) \sqsubseteq \mathcal{O}_{\overline{p}}(b)$ and thus follow from the fact that if a and b are mutually exclusive actions, then $\mathcal{O}_{\overline{p}}(!b) \sqsubseteq \mathcal{O}_p(a)$ (see [7]).

However, there are differences with the original diagram:

1. The dashed-line arrows connecting power (Po$(p, \overline{p}, S)$) with freedom (Fr$(\overline{p}, p, S)$) would require the strictness inequality: $\mathcal{P}_p(!a) \sqsubseteq \mathcal{P}_{\overline{p}}(b)$, with a and b being mutually exclusive actions, which is shown not to hold in [7, Sect. 2.6].
2. The double-line arrows connecting immunity (Im$(p, \overline{p}, S)$) with freedom (Fr$(p, \overline{p}, S)$) and claim (Cl$(p, \overline{p}, S)$) with power (Po$(p, \overline{p}, S)$), however now follow because for synchronised actions, obligation for one party is stricter than permission for the other $\mathcal{P}_p(a) \sqsubseteq \mathcal{O}_{\overline{p}}(a)$ (see [7]).

4. Conflicts

Contract clauses are not always compatible with one another, giving rise to the notion of *conflict* between contract clauses [7]. Here we summarise the main aspects of our definition.

As expected, the obligation on a party to perform an action a and the obligation on the same party not to perform the same action can never be satisfied together. Another interesting example is that of $\mathcal{P}_p(!a)$ and $\mathcal{O}_p(a)$. Due to the multi-action semantics we adopt, the possibility of doing something other than a conflicts with the obligation of doing a: to satisfy the permission party $\overline{p}$ must provide a-free action sets which allow p to satisfy her obligations, but that requires that they contain a. We use the notation $C \maltese C'$ to denote that contracts C and C' are in conflict.

The basic properties are that opposite permissions conflict ($\mathcal{P}_p(x) \maltese !\mathcal{P}_p(x)$) and that obligation to perform mutually exclusive actions ($a \Join b$) also conflict ($a \Join b \vdash \mathcal{O}_p(a) \maltese \mathcal{O}_p(b)$), together with closure under symetry and increased strictness. The following propositions can be derived:

- Opposite obligations conflict with each other: $\mathcal{O}_p(x) \maltese !\mathcal{O}_p(x)$.
- Obligation to perform an action conflicts with both permission and obligation to not perform it: (i) $\mathcal{O}_p(x) \maltese \mathcal{P}_p(!x)$; and (ii) $\mathcal{O}_p(x) \maltese \mathcal{O}_p(!x)$. Obligation to perform an action also conflicts with lack of permission to perform the action: (iii) $\mathcal{O}_p(x) \maltese !\mathcal{P}_p(x)$.
- Given two conflicting clauses $C_1 \maltese C_2$, making the two clauses stricter does not resolve the conflict: if $C_1 \sqsubseteq C_1'$ and $C_2 \sqsubseteq C_2'$, then $C_1' \maltese C_2'$.

5. Atomic Types of Rights

Kanger *et al.* proceed to identify the so called *atomic types of rights* — given one has 8 possible basic types of rights, one can describe the rights regarding a particular state by identifying which of the basic rights hold, and which do not (their negation holds). This yields 256 possible atomic rights, but since some of the combinations are conflicting, Kanger *et al.* use their strength diagram to show that no more than 26 distinct combinations can be identified. Furthermore, since some of the rights or their negations imply each other, the sets of atomic rights can be simplified by removing the weaker clauses. A set of non-conflicting basic types which cannot be simplified any further is said to be complete.

As we have shown, in an interacting two-party setting, the strength diagram induced is somewhat different, which in turn leads to different atomic rights. In fact, it can be shown that in a two-party setting, one can now identify just 22 atomic types as listed below. A tick means that the basic right appears positively, while a cross means that it appears negated.

Claim	✓	✓	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕
Freedom			✓	✓	✓	✓	✓	✕	✕													
Power			✕	✕	✕	✕	✕	✕	✕	✓	✓	✓	✓	✓	✓	✓	✓	✕	✕	✕	✕	
Immunity			✓	✓	✕	✕	✕	✕	✕	✓	✓	✓	✓	✕	✕	✕	✕	✓	✓	✕	✕	
Counter-claim	✕	✕	✕	✕	✓	✕	✕	✓	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕	✕
Counter-freedom	✓	✕								✓	✓			✓	✓							✕
Counter-power	✕	✕	✓	✓		✓	✓			✕	✕	✓	✓	✕	✕	✓	✓	✕	✕	✕	✕	✕
Counter-immunity	✕	✕	✓	✕		✓	✕			✓	✕	✓	✕	✓	✕	✓	✕	✓	✕	✓	✕	✕

Unsurprisingly, this gives a very different view of atomic rights, with for instance most including no claim, but only two including no freedom.

