# Constrained Derivation in Assumption-Based Argumentation

Giovanni Buraglio[1,*], Wolfgang Dvořák[1], Anna Rapberger[2,*] and Stefan Woltran[1]

[1] *TU Wien, Institute of Logic and Computation, Austria*
[2] *Imperial College London, Department of Computing, United Kingdom*

**Abstract**
Structured argumentation formalisms provide a rich framework to formalise and reason over situations where contradicting information is present. However, in most formalisms the integral step of constructing all possible arguments is performed in an unconstrained way and is thus not under direct control of the user. This can hinder a solid analysis of the behaviour of the system and makes explanations for the results difficult to obtain. In this work, we introduce a general approach that allows constraining the derivation of arguments for assumption-based argumentation.

**Keywords**
Assumption-Based Argumentation, Normative Reasoning, Non-monotonic Reasoning

## 1. Introduction

Assumption-based argumentation (ABA) [1, 2, 3] is a well-studied formalism in the realm of structured argumentation with applications ranging from medical reasoning and decision-making to eXplainable AI [4, 5, 6]. Argumentative reasoning is hereby performed by instantiating ABA frameworks (ABAF) representing debates through (structured) arguments and an attack relation among them. Arguments are built as forward derivation supported by defeasible sentences called *assumptions*, using (strict) inference rules from the underlying knowledge base. Accordingly, attacks between arguments encode a consistency check among the assumptions that support them. As already noticed by Modgil and Prakken [7], assumption-based argumentation leaves the "set of inference rules unspecified" in the sense that rules are treated equally and no distinction can be made among them. However, in some domains of application, rules might be distinguished on the basis of their function. Such situations can be found, for instance, in the area of normative reasoning. There it may become relevant to tell apart rules that produce obligations and permissions on the one hand from those that produce institutional facts on the other, based on Searle's famous distinction between regulative and constitutive norms [8]. To prevent instances of deontic paradoxes and fallacious conclusions, the combination of rules is subject to certain restrictions [9]. In the context of multi-agent systems [10, 11], an agent's frame of reference may differ from that of others, giving rise to individual rule sets for each

*7th Workshop on Advances in Argumentation in Artificial Intelligence (AI³), November 06–09, 2023, Rome, Italy*
*Corresponding author.
giovanni.buraglio@tuwien.ac.at (G. Buraglio); wolfgang.dvorak@tuwien.ac.at (W. Dvořák);
a.rapberger@imperial.ac.uk (A. Rapberger); stefan.woltran@tuwien.ac.at (S. Woltran)

agent. Another example that requires the separation of rules is the necessity to express a qualitative distinction between them. In order to account for such situations, several argumentation formalisms such as ASPIC+ separate strict and defeasible inference rules [12, 13, 14, 15].

Let us consider the following illustrative example from the domain of normative reasoning.

**Example 1** (adapted from [16, Example 3])**.** *Our protagonist Alice has been accepted to a study program with payment obligations. Every student whose application has been accepted counts as eligible student (constitutive norm). Moreover, every eligible student must pay their tuition fee (regulative norm) and every student who pays their tuition fee counts as a self-funding student (constitutive norm). We can derive that Alice must pay her tuition fee (since she is an eligible student), and hence she counts as a self-funding student.*

*However, Alice has furthermore received a study grant which means that she is not a self-funding student after all. Hence we derive a counter-intuitive conflict, deducing Alice to be both self-funding and have received a grant.*

In the above example, we end up fallaciously deducing a contradiction from our assumptions. The underlying issue is that the application of constitutive rules after regulative ones may produce fallacious conclusions, called institutional wishful thinking. The undesired situation in Example 1 could be circumvented by preventing the application of the rule *"tuition fee $\rightarrow$ self-funding student"* after the rule *"eligible student $\rightarrow$ tuition fee"*. In the context of formal argumentation, similar issues have been addressed in recent works, based on an ASPIC-like formalism [17, 16, 18, 19]. Standard ABA is, however, not expressive enough to account for such a qualitative distinction among inference rules. Consequently, it is not possible to constraint the rule combinations on the basis of their kind.

In this work, we propose first steps in order to close this gap. In particular, we (a) extend the ABA formalism with pairwise disjoint sets of rules in order to take into account qualitative differences among them; (b) equip this extension of ABA with formal constraints (called *derivation graphs*) that regulate its deductive machinery, by encoding applicability conditions for inference rules; (c) investigate the role of derivation constraints within the argument construction process. On the one hand, we examine the definition of constraints as pre-processing operations on the underlying knowledge base, on the other hand, we present a prototype encoding of our formalism in Answer Set Programming (ASP). Finally, we point out to the relation between the possibility of expressing conflicts in normative reasoning and the expressive power of non-flat ABA.

## 2. Background

In order to introduce our formalism, we first need to recall some preliminaries for assumption-based argumentation. In ABA, frameworks representing debates are built up from a rule-based knowledge base and defined in the following way:

**Definition 1.** *An ABA framework (ABAF) is a tuple $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\phantom{a}})$ where: (i) $\mathcal{L}$ and $\mathcal{R}$ form together a deductive system and are respectively a set of atomic sentences in a language and a set of inference rules; (ii) $\mathcal{A} \subseteq \mathcal{L}$ is a non-empty set of atoms called assumptions; (iii) $\overline{\phantom{a}}$ is a total mapping from $\mathcal{A}$ into $\mathcal{L}$, where $\overline{a}$ is said to be the contrary of $a$, for each $a \in \mathcal{A}$.*

Following [20], we write rules as $r : \phi \leftarrow \phi_1, \ldots, \phi_m$ and we say that $\phi$ is the head of the rule and $\{\phi_1, \ldots, \phi_m\}$ is its body, formally $head(r) = \phi$ and $body(r) = \{\phi_1, \ldots, \phi_m\}$. For a set of rules $R$, we use $head(R)$ to indicate the set of atoms which are head of the rules contained in it. We consider here the finite flat version of ABAF, i.e. frameworks where $\mathcal{L}$ and $\mathcal{R}$ are finite and assumptions do not occur as conclusions of inference rules: there is no $r \in \mathcal{R}$ and $a \in \mathcal{A}$ for which $a = head(r)$. Arguments of an ABAF are based on proof-trees, constructed by forward-derivation from leaf-nodes to the root:

**Definition 2** (deduction)**.** *Let $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\phantom{a}})$ be an ABAF. A deduction for $p \in \mathcal{L}$ supported by $S \subseteq \mathcal{A}$ and $R \subseteq \mathcal{R}$, denoted $S \vdash^R p$ (or simply $S \vdash p$), is a finite rooted tree t with:*

*i) a labelling function that assigns each vertex of $t$ an element from $\mathcal{L} \cup \{\top\}$ such that the root is labelled by $p$ and leaves are labelled by either $\top$ or atoms in $S$;*
*ii) a surjective mapping $m$ from the set of internal nodes of $t$ onto rules $R$ satisfying, for each vertex $v$, that the label of $v$ is the head of the rule $m(v)$ and the children of $v$ are (one-to-one) labelled with the elements of the body of $m(v)$.*

In ABA, the attack relation is defined over sets of assumptions.

