REASON-BASED LOGIC:
a logic for reasoning with rules and reasons

Jaap Hage
Bart Verheij
University of Limburg
Department of Metajuridica
P.O. Box 616
6200 MD Maastricht
+31 43 88 3053 / 3048 / 3020
email: {jaap.hage, bart.verheij}@metajur.rulimburg.nl

ABSTRACT

The main claim of this paper is that reasoning with rules, especially rules of law, is different from reasoning with statements that are true or false. This difference is, amongst others, reflected in the defeasibility of arguments in which rules play a role. Reason-Based Logic is a logic that has special facilities for dealing with rules and with reasons based on rules. In particular it allows arguments in which conclusions are derived by 'weighing' the reasons that plead for and against them. In this article we illustrate some characteristics of reasoning with rules, and show how Reason-Based Logic deals with these characteristics. The article is concluded with some general considerations concerning Reason-Based Logic, and a comparison with some other logics for defeasible reasoning.

1.1. INTRODUCTION

Rules are different from statements, and a logic that deals with rules should take this difference seriously. This is maybe the most important message that this article contains. The rest might be considered as a mere elaboration of this claim. We will validate this claim and show its implications. In this introduction we focus on the application of rules to illustrate the difference between rules and statements. We close this introduction with an overview of the remainder of this article.

1.1. Rules need to be applied

Consider the following argument:

Thieves are punishable.
John is a thief.
Therefore: John is punishable.

This argument is a simplified version of how actual legal reasoning might run. We start with a rule, subsume some facts under it, and draw the legal consequences of the case as logical conclusions. If we reconstruct this argument in classical predicate logic, we obtain something like:

For all *x*, if *x* is a thief, then *x* is punishable.
John is a thief.
Therefore: John is punishable.

The rule of the first premise has become a universal material implication.

As long as the information given in the argument is all there is, nothing seems to go wrong with this logical reconstruction of the argument. Suppose, however, that the prosecution of John's theft is barred by lapse of time. In that case, the conclusion of the argument is false, and consequently - since the argument is deductively valid - one of the premises must be false too. The only premise that candidates for being false is the first one, the statement that everybody who is a thief, is punishable. Clearly John is not punishable, and therefore not everybody who is a thief is punishable.

However, if we assume that the first premise is false, it remains unexplained why the argument in its first form seemed to be sound. Somehow, the sentence 'Thieves are punishable' has a quality that allows to infer that John is punishable from the fact that John is a thief, even though the statement that all thieves are punishable is false. We suggest that this quality is not truth as a statement, but rather validity (or acceptability) as a rule. The point of the example is to show that it is possible that the *rule* that thieves are punishable is *valid*, while the corresponding *statement* that all thieves are punishable is *false*.

Sometimes rules are not applied, even if their conditions are satisfied. Reasoning with rules has a characteristic that does not obtain in the case of statements. Rules need to be *applied* to have consequences, and if it occurs that they are not applied, the consequences of the rule fail to obtain, even if the rule conditions are satisfied. The issue of application does not even arise in the case of statements. That is why rules cannot faithfully be represented a statements.

1.2. The application of rules and reasoning about rules

That rules need to be applied to have effects, is reflected in a number of characteristics of reasoning with rules.

a. Defeasibility

Reasoning with rules is defeasible in the sense that it is not always possible to derive the conclusion of a rule if the rule conditions are satisfied. If it is known that the theft is prescribed, it is not possible to conclude that John is punishable from the fact that John committed a theft and the rule that thieves are punishable. A civil law example that was discussed by Hart [1949] is that the conclusion that parties entered into a valid contract is defeated by the additional information that the consent of one of the parties was not free.

One might argue that legal reasoning is not defeasible, but that rules of law contain so-called 'hidden conditions' that are not satisfied in case of an exception. This alternative hypothesis to explain exceptions to rule application suffers from at least two drawbacks.

First it overlooks the change of the burden of proof that is usually connected with defeating conditions. An attorney need not prove that a punishable fact was not prescribed, nor does a contract party who invokes a contract need to prove that the consent of his contract partner was given freely. It is rather the case that who wants to invoke a defeating condition has the burden of proof for this condition. In other words, if defeating conditions are actually hidden rule conditions, they have at least an exceptional status [Baker 1977].

The second drawback is even more serious, because the theory of hidden rule conditions makes it impossible to determine the conditions of a rule unless one knows exactly and in advance to which hypothetical and real cases a rule can be applied and to which cases it cannot be applied. That is because only from the enumeration of the cases to which a rule can be applied we can reconstruct the hidden rule conditions, which cannot be known in another way. (That is precisely why they are 'hidden'.) A consequence of this view is that we cannot use rules to determine the legal consequences of a case, because we should first know whether the rule is to be applied before we can establish the rule conditions. In other words, we cannot use the rule to determine the legal consequences of a case, but rather we must reconstruct the rule conditions from the legal consequences of all relevant cases. This may be a viable analysis of the development of common law through cases, but it is not an acceptable analysis of dealing with exceptions.