Kanger reduced his 26 types to 10, so the rest can be obtained by *inversions* ($S^{\uparrow}$ becomes $S^{\downarrow}$ and viceversa) and *conversions* (swapping parties). In our model conversions do not reduce the number of clauses, because modalities applied to one party also constrain the other (in a different way). Inversions, however, do reduce our 22 to the 13 columns shaded in the table above.

6. Related Work

Makinson [6] proposed a representation of Kanger's types $((\pm)O(\pm)\binom{x}{y}$ do $(\pm)S)$ which is strikingly similar to ours. It gives 16 possibilities, just like our analysis. Makinson does not provide formal semantics, deals with a state-of-affairs type of logic, does not analyse Kanger's atomic types and does not work with contracts, although he does analyse that there might be two parties, one bearing the right and the other being the counterparty.

Makinson also shifts the view when addressing *power*, presenting the modern assumption, followed by most authors nowadays, where a power is a permission to dynamically bring about changes in the deontic norms that are valid in a particular state. Actions that modify contracts are beyond the scope of our work. Jones and Sergot [3] take over the analysis of power, specially institutionalised power, introducing a modality to express that an agent brings about a state of affairs (E_xS), which allows them to state that an agent should bring about that another one brings about a particular state (E_xE_yS).

The number of atomic types is a subject for debate. Kanger *et al.* [4] presents 26, later extended to 35 by Lindahl's [5]. The same work takes them to 127 if *collectivistic* propositions are considered.[4] Sergot [10] presents a detailed comparative analysis. Neither of them works in the context of interacting two-party systems, where deontic modalities applied to a party also place onus over the other, thus reducing the number of consistent atomic types, as explained in Sections 3 and 4. In keeping with Kanger, we identified only atomic types which include all the rights — whether positively or negatively.

[4]Collectivistic propositions are the ones that place the burden of obligation in more than one agent i.e., *'it is mandatory that agent a or agent b perform action c'*.

However, in our setting, further analysis can be performed to consider the possibilities when a particular right is not present — neither positively nor negatively.

From a semantics point of view, most of the attempts at formalising the Hohfelian concepts went no further than structured language, leaving many questions unresolved. For instance, Sartor [9] introduces the concept of directed modalities to express sentences like *'It is obligatory that Tom pays Mary $1000 in order to advance Mary's interests'*. If Mary is using the money to pay a blackmailer or to buy cancer-causing cigarettes, is she advancing her interests? According to whom? Can Tom deny the paying claiming that she would not use the money *'to advance her interests'*? Sergot [10] is more precise about which operator combinations are consistent given a few assumptions about the underlying logic, but because it only considers some basic modalities, and because the logic is not fixed, we still do not know if, for example, being empowered but forbidden makes any sense.

7. Conclusions

In this article we give formal semantics to Kanger's types of rights in the context of action-based and interacting two-party systems. Doing so allows to show that, in this context, the number of atomic types (maximally consistent sets of rights) is reduced, as compared to Kanger's *et al.* original formulation.

Also interestingly, all of Kanger's rights (claim, power, freedom, immunity and their negated versions) can be expressed in terms of positive and negative permission and obligation, over presence or lack of actions. Another way of saying this is that, at least in the context of contracts, neither claim, nor power, nor freedom nor immunity are foundational modalities, as they can be defined in terms of others.

An interesting next step would be to present an automata-based formalism in which there are multiple parties, but also general obligations and permissions. This would allow us to reason about general obligations (such as *'forbidden to kill'*), and how they interact with contracts.

References

[1] André Arnold. Nivat's processes and their synchronization. *Theor. Comput. Sci.*, 281:31–36, June 2002.

[2] W.N. Hohfeld. Some fundamental legal conceptions as applied in judicial reasoning. *Yale Lj*, 23:16, 1913.

[3] A.J.I. Jones and M. Sergot. A formal characterisation of institutionalised power. *Logic Journal of IGPL*, 4(3):427, 1996.

[4] S. Kanger and H. Kanger. Rights and parliamentarism. *Theoria*, 32(2):85–115, 1966.

[5] L. Lindahl. *Position and change: A study in law and logic*, volume 112. Springer, 1977.

[6] D. Makinson. On the formal representation of rights relations. *Journal of philosophical Logic*, 15(4):403–425, 1986.

[7] Gordon Pace and Fernando Schapachnik. Contracts for interacting two-party systems. In *FLACOS 2012: Sixth Workshop on Formal Languages and Analysis of Contract-Oriented Software*, sep 2012.

[8] Gordon Pace and Fernando Schapachnik. Types of rights in interacting two-party system: A formal analysis. Technical report, FCEyN, Universidad de Buenos Aires, feb 2012. http://tinyurl.com/8yfshtw.

[9] G. Sartor. Fundamental legal concepts: A formal and teleological characterisation*. *Artificial Intelligence and Law*, 14(1):101–142, 2006.

[10] M. Sergot. A computational theory of normative positions. *ACM Transactions on Computational Logic (TOCL)*, 2(4):581–622, 2001.