**Definition 3** (attack)**.** *Let $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\phantom{a}})$ be an ABAF, let $S, T \subseteq \mathcal{A}$ be two sets of assumptions. $S$ attacks $T$ ($S \rightarrow T$) iff there is a set $S' \subseteq S$ such that $S' \vdash \overline{a}$ for some $a \in T$.*

Semantics can be defined then in the usual way.

**Definition 4** (semantics)**.** *Given ABAF $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\phantom{a}})$ and $S, T \in \mathcal{A}$. The set $S$ is conflict-free iff it does not attack itself; and admissible iff it is conflict-free and $T \rightarrow S$ implies $S \rightarrow T$. stable iff it is conflict-free and $a \in \mathcal{A} \setminus S$ implies $S \rightarrow \{a\}$; preferred iff it is a $\subseteq$-maximal admissible set. We write $S \in \sigma(D)$ with $\sigma \in \{cf, adm, stb, prf\}$ to say that $S$ is a conflict-free, admissible, stable or preferred set of assumptions (or extension) of $D$.*

Likewise, we can define the corresponding AF for a given constrained ABAF.

**Definition 5.** *Given an ABAF $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\phantom{a}})$, we call $F = (\mathcal{A}, \mathcal{R})$ its corresponding AF such that:*

- $\mathcal{A} = \{S \vdash p \mid S \subseteq \mathcal{A}\}$;
- $\mathcal{R} = \{(S \vdash p, T \vdash q) \mid p = \overline{a}$ *for some* $a \in T\} \subseteq \mathcal{A} \times \mathcal{A}$.

## 3. ABA Frameworks with Multiple Rule-Sets

We can now define ABAFs with multiple rule-sets and derivation graphs. Jointly, these enable to trace rule kinds along with some constraint on their combination. We consider only frameworks where rule-sets are pairwise disjoint. Further, it is often desirable to evaluate scenarios where the same atom cannot be derived by rules of different kinds. Take for instance a legal debate built up using constitutive and regulative rules, as the one described in Example 1. The very difference between the two type of rules concern their output (i.e. their heads): constitutive rules produce institutional facts whereas regulative rules produce deontic statements such as obligations or permissions. It is therefore an intuitive requirement to separate these two heterogeneous groups of statements. For this, we focus on the class of *separated* $n$-ABAFs, for which heads of rules in different rule-sets are pairwise disjoint.

**Definition 6** ($n$-rule-sets ABA). *A* $n$-rule-sets ABAF *($n$-ABAF) is a tuple* $D = (\mathcal{L}, \{\mathcal{R}_i \mid 1 \leq i \leq n\}, \mathcal{A}, \overline{\phantom{x}})$ *such that* $(\mathcal{L}, \bigcup_{i=1}^{n} \mathcal{R}_i, \mathcal{A}, \overline{\phantom{x}})$ *is an ABAF. Moreover, we call* $D$ *separated whenever* $head(\mathcal{R}_i) \cap head(\mathcal{R}_j) = \emptyset$ *for all* $i, j$ *with* $i \neq j$.

As mentioned earlier, one might want to represent some constraint on rules combinations on $\bigcup_{i \leq n} \mathcal{R}_i$ depending on the particular application domain. Inspired by input/output combinations presented in [17], we introduce the more expressive concept of derivation graph to formalise combination constraints:

**Definition 7** (derivation graph). *Let* $D = (\mathcal{L}, \{\mathcal{R}_i \mid 1 \leq i \leq n\}, \mathcal{A}, \overline{\phantom{x}})$ *be an $n$-ABAF. A* derivation graph $G = (V, E)$ *for* $D$ *is a directed graph with* $|V| \geq n + 1$ *such that* $V$ *contains:*

*i) a distinct vertex* $\mathsf{s}$ *(called "starting node") with no incoming edges;*
*ii) at least one vertex* $\mathsf{r}_i$ *for each* $\mathcal{R}_i$ *(called "rule-node" for* $\mathcal{R}_i$*) such that there is a surjective mapping from rule-nodes onto the set of rule-sets* $\{\mathcal{R}_i \mid 1 \leq i \leq n\}$.

The outcome of the constraint encoded by some derivation graph is a limitation on the possibility of rules chaining. This affects the derivation process from the underlying deductive system. In particular, the idea consists in allowing only those sequential combinations of rules for which there is a path within the derivation graph. As a result, we extend the usual notion of deduction presented in ABA to accommodate this additional requirement.

**Definition 8.** *Let* $D = (\mathcal{L}, \{\mathcal{R}_i \mid 1 \leq i \leq n\}, \mathcal{A}, \overline{\phantom{x}})$ *be an $n$-ABAF and let* $G = (V, E)$ *be a derivation graph for* $D$. *A* $G$*-deduction for* $p \in \mathcal{L}$ *supported by* $S \subseteq \mathcal{A}$ *and* $R = \bigcup_{i=1}^{n} R_i$ *with* $R_i \subseteq \mathcal{R}_i$*, denoted* $S \vdash_G^R p$ *(or simply* $S \vdash_G p$*), is a deduction* $t$ *with a surjective mapping that maps every internal* $v$ *node of* $t$ *to a rule node* $w$ *in* $G$ *such that i)* $v$ *corresponds to a rule that is in the rule set of* $w$ *and ii) for each leaf-to-root-path* $v_0 \dots v_k$ *in* $t$*, the corresponding series of nodes* $w_0, \dots, w_k$ *in* $G$ *form a path in* $G$ *with* $w_0 = \mathsf{s}$.

Notions of $G$-attack, $G$-semantics and corresponding AF under $G$ can be easily adapted from the standard ones by employing the notion of $G$-deduction instead of regular deduction.

To show our new adaption at work, let us revisit our introductory example.

**Example 2.** *Consider again Example 1. We construct a* 2*-ABAF* $D = (\mathcal{L}, \mathcal{R}_1, \mathcal{R}_2, \mathcal{A}, \overline{\phantom{x}})$ *where* $\mathcal{R}_1$ *and* $\mathcal{R}_2$ *contain our constitutive and regulative rules, respectively. We assume the language* $\mathcal{L}$ *to contain a modality* $\mathcal{O}$ *where* $\mathcal{O}p$ *stands for "it is obligatory that* $p$*". We let* $\mathcal{L} = \{a := accepted\_application(x),\ b := received\_grant(x), p := eligible\_student(x), q := \mathcal{O}(pay\_fee(x)), s := self\_funding\_student(x)\}$*;* $\mathcal{A} = \{a, b\}$*;* $\mathcal{R}_1 = \{p \leftarrow a, s \leftarrow q\}$ *and* $\mathcal{R}_2 = \{q \leftarrow p\}$*. Moreover,* $s = \overline{b}$*. One can build the following conflicting derivations:*

$$\{b\} \vdash^{\{\}} b \qquad \{a\} \vdash^{\{p \leftarrow a, q \leftarrow p, \overline{b} \leftarrow q\}} \overline{b}$$

*We conclude that $\{a\}$ attacks $\{b\}$. However, there should not be any conflict between a student being accepted and receiving a grant. By prohibiting the application of constitutive rules in the scope of regulative ones, $\bar{b}$ is no longer derived from the assumption $a$. Let us consider the following derivation graph $G$:*

$$\begin{tikzcd} \mathsf{s} \arrow[r] \arrow[rr, bend left] & \boxed{\mathsf{r}_1} \arrow[r] & \boxed{\mathsf{r}_2} \end{tikzcd}$$