It might be objected that the present argument mixes up the validity of arguments as a relation between a set of premises and a conclusion on the one hand, and the process of legal decision making on the other hand. This objection would not be justified, because the point made concerns the usefulness of formal logic as a tool for the evaluation of real life arguments. If we want to use some system of formal logic to evaluate the validity of informal arguments, there must be some relatively straightforward translation between the informal and the formal argument, and vice versa. If we need to add hidden conditions to what seems at first sight to be the formal version of an informal argument, the necessary connection between the form of the two versions of the argument is lost.

b. Analogous application

Sometimes a rule can be applied to a case that does not satisfy all of the rules conditions. This may happen if the case is sufficiently similar to the cases to which the rule is applicable, and then we speak of *analogous rule application*. For instance, until recently the Dutch law contained a rule that if the owner of a rented house sold the house, the rent contract was continued with the new owner. This rule was applied analogously to cases in which the property of the house was transferred on the basis of another title than sale.

It has been argued that analogous rule application does not occur, but that rather another, more abstract rule is applied [e.g. Prakken 1993, pp. 22-23]. We do not agree with this view, because it comes down to the same manoeuvre as the hidden conditions construction for dealing with exceptions. It leads to the consequence that the conditions of the abstract rule that was allegedly applied can only be reconstructed from the cases to which this abstract rule is said to be applied. This would reverse the order of rules of law and legal consequences. It is not anymore the case that legal consequences are established on the basis of the rule, but rather that the conditions of the rule are to be inferred from the cases to which the rule is presumedly applied.

Defeat of legal conclusions is (amongst others) caused by the non-application of a rule whose conditions are satisfied. Analogous rule application is in a sense the opposite; a rule is applied although its conditions are not (all) satisfied. In both cases the natural connection between the satisfaction of the rule conditions and the application of the rule is broken.

c. Reasoning about rules

Rules are not only used in arguments, but they are often also the topic of the conclusions of arguments. In other words, it is not only possible to reason *with* rules, it is also possible to reason *about* rules. An important type of reasoning about rules deals with whether a rule *should* be applied in a particular case.[1]

Because of the disconnection of the satisfaction of the rule conditions and the application of the rule, the question whether a rule should be applied has to be argued over and above the question whether the rule's conditions are satisfied. Analogous rule application is a case in point. Is the similarity between two cases of the right type and sufficient to justify analogous application of a rule? A related type of reasoning about rules deals with the purpose of a rule.

One important type of argument about rules deals with the question whether a particular rule is valid. There may be an argument about the authority for a particular rule. Can the rule be traced back to a recognized source of law? Most often there will be no serious disputes on this matter.

Another variant deals with the interpretation of a rule. It is possible to distinguish at least two senses in which a rule can be said to be interpreted. The first sense deals with the question whether the conditions of a rule are satisfied in a particular case. The issue at stake is a match between a rule and a case. Interpretation in this first important sense is better not considered as dealing with the validity of the rule.

However, there is a second sense in which a rule can be said to be interpreted. In civil law countries most rules originate from legislation. The texts by means of which rules of law are created often do not exhibit a condition/conclusion structure in their formulation of rules. To apply them, they must be translated into the logical model of rules in which it is clear what the conditions and the conclusions of the rules are. Sometimes more than one translation can be defended, and then there may be an argument which of the proposed alternative rule formulations derives authority from the legal text. Discussions of this kind can be conducted along the lines of rule validity. Only those rules are valid, that are justified by the text of the regulation. Clearly, in this case, an interpretation issue is dealt with in a discussion about rule validity.

1.3. An overview of the article

It turns out that reasoning with rules differs fundamentally from reasoning with statements. Reason-Based Logic (RBL) is an extension of classical predicate logic that is especially developed to deal with rules and the reasons that are based on them. In this article we give an exposition of RBL and illustrate its use in a number of - mostly legal - examples. Section 2 contains an informal discussion of RBL and of some of the characteristics of legal reasoning that inspired this logic. This section can be skipped by those who are not interested in the legal theoretical background of RBL. In section 3 we give a formal description of RBL. This section, and in particular its subsection 3.3, may be skipped by those who are not interested in technical details. As mentioned, section 4 contains applications and examples that illustrate how RBL deals with reasons and rules. RBL is not the only logic that was developed to deal with defeasible reasoning. In section 5 and its subsections we compare RBL with some of its alternatives. The article is concluded in section 6.

---

1 Notice the normative, rather than the purely factual nature of the question of rule application. The application of a rule is an *act* that can be justified, and not a timeless logical characteristic of a rule in connection with a case.

2.2. BASIC IDEAS OF REASON-BASED LOGIC

2.1. Two types of rule application?

Especially in the civil law tradition, it sometimes seems that lawyers have two ways of reasoning. The one way occurs if there is a rule of law that can be applied to a case because the rule conditions are satisfied. In this case lawyers just seem to apply the rule as a logical inference, with the conclusion of the rule as the legal consequence for the case at hand. This simple case of rule application is sometimes complicated a little if a rule whose conditions are satisfied is not applied for some reason or another, or if a rule is applied even though its conditions are not satisfied. But even in these cases the conclusion of a single rule dictates what will be the legal consequence, whether this consequence obtains in a particular case or not. There is no conflict of rules, because seeming conflicts are dealt with by leaving one of the conflicting rules out of application on the basis of a conflict rule; at least, this is prevailing legal ideology.

The other way of reasoning occurs if there is no rule of law available and the case is dealt with on the basis of principles, values, interests, policies etc. This situation occurs usually in matters of classification. Then the principles, etc., identify reasons that plead for or against a particular classification.