*Since $(\mathsf{r}_2, \mathsf{r}_1) \notin E(G)$, we get $\{a\} \nvdash_G^{\{p \leftarrow a, q \leftarrow p, \bar{b} \leftarrow q\}} \bar{b}$. Therefore, $\{a\}$ does not $G$-attack $\{b\}$ and the assumption set $\{a, b\}$ is an extension under any $G$-semantics.*

## 4. Investigating Constraints in ABA

In the present section, we examine the role of derivation constraints in the $n$-ABA formalism. In doing so, we take three different paths: first, we show that under certain conditions it is possible to exploit the information given by derivation graphs to pre-process the knowledge base in order to obtain equivalent results in terms of ABAFs instantiation; Then, we present an encoding for our formalism in Answer Set Programming that captures derivation constraints and their effect on the procedure of argument construction; Finally, we devote a paragraph to show how our formalism hints towards a possible relationship among non-flat ABA and certain instances of normative reasoning.

### 4.1. Equivalence under Derivation Function

A core feature of ABA is that it comes with guidelines that specify how to instantiate a framework from a given knowledge base. This job is largely done by the notions of deduction and attack. In turn, derivation graphs work as a device for controlling and manipulate such instantiating process. An interesting research question would be that of asking under which conditions one can obtain an equivalent framework by pre-processing the knowledge base while leaving the derivation process untouched. Initial results show that if a derivation graph contains exactly one rule node for each rule set in the given $n$-ABAF, it is possible to define a *derivation function* that works in such a way. This is an operation on the knowledge base which automatically identifies and removes rules that would not be allowed under a derivation graph $G$, allowing unconstrained deductions. For each graph constraint $G$, there is a derivation function $\gamma$ that extracts from the rules $\mathcal{R}$ of an $n$-ABAF the subset of rules whose application is allowed under $G$.

**Definition 9.** *Let $D$ be an $n$-ABAF, $G = (V, E)$ be some derivation graph with $V = \{\mathsf{s}, \mathsf{r}_1, \ldots, \mathsf{r}_n\}$. The* derivation function *$\gamma_G : 2^{\mathcal{R}} \to 2^{\mathcal{R}}$ corresponding to $G$ is defined as $\gamma(\mathcal{R}) = \mathcal{R} \setminus \{A \cup B\}$ with*

- $A = \{r \in \mathcal{R} \mid r \in \mathcal{R}_i, \mathcal{A} \cap body(r) \neq \emptyset$ *or* $body(r) = \emptyset$ *and* $(\mathsf{s}, \mathsf{r}_i) \notin E\}$;
- $B = \{r \in \mathcal{R} \mid r \in \mathcal{R}_i, \exists j \leq n$ *s.t.* $head(\mathcal{R}_j) \cap body(r) \neq \emptyset$ *and* $(\mathsf{r}_j, \mathsf{r}_i) \notin E\}$.

We omit subscript $G$ if clear from context.

Given some ABAF $D$, we use $D_\gamma$ to indicate the ABAF obtained by restricting rule sets of $D$ via the derivation function $\gamma$. As a result, the set of $G$-deductions for $D$ is equivalent to the set of standard deductions that can be built using rules in $\gamma(\mathcal{R})$ only.

**Lemma 1.** *Let $D$ be a separated $n$-ABA framework, $G = (V, E)$ some derivation-graph and $\gamma : \mathcal{R} \to \mathcal{R}$ its corresponding derivation function. For any set of atomic sentences $S \subseteq \mathcal{L}$, $p \in \mathcal{L}$ and $R \subseteq \mathcal{R}$, $S \vdash_G^R p$ is a $G$-deduction for $D$ if and only if $S \vdash^{\gamma(R)} p$ is a deduction for $D_\gamma$.*

*Proof.* ($\Rightarrow$) Assume $S \vdash_G^R p$ is a $G$-deduction for $D$. We show that $S \vdash^{\gamma(R)} p$ is a deduction for $D_\gamma$. In order to do this, we first show that $R = \gamma(R)$. We prove this by contradiction. Suppose there is an $r \in R$ such that $r \notin \gamma(R)$. Hence, either (a) $r \in A$ or (b) $r \in B$. We proceed by case distinction.

(a) Suppose $r \in A$. By Definition 9, the following holds true: (i) $r \in \mathcal{R}_i$ and (ii) $\mathcal{A} \cap body(r) \neq \emptyset$ or $body(r) = \emptyset$ and (iii) $(\mathsf{s}, \mathsf{r}_i) \notin E$, where $\mathsf{r}_i$ denotes the node in $G$ corresponding to the class $\mathcal{R}_i$.
By hypothesis, it holds that $S \vdash_G^R p$. By Definition 2, this is a finite rooted tree $t$ that assigns to each vertex an element in $\mathcal{L} \cup \top$. $t$ is constructed in such a way that there is a surjective mapping from rules in $R$ onto the nodes in $t$ such that each of these nodes and their children respectively correspond to the head and the elements in the body of a rule of $R$. Since $r \in R$, there is a node $v$ in the deduction tree $t$ that corresponds to $r$ and is labelled with $head(r)$ and its children are labelled with the elements in $body(r)$ or $\top$.
We show that (i)-(iii) cannot be true at the same time. We do this for each disjunct in (ii). First, consider $\mathcal{A} \cap body(r) \neq \emptyset$. In this case, there is at least one node $v'$, which is a child of $v$, labelled with an assumption. Hence, it is a leaf (since we assume $D$ to be flat). Consider now the leaf-to-root path $v_0 \dots v_k$ with $v_0 = v'$ and $v_1 = v$. By Definition 8, this path can be mapped to a path $w_0 \dots w_k$ in $G$ such that $w_0 = s$. Given that $r \in \mathcal{R}_i$ and $r$ corresponds to $v_1$, we know that $w_1$ corresponds to the class $\mathcal{R}_i$, i.e. $w_1 = \mathsf{r}_i$. Hence, we have shown that $(s, \mathsf{r}_i) \in E$, in contradiction with the assumption (iii): $(\mathsf{s}, \mathsf{r}_i) \notin E$.
Next, assume $body(r) = \emptyset$. In this case, the only child $v'$ of $v$ is labelled with $\top$. Again, the corresponding node $v'$ of $\top$ is a leaf in $t$. Consider the leaf-to-root path $v_0 \dots v_k$ with $v_0 = v'$ and $v_1 = v$. By definition 8, this path can be mapped to a path $w_0 \dots w_k$ in $G$ such that $w_0 = s$ and $w_1 = \mathsf{r}_i$. Hence, $(s, \mathsf{r}_i) \in E$, contradiction to the assumption $(\mathsf{s}, \mathsf{r}_i) \notin E$. Consequently, $r \notin A$.

(b) Suppose $r \in B$. By Definition 9, this means that (i) $r \in \mathcal{R}_i$, (ii) $\exists j \leq n$ such that $head(\mathcal{R}_j) \cap body(r) \neq \emptyset$ and (iii)$(\mathsf{r}_j, \mathsf{r}_i) \notin E$. We show that (i)-(iii) cannot be true at the same time.
Given that D is separated, there is no atom $h \in \mathcal{L}$ such that $h \in head(\mathcal{R}_k) \cap head(\mathcal{R}_j)$ with $k \neq j$. Hence, for every atom in $body(r)$ there exists at most one $j \leq n$ such that $head(\mathcal{R}_j) \cap body(r) \neq \emptyset$.
By hypothesis, it holds that $S \vdash_G^R p$. By Definition 2, this is a finite rooted tree, denoted as $t$, where each vertex is associated with an element from $\mathcal{L} \cup \top$. The construction of tree $t$ ensures a surjective mapping from rules in $R$ onto its nodes. In $t$ each node and its children respectively correspond to the head and body elements of a rule in $R$. For a given rule $r \in R$, there exists a node $v$ in the deduction tree $t$ that corresponds to $r$. This

node is labelled with $head(r)$, and its children are labelled with elements from $body(r)$ or $\top$.

Thus, there is at least one node $v'$ which is a child of $v$, labelled with $head(r')$ for some $r' \in \mathcal{R}_j$. Consider now the leaf-to-root path $v_0 \dots v_k$ with $v_m = v'$ and $v_{m+1} = v$ (with $1 < m < k$). By Definition 8, this path can be mapped to a path $w_0 \dots w_k$ in $G$ such that $w_0 = s$. Moreover, $r \in \mathcal{R}_i$. Since $r$ and $r'$ correspond to $v_m$ and $v_{m+1}$, the nodes $w_m$ and $w_{m+1}$ in $G$ correspond to the classes $\mathcal{R}_j$ and $\mathcal{R}_i$. Thus, $w_m = \mathsf{r}_j$ and $w_{m+1} = \mathsf{r}_i$. Hence, we have shown that $(\mathsf{r}_j, \mathsf{r}_i) \in E$, in contradiction with the assumption (iii): $(\mathsf{r}_\mathsf{j}, \mathsf{r}_i) \notin E$. Consequently, $r \notin B$.

Since there is a rule $r \in R$ such that $r \notin A$ and $r \notin B$, we conclude that $r \in \gamma(R)$, in contradiction with our initial assumption. We derive $R = \gamma(R)$. Thus we replace $R$ with $\gamma(R)$ in $S \vdash_G^R p$, deriving that $S \vdash_G^{\gamma(R)} p$ is a $G$-deduction in $D_\gamma$. Hence, $S \vdash^{\gamma(R)} p$ is a deduction in $D_\gamma$.

($\Leftarrow$) Suppose $S \vdash^{\gamma(R)} p$ is a deduction in $D_\gamma$. We show $S \vdash_G^R p$ is a $G$-deduction in $D$. Since $\gamma(\mathcal{R}) \subseteq \mathcal{R}$, we know that each rule in $\gamma(R)$ is contained in $D$, i.e. we can use each rule in $\gamma(R)$ under $G$. Hence, we can assume that $R = \gamma(R)$. We can thus replace $\gamma(R)$ with $R$ in $S \vdash^{\gamma(R)} p$, thus showing that it is a deduction for $D_\gamma$ (and *a fortiori* for $D$). It remains to show that $S \vdash^R p$ is a $G$-deduction for $D$.

By Definition 2, there is a finite tree $t$ rooted in $p$ with leaves corresponding to assumptions $S \cup \{\top\}$ and each node $v_i$ is associated to a rule $r \in R$. To prove further that $S \vdash^R p$ is a $G$-deduction, we need to show that it is possible to map each leaf-to-root path $v_0 \dots v_k$ in $t$ to a path $w_0 \dots w_k$ in the graph $G$ (condition (ii) of Definition 8). Take one leaf-to-root-path $v_0 \dots v_k$ in $t$ such that each node $v_m$ (corresponding to $w_m$) is associated with a rule $r \in \mathcal{R}_i$.

Assume that there is no path $w_0 \dots w_k$ in $G$ corresponding to $v_0 \dots v_k$. This means that there is at least one edge $(v_m, v_{m+1})$ in $t$ for which the corresponding $(w_m, w_{m+1}) \notin E$. To prove that this always leads to a contradiction, we distinguish the following two cases.

(a) First suppose $m = 0$, that is, $(w_0, w_1) \notin E$. Let $r$ denote the rule corresponding to $v_1$. Since $r \in \gamma(R)$ by hypothesis, $r$ is not deleted by $\gamma$. By Definition 9 we have that $r \notin A$, that is $\mathcal{A} \cap body(r) = \emptyset$ and $body(r) \neq \emptyset$. However, by Definition 2, it holds that each leaf is labelled with $S \cup \{\top\}$. Hence, $v_0$ corresponds either to an assumption or $\top$. In the first case, $\mathcal{A} \cap body(r) \neq \emptyset$. In the second case, $body(r) = \emptyset$. Both cases contradict the assumption that $r \in \gamma(R)$. Hence we obtain $(w_0, w_1) \in E$.

(b) Now suppose $m > 0$ and $(w_m, w_{m+1}) \notin E$. Let $r'$ and $r$ denote the rules respectively corresponding to $v_m$ and $v_{m+1}$. Since $r$ is not deleted by $\gamma$ ($r \in \gamma(R)$), by Definition 9 we have that $r \notin B$, that is for all $j \leq n$ it holds that $head(\mathcal{R}_j) \cap body(r) = \emptyset$.
By Definition 2, there is a surjective mapping from the set of internal nodes of $t$ onto rules $R$ satisfying, for each vertex $v$, that the label of $v$ is the head of the rule corresponding to $v$ and the children of $v$ are (one-to-one) labelled with the elements of the body of the rule. Therefore, $v_{m+1}$ is labelled with the head of the rule $r$ and $v_m$, being a child of $v_{m+1}$, labelled with the head of rule $r'$ which is is an element in the body of $r$. Let $\mathcal{R}_j$ denote the rule set corresponding to $r'$. Then $head(\mathcal{R}_j) \cap body(r) \neq \emptyset$. Therefore, $r \in B$, that is, $r$ is deleted in $D_\gamma$. But this is in contradiction with the assumption that $r \in \gamma(R)$. Hence we obtain $(w_m, w_{m+1}) \in E$.

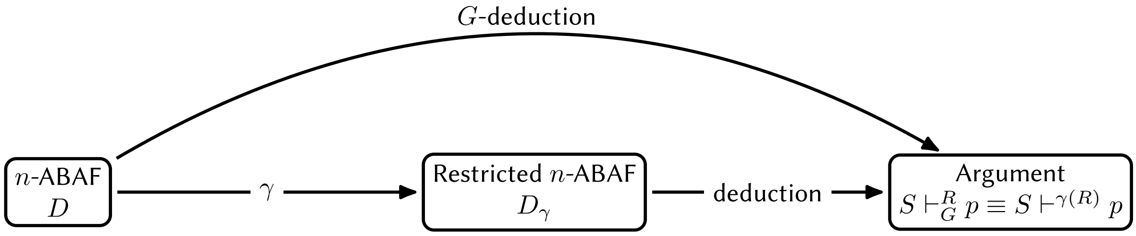



**Figure 1:** Graphical representation of Lemma 1.

These results contradict our previous assumption that there is no path $w_0 \dots w_k$ in $G$ corresponding to some fixed $v_0 \dots v_k$ in $t$. Hence, a path in $G$ corresponding to $v_0 \dots v_k$ in $t$ is always found. We conclude that $S \vdash^R p$ is a $G$-deduction. $\square$

A graphical representation of the equivalence between $G$-deductions and deduction in $D_\gamma$ is given in Figure 1.