In exceptional cases there will be rules of law that deal with classification. Such rules are said to contain legal definitions. Article 1 of the Dutch Traffic Statute (Wegenverkeerswet), for instance, specifies amongst others what has to count as a road in the sense of this statute. Most often, however, classification must take place on the basis of the normal meanings of the terms employed in a rule. The problem with this normal meaning, however, is that it is not defined by means of conditions that are individually necessary and together sufficient for the applicability of a legal term.[2] Instead there will be factors that plead for applicability of a term, and factors that plead against applicability. This ‘weighing’ of factors is most conspicuous with terms that have an evaluative meaning component, such as ‘reasonable’ and ‘due’, but also occurs with seemingly purely descriptive terms such as ‘vehicle’. If something is a toy vehicle, this may be a reason against counting it as a vehicle in the sense of a parking prohibition.

If there are conflicting reasons in a case, often each of these reasons considered by itself might decide the issue at stake. If an unemployed person is offered a new job, this job is considered as suitable in the sense of an Unemployment Benefit Act if it is similar to the previous job and has even higher wages. Both facts, similarity of work and increase of payment taken together, are a reason why the newly offered job is suitable, and this reason considered by itself might decide the issue whether the job is suitable. It is possible to formulate this by means of a principle saying that a newly offered job is suitable if it is similar to the old job and if it pays better.

On the other hand, the fact that the distance of the new job is more than three hours of traveling is a reason why the job is not suitable. If the increase in payment is left out of consideration, this fact by itself would be sufficient reason why the job is not suitable. This can also be formulated by means of a principle that says that a newly offered job is not suitable if the traveling distance is more than half an hour.

If applied to the same case, the two mentioned principles conflict. Otherwise than with rules of law, a conflict of principles is not solved by conflict rules that determine which rule must be left out of application. Rather both principles are applied, but instead of deciding the case by themselves, they only generate a reason for their respective conclusions. These reasons have to be weighed to determine the outcome of the case.[3]

2.2. Exclusionary reasons

It seems that the application of rules of law and of legal principles differs essentially. The application of a principle only generates a reason for its conclusion. Consequently, two principles with incompatible conclusions can both be applied in one case, without leading to an inconsistency. Therefore, there is no need for conflict rules (or principles) in the case of colliding principles.

---

2 This means that the classical theory of meaning in which terms are said to have an ‘intention’, that consists precisely of such a set of conditions, is wrong. Cf. Schwartz 1977 and Smith 1990.

3 Cf. Dworkin’s account of the difference between legal rules and legal principles. [Dworkin 1978, pp. 22f.]

Legal rules, on the contrary, seem either to apply or not. If a rule of law applies, it seems that its conclusion follows necessarily. Application of two rules of law with incompatible conclusions would necessarily lead to an inconsistency, and should therefore be avoided. The role of conflict rules is to assure that in the case of two incompatible rules of law, one of the rules will not be applied.

We want to argue, however, that the difference between rules of law and legal principles is not so fundamental as might seem at first sight and that it is only a matter of degree.[4] We propose the theory that both rules of law and legal principles only generate *reasons* for their conclusions. The difference between rules of law and legal principles is that a rule of law, if it is applied, not only generates reasons for its own conclusion, but also generates reasons against the application of conflicting rules. We will call these latter reasons *exclusionary reasons.*[5]

Because the application of a rule of law excludes the application of conflicting rules, there will usually not be any reasons that conflict with the reason generated by the applied rule. As a consequence, no weighing of reasons is necessary and it seems as if the applied rule works in an absolute sense. Cf. figure 1.