To see how the translation works from $G$-deduction of $D$ to regular deductions in $D_\gamma$, consider the following example.

**Example 3.** *Let $D = (\mathcal{L}, \mathcal{R}_1, \mathcal{R}_2, \mathcal{A}, \overline{\phantom{x}})$ be a 2-ABAF with $\mathcal{L} = \{a, b, p, q, s\}$, $\mathcal{R}_1 = \{p \leftarrow a\}$, $\mathcal{R}_2 = \{q \leftarrow p, s \leftarrow b\}$ and $\mathcal{A} = \{a, b\}$. Let $G = (V, E)$ be a derivation graph with $V = \{\mathsf{s}, \mathsf{r}_1, \mathsf{r}_2\}$ and $E = \{(\mathsf{s}, \mathsf{r}_1), (\mathsf{r}_1, \mathsf{r}_2)\}$ as follows:*

$$\mathsf{s} \longrightarrow \boxed{\mathsf{r}_1} \longrightarrow \boxed{\mathsf{r}_2}$$

*Under $G$, the set of $G$-deduction that can (and cannot) be built are the following:*

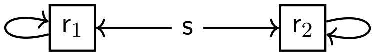
$$\{a\} \vdash_G a, \qquad \{b\} \vdash_G b \qquad \{a\} \vdash_G^{\{p \leftarrow a\}} p \qquad \{a\} \vdash_G^{\{p \leftarrow a, q \leftarrow p\}} q, \quad \textit{but} \qquad \{b\} \nvdash_G^{\{s \leftarrow b\}} s$$

*Let us now take the corresponding derivation function $\gamma$ for $G$. By definition 9, we have $\gamma(\mathcal{R}) = \mathcal{R} \setminus \{A \cup B\}$, where $A = \{s \leftarrow b\}$ and $B = \emptyset$ such that $D_\gamma = (\mathcal{L}, \gamma(\mathcal{R}), \mathcal{A}, \overline{\phantom{x}})$. Eventually, for $D_\gamma$ we obtain the following:*

$$\{a\} \vdash a, \qquad \{b\} \vdash b \qquad \{a\} \vdash^{\{p \leftarrow a\}} p \qquad \{a\} \vdash^{\{p \leftarrow a, q \leftarrow p\}} q, \quad \textit{but} \qquad \{b\} \nvdash^{\{s \leftarrow b\}} s$$

*As it can be seen, every $G$-deduction for $D$ is also a deduction for $D_\gamma$, and viceversa.*

**Remark 1.** *In Lemma 1, we require that the $n$-ABAF $D$ is separated, that is $head(\mathcal{R}_i) \cap head(\mathcal{R}_j) = \emptyset$ for all $i, j$ with $i \neq j$. The motivation behind this choice lies in the fact that the derivation function could in some occasions restrict the rule-set causing the set of deductions for $D_\gamma$ to be a subset of the set of $G$-deductions for $D$. To show this, let us consider the following example: take $D$ such that $\mathcal{L} = \{a, b, p, q\}$, $\mathcal{R}_1 = \{p \leftarrow a, q \leftarrow p\}$, $\mathcal{R}_2 = \{p \leftarrow b\}$ and $\mathcal{A} = \{a, b\}$. $D$ is not separated due to the fact that $p \in head(\mathcal{R}_1) \cap head(\mathcal{R}_2)$. Moreover, let $G = (V, E)$ be the following derivation graph:*




*As it can be seen easily, the rule $r : q \leftarrow p$ would be eliminated by the function $\gamma$ since $r \in B$. Indeed, we have $p \in head(\mathcal{R}_2) \cap body(r)$, $r \in \mathcal{R}_1$ and $(\mathsf{r}_2, \mathsf{r}_1) \notin E$. Thus, $\{a\} \vdash^{\gamma(\mathcal{R})} q$ is not a deduction for $D_\gamma$. However, we would at the same time allow the rule $r$ under $G$ since $(\mathsf{s}, \mathsf{r}_1) \in E$ and $(\mathsf{r}_1, \mathsf{r}_1) \in E$, so that $\{a\} \vdash^{\mathcal{R}}_{G} q$ is a $G$-deduction for $D$. In order to avoid such undesired behaviour, we restrict our study to separated ABAFs.*

From Lemma 1 it follows that the corresponding AF instantiated by means of $G$-deductions is equivalent to the one instantiated through standard deductions after its rule-set has been restricted by the derivation function. This assures that the same outcome is reached by limiting deductions via some derivation graph or by restricting the knowledge base accordingly. This is captured by the following theorem:

**Theorem 1** (Equivalence under instantiation)**.** *Let $D$ be a separated $n$-ABA framework, $G$ some derivation graph and $\gamma : \mathcal{R} \rightarrow \mathcal{R}$ some derivation function. Let $F_G = (\mathcal{A}_G, \mathcal{R}_G)$ be the corresponding AF with respect to $D$ under $G$ and $F' = (\mathcal{A}', \mathcal{R}')$ the corresponding AF with respect to $D_\gamma$. For these, we derive that $F_G \equiv F'$, in the sense that:*

*(1) $\mathcal{A}_G = \mathcal{A}'$;*
*(2) $\mathcal{R}_G = \mathcal{R}'$.*

*Proof.* We start by considering (1). By Definition 5, we have $\mathcal{A}_G = \{S \vdash^{R}_{G} p \mid S \subseteq \mathcal{A}\}$. Given Lemma 1, we know that for each argument in such set, there is an equivalent argument that can be obtained through some derivation function $\gamma$ such that $\{S \vdash^{\gamma(R)} p \mid S \subseteq \mathcal{A}\} = \mathcal{A}'$. Hence, $\mathcal{A}_G = \mathcal{A}'$.

The proof of (2) is similar. By Definition 5, we have $\mathcal{R}_G = \{(S \vdash^{R}_{G} p, T \vdash^{R^\star}_{G} q) \mid p = \overline{a} \text{ for some } a \in T\}$. Given Lemma 1, we know that for each pair of arguments in such set, there is an equivalent pair of arguments that can be obtained through some derivation function $\gamma$ such that $S \vdash^{R}_{G} p \iff S \vdash^{\gamma(R)} p$ and $T \vdash^{R^\star}_{G} q \iff T \vdash^{\gamma(R^\star)} q$. Since these arguments are pairwise equivalent, the attack relation among them will be equivalent as well. Thus, we obtain $\{(S \vdash^{\gamma(R)} p, T \vdash^{\gamma(R^\star)} q) \mid p = \overline{a} \text{ for some } a \in T\} = \mathcal{R}'$ as the set of attacks for $F'$. Finally, we can state $\mathcal{R}_G = \mathcal{R}'$, as desired. □

A straightforward consequence of this is that semantics equivalence also holds:

**Corollary 1** (Equivalence)**.** *Let $D$ be a separated $n$-ABAF, $D_\gamma$ its restriction under $\gamma$ and $G$ the corresponding derivation graph. Given any ABA semantics $\sigma \in \{cf, adm, stb, prf\}$ and their constrained version $\sigma_G$, it holds that $\sigma_G(D) = \sigma(D_\gamma)$.*

## 4.2. Encoding constrained $n$-ABA in ASP

In addition, we present an encoding of the $n$-ABA formalism and derivation graphs in ASP (available at: https://www.dbai.tuwien.ac.at/research/argumentation/abasp/), inspired by the one provided in [21] for regular ABA frameworks and semantics. Given an $n$-ABAF and derivation graph as input, the encoding provides an answer set $M$ for each $\sigma_G$-extension of a given $n$-ABAF under the graph constraint $G$. Regarding the $n$-ABAF in input, we extend the encoding presented in [21] by introducing a new predicate specifying for each rule the (unique) rule-set $\mathcal{R}_i$ it belongs to. Let $D = (\mathcal{L}, \{\mathcal{R}_i \mid 1 \leq i \leq n\}, \mathcal{A}, \overline{\phantom{a}})$ be an $n$-ABAF with $\mathcal{R}_i$ be the $i$-th set of rules. We use the following set of facts in ASP to represent the $n$-ABAF $D$:

$$
\begin{aligned}
\texttt{D} = &\{\mathbf{assumption}(a). \mid a \in \mathcal{A}\} \cup \\
&\{\mathbf{head}(m, b). \mid b \in head(r_m), r_m \in \mathcal{R}\} \cup \\
&\{\mathbf{body}(m, b). \mid b \in body(r_m), r_m \in \mathcal{R}\} \cup \\
&\{\mathbf{rule\_set}(r_m, rs_i). \mid r_m \in \mathcal{R}_i\} \cup \\
&\{\mathbf{contrary}(a, b). \mid b = \overline{a}, a \in \mathcal{A}\}.
\end{aligned}
$$

Following [21], $\mathbf{assumption}(a)$ and $\mathbf{contrary}(a, b)$ respectively mean that $a$ is an assumption and that $b$ is the contrary of $a$. Moreover, $\mathbf{head}(m, b)$ and $\mathbf{body}(m, b)$ mean that $b$ is the head (resp. body) of the rule $r_m$ within $\mathcal{R}$. In addition, we introduce the predicate $\mathbf{rule\_set}(r_m, rs_i)$ to specify that $r_m$ is in the rule-set $\mathcal{R}_i$.

The derivation graph $G$ is encoded as a labelled graph using predicates for nodes and edges, specifying which node corresponds to the starting node and each rule set.

$$
\begin{aligned}
\texttt{G} = &\{\mathbf{node}(v). \mid v \in V\} \cup \\
&\{\mathbf{edge}(v_1, v_2). \mid (v_1, v_2) \in E\} \cup \\
&\{\mathbf{start\_node}(v). \mid v \textit{ is the starting node}\} \cup \\
&\{\mathbf{rule\_node}(v, rs_i). \mid v \textit{ corresponds to } \mathcal{R}_i\}.
\end{aligned}
$$

We use the following ASP program $\pi_G$ that mirrors the argument construction process for an $n$-ABAF under $G$ (see Listing 1). To present this in a more concise way, we say that "a rule $R$ is in a node $N$" whenever such rule is contained in the rule-set corresponding to the node $N$ in the derivation graph.

Listing 1: Module $\pi_G$

$\mathbf{in}(X) \leftarrow \mathbf{assumption}(X), not\ \mathbf{out}(X).$

$\mathbf{out}(X) \leftarrow \mathbf{assumption}(X), not\ \mathbf{in}(X).$

$\mathbf{fact\_rule}(R, X) \leftarrow \mathbf{head}(R, X), not\ \mathbf{non\_fact\_rule}(R, X).$

$\mathbf{non\_fact\_rule}(R, X) \leftarrow \mathbf{head}(R, X), \mathbf{body}(R, Y).$

$\mathbf{supported\_by\_node}(X, N) \leftarrow \mathbf{in}(X), \mathbf{start\_node}(N).$

$\mathbf{supported\_by\_node}(X, N) \leftarrow \mathbf{head}(R, X), \mathbf{rule\_set}(R, I), \mathbf{rule\_node}(N, I), not\ \mathbf{non\_supported\_by\_node\_via\_rule}(X, N, R).$

$\mathbf{non\_supported\_by\_node\_via\_rule}(X, N, R) \leftarrow \mathbf{fact\_rule}(R, X), \mathbf{rule\_set}(R, I), \mathbf{rule\_node}(N, I), \mathbf{start\_node}(M), not\ \mathbf{edge}(M, N).$

$\mathbf{non\_supported\_by\_node\_via\_rule}(X, N, R) \leftarrow \mathbf{head}(R, X), \mathbf{rule\_set}(R, I), \mathbf{rule\_node}(N, I), \mathbf{non\_supported\_by\_node\_via\_rule\_because\_of}(X, N, R, Y), \mathbf{body}(R, Y).$

$\mathbf{non\_supported\_by\_node\_via\_rule\_because\_of}(X, N, R, Y) \leftarrow \mathbf{head}(R, X), \mathbf{rule\_set}(R, I), \mathbf{rule\_node}(N, I), \mathbf{body}(R, Y), \{\mathbf{supported\_by\_node}(Y, M) : \mathbf{edge}(M, N)\} = 0.$

$\leftarrow \mathbf{rule\_set}(R, I), \mathbf{rule\_set}(R, J), I != J.$

Lines 1 and 2 encode a guess of some possible extension in the set of assumptions, specifying which of them are taken to be **in** and **out** respectively. We label facts via the predicate **fact_rule**, telling them apart from rules with non-empty body (Lines 4 and 5). Lines 5-9 encode the construction process of $G$-deductions as forward derivations from subsets of $\mathcal{A}$ to supported claims. These establish the connection between nodes of a derivation graph, rules and supported atoms, represented by the predicate **supported_by_node**. As for Line 5, the set of assumptions $A \subseteq \mathcal{A}$ that is guessed to be **in** is set to be supported by the starting node of the derivation graph. Subsequently, as in [21], for any atom $p$ that can be $G$-deduced from (a subset of) $A$, we obtain **supported_by_node**$(x)$ in some answer set. In particular, Line 9 says that a rule in a node $N$ that might fire is blocked when its body is not supported by any node from an incoming edge $(M, N)$. Combinations of rules which are not allowed under the derivation graph in input are thus ruled out. Further, Line 8 enforces that *all* elements in the body of a rule in some rule-node have to be supported for its head to be supported as well. Thus, it prevents rules with partially supported body to fire, even when the derivation graph would allow for such combination. Line 7 blocks facts in a node $N$ to be derived when there is no incoming edge from the starting node. Finally, Line 6 makes use of the concept of negation as failure to establish that an atom $p$ is supported by a node $N$ if it occurs in the head of some rule in $N$ and no rule can be found in $N$ for which $p$ is **non_supported**. A constraint in Line 10 checks that the same rule is not contained in two different rule sets, encoding the requirement that rule sets are pairwise disjoint.