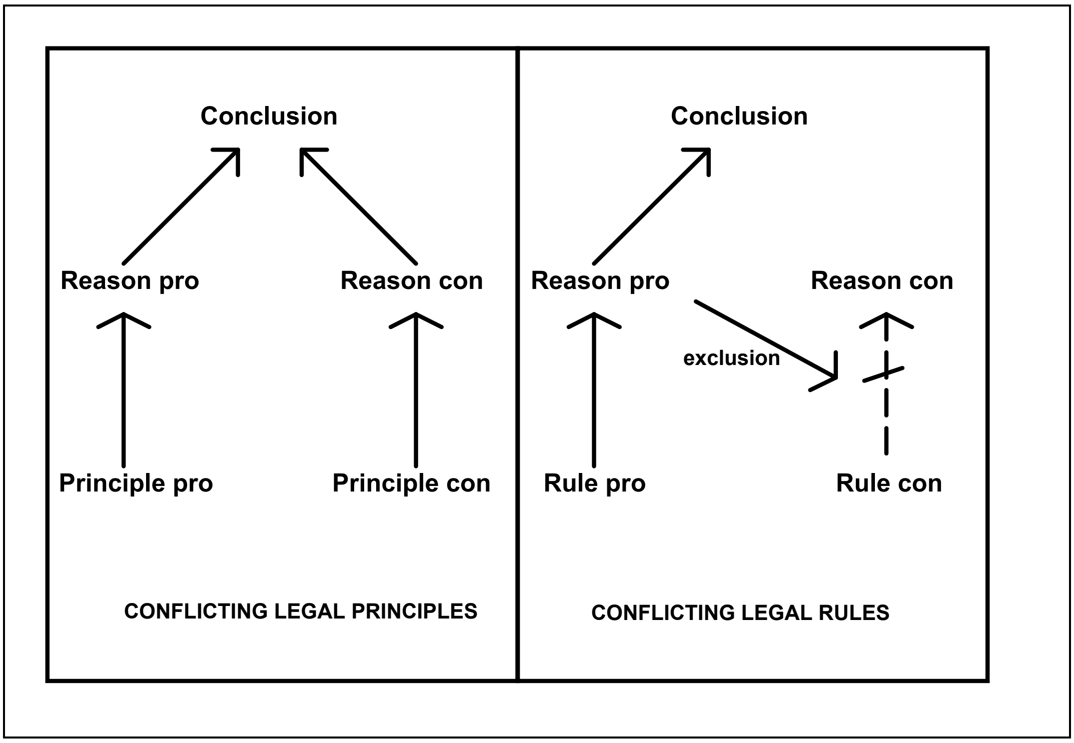

Figure 1. Conflicting principles and conflicting rules

Let us consider an example to illustrate this essential point. Suppose that the constitution of a country unconditionally gives everybody the right to practise his religious duties. The municipal authorities, on the other hand, forbid religious processions on working days to guarantee a free flow of traffic. If both are rules of law, there will be a conflict rule (Lex Superior) that says that the constitution prevails over the municipal regulation. As a consequence, the local prohibition of procession is not applied and does not generate a reason against processions, and there will only be a reason based on the constitution why a particular procession is allowed.[6] No weighing is necessary and the constitution seems to be the only relevant factor. (This is the situation that is depicted in the right hand side of figure 1.)

However, if we change the example so that the constitution does not grant an absolute right, but only states a general principle of law, the situation changes drastically. In this case, there

---

4 Cf. the criticism by Alexy on Dworkin's account of the difference between legal rules and legal principles. [Alexy 1979]

5 Cf. Raz's theory about exclusionary reasons and their relation to mandatory norms. [Raz 1975, pp. 49f. and 85f.]

6 For the purpose of illustration we neglect the real possibility that the municipal law is more specific.

is no conflict of rules of law, but only a conflict between a principle and a municipal rule. Most probably, the application of the municipal rule excludes the application of the principle expressed in the constitution. This becomes more plausible if we assume that the municipal regulation already takes account of the constitutional principle by allowing processions on non-working days. Again there is no conflict of reasons, because only the reason against the procession based on the municipal rule obtains. The municipal rule seems to govern the situation by itself.

We can still imagine another case, in which there is no municipal regulation. In that case the legality of the procession is solely to be decided on the basis of legal principles. On the one hand there is the principle of freedom of religion; on the other hand there is the principle that protects the free flow of the traffic. Both principles generate a reason; one reason pleads for the legality of the procession, the other against. To determine the outcome of the conflict, additional information is needed about the relative weight of the reasons. (This is the situation that is depicted in the left hand side of figure 1.)

As becomes clear from these examples, it is possible to deal with the distinction between rules of law and legal principles by making use of exclusionary reasons. Rules differ from principles in that they do not only generate reasons for their own conclusions, but also reasons against the application of other rules with conflicting conclusions.

It is not necessarily so that, if a rule applies, *all* other rules with conflicting conclusions are excluded. It may be the case that the application of some conflicting rules is excluded, while other conflicting rules are still applied.[7] If this happens, there will still be a conflict of reasons, although the conflict will contain less reasons than if only principles would have been involved.

The notion of an exclusionary reason makes it possible to deal uniformly with both legal rules and principles, and even to abolish the distinction for logical purposes. For each individual rule it must explicitly be decided whether its application excludes the application of other rules, and which other rules are excluded. The more conflicting rules are excluded, the more prototypical a legal rule is; the less conflicting rules are excluded, the more a legal rule is like a prototypical legal principle.

2.3. An informal introduction to Reason-Based Logic

We developed a logic to deal with rules and the reasons that are based on them. This logic is called Reason-Based Logic (RBL). RBL does not use the notion of a principle and captures both rules of law and principles under the notion of an RBL-rule.

The basic idea of RBL is that the application of any rule leads to a reason which pleads for the rule's conclusion, and that the actual derivation of a sentence is based on weighing all the reasons that plead either for or against the truth of this sentence. As a consequence, derivation is a two step procedure. The first step consists of the determination of all reasons that plead for or against the possible conclusion; the second step consists of weighing these reasons.

a. Weighing reasons

Suppose we have the following two rules:

Rule 1: If A and B, then P.

Rule 2: If C, then not-P.

If we want to derive that P, we should collect all the reasons that plead for and against P. Rule 1 can generate a reason for P. If it is applied, it makes the facts A and B together into a reason for P. Rule 2 can generate a reason against P. If rule 2 is applied, C becomes a reason against P. If both rules are applied, we have two reasons: A & B plead for P, while C pleads against P. To determine whether we can derive P, we must weigh A & B on the one hand against C on the other hand.

To determine which reason wins, we need additional information. Such information is supplied by an additional premise, for instance, that C outweighs A & B. Since C pleads against P, this means that P cannot be derived; on the contrary, the reasons against P outweigh the reasons for P, and therefore it is possible to derive not-P.