Eventually, each semantics-related module presented in [21] can be integrated into ours, after being carefully adapted to take into account rule-sets and derivation constraints.

### 4.3. Non-flat ABA for Normative Reasoning

In the area of normative reasoning, conflicts may occur not only in presence of inconsistent information regarding brute facts, but also regarding institutional facts and norms[16]. Scenarios that concern the detachment of conflicting institutional facts are called *normative conflicts* and arise when more agents agree on the same brute fact, but assign conflicting institutional values to it. For example, an homosexual couple counts as married after having signed the marriage contract in some legal systems, but not in others. Similarly, conflicts among obligations can arise, giving rise to so-called *moral dilemmas.* Instances of these arise in presence of an obligation for $p$ and for its opposite (formally, this translates to deriving $\mathcal{O}p \wedge \mathcal{O}\neg p$ from assumptions).

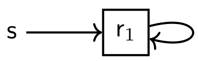
In assumption-based argumentation, conflicts between sentences are encoded by the contrary function over the assumption set. This represent a fundamental design property of ABA, because it allows in turn to define semantics directly on the assumption level. For this reason, in order to express normative conflicts and moral dilemmas, it is required that assumptions may not only consist of so-called brute facts, but also of institutional facts (produced by constitutive rules) or obligations (produced by regulative rules). Since these are always derived by rules in our knowledge base, flat ABA may not always be expressive enough to encompass such cases.

Therefore, we anticipate here that the full expressiveness of non-flat ABA may be required for capturing instances of normative reasoning. In order to see this, let us consider an instance of Forrester's paradox [22]. In Standard Deontic Logic [23], Forrester's paradox, also known in the literature as "gentle murderer paradox", follows from the statements A: "One should not (under the law) commit murder" and B: “if someone commits murder, then they should do it gently". Moreover, B implies C: “if someone should commit murder gently, then they should commit murder". Under the assumption that D: “someone commits murder", this eventually creates a paradoxical situation whereby it is obligatory to commit and not to commit murder at the same time. Therefore, a moral dilemma is created where it is contradictory to assume that a law exists and someone violates it. In the following, we show that such an undesired outcome can be avoided by imposing some constraint by means of a derivation graph $G$.

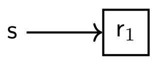
**Example 4.** *Take a non-flat* $1$*-ABAF* $D = (\mathcal{L}, \mathcal{R}_1, \mathcal{A}, \overline{\phantom{x}})$ *such that* $\mathcal{L} = \{a, b, p\}$, $\mathcal{A} = \{a, b\}$ *where* $a := murder(x)$ *and* $b := \mathcal{O}(\neg murder(x))$ *and* $\mathcal{R}_1 = \{p \leftarrow a, \overline{b} \leftarrow p, b \leftarrow\}$ *where* $p := \mathcal{O}(murder\_gently(x))$. *Then one could build the arguments:*

$$\{\} \vdash^{\{b\leftarrow\}} b \qquad \{a\} \vdash^{\{\}} a \qquad \{a\} \vdash^{\{p\leftarrow a\}} p \qquad \{a\} \vdash^{\{p\leftarrow a, \overline{b}\leftarrow p\}} \overline{b}$$

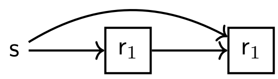
*As it can be seen immediately,* $\{a\}$ *attacks* $\{b\}$. *Hence the contrary-to-duty paradox: the assumptions that murdering is forbidden and that someone murders are mutually exclusive and their union is not conflict-free. For this example, not every derivation graph will prevent the paradox to arise. Consider the following:*




$G_1$




$G_2$




$G_3$

*$G_1$ puts no restriction on deductions, $G_2$ restricts deductions to using only one rule, and $G_3$ restricts deductions to using only two rules per branch. In our example both $G_1$ and $G_3$ do not prohibit any of the possible deductions, while $G_2$ does and in fact is the only graph which prevents the paradox. In fact, by forbidding the iteration for rules, it blocks the derivation of $\overline{b}$ from the assumption $a$. Indeed, we get $\{a\} \nvdash_G^{\{p\leftarrow a, \overline{b}\leftarrow p\}} \overline{b}$. Therefore, $\{a\}$ does not $G$-attack $\{b\}$ and the assumption set $\{a, b\}$ is an extension under any $G$-semantics.*

In this example, an instance of a contrary-to-duty paradox is easily formalised in non-flat 1-ABA and solved through some constraints imposed by a derivation graph. This suggest that an exhaustive analysis of conflict resolution in the context of normative reasoning requires the full expressiveness of non-flat $n$-ABA.

## 5. Concluding Remarks

This work introduces an extension of assumption-based argumentation with multiple rule-sets together with some formal constraints on its deductive machinery. These constraints, called *derivation graphs*, regulate the argument construction process from the underlying knowledge base, thereby limiting the procedure for its instantiation into an ABA framework. While this allows to avoid undesired conclusions as shown in Examples 1 and 4, we are currently working on defining constraints that operate directly on the knowledge base. In addition, we presented an encoding of our formalism in ASP, building up on the work presented in [21]. Finally, we discussed the possibility to capture certain instances of normative reasoning in assumption-based argumentation, using the full expressive power of non-flat ABA to represent and reason about normative conflicts and moral dilemmas.

The derivation constraints presented in this work successfully avoid some paradoxes and fallacies in the domain of normative reasoning, but they do so at the expense of the deductive power of the ABA formalism. As a general direction for future research we want to broaden our horizon and investigate different kinds of reasoning constraints that minimise this loss. In doing so, we aim at positioning our formalism with respect to related frameworks: the work by Pigozzi and Van der Torre on constitutive and regulative norms in argumentation [16]; modular ABA [24] as it was proposed in connection with normative reasoning; Deontic ASP [25] encoding input/output logics. Although we restricted our studies on flat ABAFs so far, we anticipate that the full expressiveness of non-flat ABAFs may be needed to capture general instances of normative reasoning (cf. Example 4). Equipping non-flat ABAFs with derivation graphs might pose additional challenges since non-flat ABAFs require certain closure conditions on the set of acceptable assumptions. In addition, we aim at studying how size and complexity of instantiated ABAFs are influenced under our derivation constraints, in line with [26].

## Acknowledgements

This research has been supported by Vienna Science and Technology Fund (WWTF) through project ICT19-065 and by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreements No. 101034440).

## References

[1] A. Bondarenko, F. Toni, R. A. Kowalski, An assumption-based framework for non-monotonic reasoning, in: L. M. Pereira, A. Nerode (Eds.), Logic Programming and Non-monotonic Reasoning, Proceedings of the Second International Workshop, Lisbon, Portugal, June 1993, MIT Press, 1993, pp. 171–189.

[2] A. Bondarenko, P. M. Dung, R. A. Kowalski, F. Toni, An abstract, argumentation-theoretic approach to default reasoning, Artif. Intell. 93 (1997) 63–101. URL: https://doi.org/10.1016/S0004-3702(97)00015-5. doi:10.1016/S0004-3702(97)00015-5.

[3] P. M. Dung, R. A. Kowalski, F. Toni, Assumption-based argumentation, in: G. R. Simari, I. Rahwan (Eds.), Argumentation in Artificial Intelligence, Springer, 2009, pp. 199–218. URL: https://doi.org/10.1007/978-0-387-98197-0_10. doi:10.1007/978-0-387-98197-0_10.

[4] K. Cyras, T. Oliveira, A. Karamlou, F. Toni, Assumption-based argumentation with preferences and goals for patient-centric reasoning with interacting clinical guidelines, Argument Comput. 12 (2021) 149–189. URL: https://doi.org/10.3233/AAC-200523. doi:10.3233/AAC-200523.

[5] X. Fan, F. Toni, A. Mocanu, M. Williams, Dialogical two-agent decision making with assumption-based argumentation, in: A. L. C. Bazzan, M. N. Huhns, A. Lomuscio, P. Scerri (Eds.), International conference on Autonomous Agents and Multi-Agent Systems, AAMAS

’14, Paris, France, May 5-9, 2014, IFAAMAS/ACM, 2014, pp. 533–540. URL: http://dl.acm.org/citation.cfm?id=2615818.

[6] X. Fan, On generating explainable plans with assumption-based argumentation, in: T. Miller, N. Oren, Y. Sakurai, I. Noda, B. T. R. Savarimuthu, T. C. Son (Eds.), PRIMA 2018: Principles and Practice of Multi-Agent Systems - 21st International Conference, Tokyo, Japan, October 29 - November 2, 2018, Proceedings, volume 11224 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 344–361. URL: https://doi.org/10.1007/978-3-030-03098-8_21. doi:10.1007/978-3-030-03098-8_21.

[7] S. Modgil, H. Prakken, Abstract rule-based argumentation, FLAP 4 (2017). URL: http://www.collegepublications.co.uk/downloads/ifcolog00017.pdf.

[8] J. R. Searle, Speech Acts: An Essay in the Philosophy of Language, Cambridge, England: Cambridge University Press, 1969.

[9] D. Makinson, L. W. N. van der Torre, Input/output logics, J. Philos. Log. 29 (2000) 383–408. URL: https://doi.org/10.1023/A:1004748624537. doi:10.1023/A:1004748624537.

[10] J. Xie, C.-C. Liu, Multi-agent systems and their applications, Journal of International Council on Electrical Engineering 7 (2017) 188–197.

[11] I. Rahwan, Guest editorial: Argumentation in multi-agent systems, Auton. Agents Multi Agent Syst. 11 (2005) 115–125. URL: https://doi.org/10.1007/s10458-005-3079-0. doi:10.1007/s10458-005-3079-0.

[12] H. Prakken, An abstract framework for argumentation with structured arguments, Argument Comput. 1 (2010) 93–124. URL: https://doi.org/10.1080/19462160903564592. doi:10.1080/19462160903564592.

[13] S. Modgil, H. Prakken, A general account of argumentation with preferences, Artif. Intell. 195 (2013) 361–397. URL: https://doi.org/10.1016/j.artint.2012.10.008. doi:10.1016/j.artint.2012.10.008.

[14] M. Caminada, S. Modgil, N. Oren, Preferences and unrestricted rebut, in: S. Parsons, N. Oren, C. Reed, F. Cerutti (Eds.), Computational Models of Argument - Proceedings of COMMA 2014, Atholl Palace Hotel, Scottish Highlands, UK, September 9-12, 2014, volume 266 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, 2014, pp. 209–220. URL: https://doi.org/10.3233/978-1-61499-436-7-209. doi:10.3233/978-1-61499-436-7-209.

[15] S. Modgil, H. Prakken, The *ASPIC*$^+$ framework for structured argumentation: a tutorial, Argument Comput. 5 (2014) 31–62. URL: https://doi.org/10.1080/19462166.2013.869766. doi:10.1080/19462166.2013.869766.

[16] G. Pigozzi, L. van der Torre, Arguing about constitutive and regulative norms, J. Appl. Non Class. Logics 28 (2018) 189–217. URL: https://doi.org/10.1080/11663081.2018.1487242. doi:10.1080/11663081.2018.1487242.

[17] X. Sun, L. W. N. van der Torre, Combining constitutive and regulative norms in input/output logic, in: F. Cariani, D. Grossi, J. Meheus, X. Parent (Eds.), Deontic Logic and Normative Systems - 12th International Conference, DEON 2014, Ghent, Belgium, July 12-15, 2014. Proceedings, volume 8554 of *Lecture Notes in Computer Science*, Springer, 2014, pp. 241–257. URL: https://doi.org/10.1007/978-3-319-08615-6_18. doi:10.1007/978-3-319-08615-6_18.

[18] B. Liao, M. Slavkovik, L. W. N. van der Torre, Building jiminy cricket: An architecture for moral agreements among stakeholders, CoRR abs/1812.04741 (2018). URL: http://arxiv.org/abs/1812.04741. arXiv:1812.04741.

[19] B. Liao, P. Pardo, M. Slavkovik, L. van der Torre, The jiminy advisor: Moral agreements among stakeholders based on norms and argumentation, J. Artif. Intell. Res. 77 (2023) 737–792. URL: https://doi.org/10.1613/jair.1.14368. doi:10.1613/jair.1.14368.

[20] K. Cyras, X. Fan, C. Schulz, F. Toni, Assumption-based argumentation: Disputes, explanations, preferences, FLAP 4 (2017). URL: http://www.collegepublications.co.uk/downloads/ifcolog00017.pdf.

[21] T. Lehtonen, J. P. Wallner, M. Järvisalo, Declarative algorithms and complexity results for assumption-based argumentation, J. Artif. Intell. Res. 71 (2021) 265–318. URL: https://doi.org/10.1613/jair.1.12479. doi:10.1613/jair.1.12479.

[22] J. W. Forrester, Gentle murder, or the adverbial samaritan, The Journal of Philosophy 81 (1984) 193–197.

[23] G. H. von Wright, Deontic logic, Mind 60 (1951) 1–15. doi:10.1093/mind/lx.237.1.

[24] P. M. Dung, P. M. Thang, Modular argumentation for modelling legal doctrines in common law of contract, Artif. Intell. Law 17 (2009) 167–182. URL: https://doi.org/10.1007/s10506-009-9076-x. doi:10.1007/s10506-009-9076-x.

[25] R. Gonçalves, J. J. Alferes, An embedding of input-output logic in deontic logic programs, in: T. Ågotnes, J. M. Broersen, D. Elgesem (Eds.), Deontic Logic in Computer Science - 11th International Conference, DEON 2012, Bergen, Norway, July 16-18, 2012. Proceedings, volume 7393 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 61–75. URL: https://doi.org/10.1007/978-3-642-31570-1_5. doi:10.1007/978-3-642-31570-1_5.

[26] A. Rapberger, M. Ulbricht, J. P. Wallner, Argumentation frameworks induced by assumption-based argumentation: Relating size and complexity, in: O. Arieli, G. Casini, L. Giordano (Eds.), Proceedings of the 20th International Workshop on Non-Monotonic Reasoning, NMR 2022, Part of the Federated Logic Conference (FLoC 2022), Haifa, Israel, August 7-9, 2022, volume 3197 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 92–103. URL: https://ceur-ws.org/Vol-3197/paper9.pdf.