If we do not have any information about the relative weight of the reasons for and against a conclusion, it is not possible to derive anything. There is, however, one important case in

---

7 It is beyond the scope of this article to discuss the situations in which this would occur.

which we need no explicit weighing information. This is the case in which there are only reasons for a conclusion, or only reasons against a conclusion. If, for instance, only Rule 1 would be applied, we only have a reason for P, and no reasons against P. In such a case we should of course be able to derive P without the need of additional information.

b. Exclusionary reasons

Until now we have assumed that both Rule 1 and Rule 2 are applied. Normally a rule is applied if its conditions are satisfied, This means that Rule 1 is normally applied if A and B are both true. Similarly, Rule 2 is normally applied if C is true.

However, we have seen that the application of a rule can be excluded. In fact, the example at the beginning of this article about John's theft that was prescribed, is a case of rule exclusion. In that case, we had the rule

Rule 3: If somebody is a thief, this person is punishable.

But we also had the rule

Rule 4: Nobody is punishable if his crime is prescribed.

Since Rule 4 prevails over Rule 3 (this should be explicitly stated in an additional premise), the application of Rule 3 is excluded by the application of Rule 4. As a consequence, the fact that John is a thief does not become a reason why he can be punished.

c. Reasons against the application of a rule

If, in a particular case, the conditions of a rule are satisfied, and the application of this rule is not excluded, we say that the rule is *applicable* to that case. Being applicable is an important step toward application of a rule, but it is not yet all the way. If a rule is applicable, the facts that make it applicable are a reason to apply the rule. Usually it is the only reason concerning application, and as often when there is only one reason, this reason decides about the application, and the rule is applied.

However, in exceptional cases it occurs that there are also reasons against the application of a rule. Most notably this happens if application would be against the rule's purpose. In these cases, the facts that make the rule applicable as a reason that pleads for application, must be weighed against the reason against application, viz. that application would be against the rule's purpose. There is no standard outcome in this situation. Sometimes, if legal certainty and predictability of the law are very important, the facts that make the rule applicable will win, and the rule will be applied. In other cases, where predictability is less important and justice and fairness make heavy claims, the purpose of the rule will win, and the applicable rule is not applied.

Reasons against the application of a rule are not the same as exclusionary reasons. Exclusionary reasons block the application of a rule; no weighing of reasons for and against application is necessary. Reasons against application are less powerful. They need to be weighed against the facts that make the rule applicable as a reason for application of the rule. It depends on the circumstances of the case whether the rule will be applied or not.

3. A FORMALIZATION OF REASON-BASED LOGIC

RBL is an extension of first order predicate logic (FOPL) in the sense that the language of RBL is based on the language of FOPL, and that the axioms and rules of inference of FOPL are valid in RBL. In section 3.1 we describe the language of RBL, and in section 3.2 the rules of inference of RBL. In section 3.3 we define what can be derived from a set of RBL sentences.

3.1. The language of RBL

One part of the extension with regard to FOPL concerns the use of the logical language. In RBL it is not only necessary to *use* sentences, but also to *mention* sentences. Formally, this means that a sentence can also be an argument of a predicate. Since predicates can only have terms as arguments, we need a translation of sentences (and, more generally, formulas) to terms. For this purpose, we will write predicate symbols as strings of characters beginning with an uppercase letter and function symbols as strings of characters beginning with a lowercase letter. To obtain the term that corresponds to a formula, the first (uppercase) letter of each predicate symbol in the formula is replaced by the same letter in lowercase.[8] For example, the formulas

Thief(mary)
Guilty(mary) & Punish(mary)

are respectively referred to by the terms

thief(mary)
guilty(mary) & punish(mary).

In this way we avoid confusion between use and mention of a sentence.

The language of RBL is that of FOPL, but contains a number of special function and predicate symbols, that is rule/3,[9] rule/1, { **.**, **.**, ..., **.** }/n (for n = 0, 1, 2, ...),[10] True/0, Valid/1, Excluded/1, Applicable/3, Applies/3, Reason/3, Outweighs/3 and Ought/1.

- rule/3

  In RBL rules are represented by *terms* of the language. In this way it is possible to refer to them and to reason about them. A term denoting a rule has the form:[11]

  rule(*id*, *condition*, *conclusion*)

  Here *condition* is a formula of RBL and *conclusion* a literal of RBL. If a variable occurs free in one of them, it must also be free in the other. We assume that *condition* is a disjunction of conjunctions of one or more literals.[12] Each disjunct of *condition* is a possible reason for *conclusion*.

  The first argument of a rule, namely *id*, is called the identifier of the rule. It is assumed that in an RBL-theory (the set of sentences on which derivations are based) each rule has a unique identifier.

- rule/1

  The term rule(*id*) is used as an abbreviation of the term rule(*id*, *condition*, *conclusion*). Because the identifier of a rule is unique, this does not lead to confusion.

- { **.**, **.**, ..., **.** }/n (for n = 0, 1, 2, ...)

  These symbols are used to refer to sets of reasons. The term { thief(mary), minor(mary) } denotes the set of reasons that consists of the formulas Thief(mary) and Minor(mary). The term { } (without arguments) is used to denote the empty set of reasons.[13]

- True/0

  The formula True can be derived in FOPL from any set of sentences.

- Valid/1

  The formula Valid(rule(*id*)) means that the rule with identifier *id* is valid.

- Excluded/1

  The formula Excluded(rule(*id*)) means that the rule with identifier *id* is excluded.

---

8 The connectives of FOPL, e.g. $\rightarrow$ and &, are treated as if they also are function symbols. By overloading the notation, the translation of formulas to terms is as simple as mentioned.

We do not use quantifiers in this paper. A universally quantified formula can be mimicked by a formula with free variables: a formula with free variables is considered as a *scheme* for its closed instances. An existentially quantified formula can be mimicked by replacing the existentially quantified variables by appropriate terms. Details on these so-called Skolemized formulas can be found in, for example, Lukaszewicz (1990).

9 The number following / denotes the arity of the function or predicate symbol.

10 What is meant by this notation will become clear below.

11 Metavariables for formulas will be denoted as strings of italic characters beginning with an upper case character, e.g., *Atom*. Metavariables for terms will be denoted as strings of italic lower case characters, e.g., *atom*. We use the convention that matching metavariables, such as *Atom* and *atom*, represent a formula and its corresponding term.

12 If we speak informally of the conditions of a rule we mean these literals. In formal notations we write *condition* (singular).

13 There is a problem here with different terms that denote identical sets, such as { thief(mary), minor(mary) } and { minor(mary), thief(mary) }. Axioms should be included in the theory of RBL such that formulas that only differ in such equivalent terms for sets are equivalent. We will not do this explicitly.

- Applicable/3

  The formula Applicable(rule(*id*), *facts*, *conclusion*) means that the rule with identifier *id* is made applicable by the facts denoted by the term *facts* and may generate a reason for the conclusion denoted by the term *conclusion*.
- Applies/3

  The formula Applies(rule(*id*), *facts*, *conclusion*) means that the rule with identifier *id* applies on the basis of the facts denoted by the term *facts* and generates a reason for the conclusion denoted by the term *conclusion*. The difference with the predicate Applicable is explained in the next section.
- Reason/3

  The formula Reason(*facts*, *atom*, pro) means that the facts denoted by the term *facts* are a reason for the conclusion denoted by the term *atom*. The formula Reason(*facts*, *atom*, con) means that *facts* are a reason against *atom*.
- Outweighs/3

  The formula Outweighs($reasons_1$, $reasons_2$, *atom*) means that the reasons in the set denoted by the term $reasons_1$ outweigh the reasons in the set denoted by the term $reasons_2$ (as reasons concerning *atom*). The terms $reasons_1$ and $reasons_2$ must both have the form { $facts_1$, $facts_2$, ..., $facts_n$ }, where $n \geq 0$.
- Ought/1

  The predicate Ought has to be interpreted as an operator on sentences: it transforms the sentence *Literal* into the deontic sentence Ought(*literal*).[14]

3.2. Inference in RBL

The second respect in which RBL is an extension of FOPL is that there are extra rules of inference. In this section we describe five rules and two axiom schemes, that characterize inference in RBL.

An RBL theory is a set of RBL formulas. The derivations from RBL theories are governed by the following rules. Let T be an RBL theory.

R1 a. Any formula that can be derived from T in FOPL can be derived from T in RBL.

b. Any formula that can be derived in FOPL from formulas that can be derived from T in RBL can be derived from T in RBL.

This rule implies that the consequences of an RBL theory are deductively closed.

R2 If Valid(rule(*id*, *condition*, *conclusion*)), and *Facts* can be derived,[15] and Excluded(rule(*id*)) cannot be derived, then Applicable(rule(*id*), *facts*, *instance-of-conclusion*) can be derived, if:

1. The formula *Facts* is an instance of one of the disjuncts of the formula *Condition* under some substitution σ.
2. The term *instance-of-conclusion* is the instance of the term *conclusion* under σ.

Intuitively, this can be understood as saying that a valid rule is applicable if its conditions are satisfied, and if it is not excluded. Here *Facts* stands for the facts that make the rule applicable. The definition is such that reasons based on rules with alternative conditions are always based on the satisfaction of one of the alternatives.

RBL does not define under which circumstances a rule is excluded. This has to be specified in the theory T. **R2** only indicates that a rule is not applicable if it can be derived that it is excluded, even if the rule conditions are satisfied.

R3 Let *Atom* be an atom of RBL. If Applicable(rule(*id*), *facts*, *atom*) can be derived, then Reason(*facts*, applies(rule(*id*), *facts*, *atom*), pro) can be derived.[16]

---

[14] Following our convention, this should be interpreted as the statement that the sentence *Literal* ought to be the case. What is meant here is of course that *what is expressed* by the sentence ought to be the case.

[15] The word 'derive' means 'derive in RBL from T', if not otherwise specified.

[16] This is the first time the convention on matching metavariables mentioned in note 11 is used: *Atom* and *atom* represent a formula and its corresponding term.

Intuitively this rule can be understood as saying that if a rule is applicable, the facts that make the rule applicable are a reason to apply the rule.

Notice the difference between a rule's being applicable and its being applied (Applies(rule(*id*))). The former only indicates that there is a reason for the latter.

There can also be reasons against applying a rule. The circumstances under which such reasons occur have to be specified by the theory T.

R4 Let *Atom* be an atom of RBL.

a. If Applies(rule(*id*), *facts*, *atom*) can be derived, then Reason(*facts*, *atom*, pro) can be derived.
b. If Applies(rule(*id*), *facts*, ~*atom*) can be derived, then Reason(*facts*, *atom*, con) can be derived.

Intuitively this rule can be understood as saying that if a rule applies, the facts that make the rule applicable are a reason for or against the rule conclusion, depending on whether the rule pleads for or against the conclusion.

R5 Let *Atom* be an atom of RBL, let Reason($facts\_pro_1$, *atom*, pro), Reason($facts\_pro_2$, *atom*, pro), ..., Reason($facts\_pro_n$, *atom*, pro) be all the reasons for *Atom* that can be derived, and let Reason($facts\_con_1$, *atom*, con), Reason($facts\_con_2$, *atom*, con), ..., Reason($facts\_con_m$, *atom*, con) be all the reasons against *Atom* that can be derived.[17]

Let *reasons_pro*(*atom*) be an abbreviation of the term { $facts\_pro_1$, $facts\_pro_2$, ..., $facts\_pro_n$}, and *reasons_con*(*atom*) an abbreviation of { $facts\_con_1$, $facts\_con_2$, ..., $facts\_con_m$ }.

a. If Outweighs(*reasons_pro*(*atom*), *reasons_con*(*atom*), *atom*) can be derived in RBL, then *Atom* can be derived.
b. If Outweighs(*reasons_con*(*atom*), *reasons_pro*(*atom*), *atom*) can be derived in RBL, then ~*Atom* can be derived.

Intuitively this rule says that a conclusion can be derived if the derivable reasons that plead for it outweigh the derivable reasons that plead against it, and that the negation of the conclusion can be derived if it is the other way round. If neither set of reasons outweighs the other set, nothing can be derived.

In general, the knowledge which set of reasons outweighs the other set should be provided by the theory T. However, in the case that all reasons point in the same direction, i.e., all reasons are either pros or cons, the following axiom scheme provides the necessary weighing knowledge: any non-empty set of reasons outweighs the empty one.

A6 Let *Atom* be an atom of RBL, and let $facts_1$, $facts_2$, ..., $facts_n$, where $n > 0$, be a series of conjunctions of literals. Then Outweighs({ $facts_1$, $facts_2$, ..., $facts_n$ }, { }, *atom*) can be derived in RBL.

Although it is impossible to derive by means of **R2** that a rule is applicable from the fact that its conditions are satisfied in case the rule is excluded, it is still logically possible that a rule is both applicable and excluded. The following axiom scheme takes away this possibility:

A7 Let *Condition* be a disjunction of conjunctions of literals, and *Conclusion* a literal. Then Applicable(rule(*id*), *condition*, *conclusion*) $\rightarrow$ ~Excluded(rule(*id*)) can be derived.

Unfortunately, the rules **R1** to **R5** above cannot easily be turned into an inductive construction of the set of formulas that can be derived from a theory T. This is due to the rules **R2** and **R5,** that both refer to the *whole* set of formulas that can be derived. **R2** requires that some statement *cannot* be derived, which can only be checked if we know everything that can be derived. **R5** makes use of *all* the reasons for and against a conclusion that can be derived from a theory T. The other properties only require that *specific formulas* can be derived.

[17] We do not consider the theoretical case that there is an *infinite* number of reasons.

3.3. Extensions

In this section we define an extension E of a theory T of RBL, analogous to Reiter's definition of an extension in default logic (Reiter, 1980).[18] Informally, an extension of a theory T is a minimal set of formulas that contains all formulas that are supported by the other formulas in E and no formulas that are not supported. An extension must be minimal with respect to set inclusion, because otherwise it could contain unsupported sets of formulas, such as loops.

First, we define the operator Γ, that operates on arbitrary sets of formulas of RBL. If S is a set of formulas, the set Γ(S) is informally the set that contains *only* the formulas that are supported by S using T and (roughly) the rules **R1** to **R5** and the axiom schemes **A6** and **A7** (see section 3.2).

An extension can then be defined as a *fixed point* of Γ, i.e., a set of formulas E is an extension, if E = Γ (E). Intuitively, the inclusion E $\subseteq$ Γ(E) means that all formulas of E are actually supported, and the inclusion E $\supseteq$ Γ(E) that all supported formulas are actually in E.

a. Definitions

Γ is defined in terms of properties **P1** to **P7** that closely resemble **R1** to **R5**, **A6** and **A7**. The important point is the adaptation in the properties **P2** and **P5**. References to the whole set of derived formulas is replaced by references to S. Property **P2** informally means that a rule is applicable in Γ(S) if, amongst others, it is not excluded in S. Property **P5** informally means that all those conclusions are included in Γ(S), that are based on the reasons contained in S.

Let T be a theory of RBL and S a set of formulas of RBL. Then Γ(S) is defined as the smallest set that has the following properties:

P1 a. Any formula that can be derived from T in FOPL is an element of Γ(S)
b. Any formula that can be derived in FOPL from formulas that are elements of Γ(S) is an element of Γ(S).

P2 If Valid(rule(*id*, *condition*, *conclusion*)) and *Facts* are elements of Γ(S), and Excluded(rule(*id*)) is not an element of S, then Applicable(rule(*id*), *facts*, *instance-of-conclusion*) is an element of Γ(S), if:
1. The formula *Facts* is an instance of one of the disjuncts of the formula *Condition* under some substitution σ.
2. The term *instance-of-conclusion* is the instance of the term *conclusion* under σ.

P3 Let *Atom* be an atom of RBL. If Applicable(rule(*id*), *facts*, *atom*) is an element of Γ(S), then Reason(*facts*, applies(rule(*id*), *facts*, *atom*), pro) is an element of Γ(S).

P4 Let *Atom* be an atom of RBL.
a. If Applies(rule(*id*), *facts*, *atom*) is an element of Γ(S), then Reason(*facts*, *atom*, pro) is an element of Γ(S).
b. If Applies(rule(*id*), *facts*, ~*atom*) is an element of Γ(S), then Reason(*facts*, *atom*, con) is an element of Γ(S).

P5 Let *Atom* be an atom of RBL, let Reason($facts\_pro_1$, *atom*, pro), Reason($facts\_pro_2$, *atom*, pro), ..., Reason($facts\_pro_n$, *atom*, pro) be all the reasons for *Atom* that occur in S, and let Reason($facts\_con_1$, *atom*, con), Reason($facts\_con_2$, *atom*, con), ..., Reason($facts\_con_m$, *atom*, con) be all the reasons against *Atom* that occur in S.
Let *reasons_pro*(*atom*) be an abbreviation of the term { $facts\_pro_1$, $facts\_pro_2$, ..., $facts\_pro_n$ }, and let *reasons_con*(*atom*) be an abbreviation of { $facts\_con_1$, $facts\_con_2$, ..., $facts\_con_m$ }.
a. If Outweighs(*reasons_pro*(*atom*), *reasons_con*(*atom*), *atom*) is an element of Γ(S), then *Atom* is an element of Γ(S).
b. If Outweighs(*reasons_con*(*atom*), *reasons_pro*(*atom*), *atom*) is an element of Γ(S), then ~*Atom* is an element of Γ(S).

P6 Let *Atom* be an atom of RBL, and let $facts_1$, $facts_2$, ..., $facts_n$, where n > 0, be any series of conjunctions of literals. Then Outweighs({ $facts_1$, $facts_2$, ..., $facts_n$ }, { }, *atom*) is an element of Γ(S).

---

[18] Verheij (1994) describes this same technique for a restricted version of RBL.

P7 Let *Condition* be a disjunction of conjunctions of literals, and *Conclusion* a literal. Then Applicable(rule(*id*), *condition*, *conclusion*) → ~Excluded(rule(*id*)) is an element of Γ(S).

For any set of formulas S the set Γ(S) exists. It is the intersection of all sets with the properties **P1** to **P7**. This intersection exists because there exists a set that has the properties **P1** to **P7** (namely the set of all formulas), and because the intersection of sets that have the properties **P1** to **P7** also has these properties.

A set E of formulas of RBL is called an extension of T if E = Γ(E).

A formula is derivable from a theory T if and only if it is in *all* the extensions of T.

b. Characteristics of extensions

An important characteristic of an extension E is that it is based on the rules **R1** to **R5** and the axiom schemes **A6** and **A7** (see section 3.2). This can be checked by going stepwise through the definition of an extension and comparing the defining properties of the Γ-operator with the inference rules and the axiom. The phrase 'is an element of E' in the defining properties replaces 'can be derived from T in RBL' in the rules of inference.

A theory does not necessarily have an extension, and, if it has one, the extension is not necessarily unique. For instance, the theory { Valid(1, true, excluded(1)) } has no extension. The theory { Valid(1, true, excluded(2)), Valid(2, true, excluded(1)) } has two extensions, namely one in which rule 1 is excluded and rule 2 applies, the other in which rule 1 applies and rule 2 is excluded.

In the case that there is more than one extension of a theory, we prefer what is called a *skeptical* approach (as opposed to a *credulous* one). This means that the *intersection* of the extensions of a theory is considered to be the set of derivable formulas. The reason for this skepticism is that theories that have more than one extension would have circular derivations for sentences that occur in some, but not all of the extensions. Such formulas should not be derivable.

Theories that have no, or more, extensions contain a paradox based on (in)direct self-reference. These paradoxes are made possible because formulas of RBL can refer to other formulas. We consider it the tasks of theories in RBL, rather than of the language of RBL to avoid these paradoxes.

Our definition of derivability via extensions has the counterintuitive consequence that all formulas can be derived from a theory without an extension. From the first example theory above, containing Valid(1, true, excluded(1)), all formulas are derivable; from the second theory, containing Valid(1, true, excluded(2)) and Valid(2, true, excluded(1)), only the deductive closure is derivable. This difference seems strange because the theories suffer from the same type of self-referential paradox.

Finally, we wish to stress that we do not find the definition of extensions of an RBL theory completely satisfactory: a fixed point definition is not intuitive and does not lead to a constructive definition of what can be derived.

4. EXAMPLES

In the following subsections, we illustrate the facilities that RBL offers to deal with various types of reasoning in which rules are involved.

4.1. Elementary rule application

Suppose we have the sentences:

Thieves ought to be punished.
Peter is a thief.

From this information we want to derive:

Peter ought to be punished.

These natural language sentences can be represented by the following RBL sentences:

Valid(rule(theft, thief(*x*), ought(punished(*x*))))[19]
Thief(peter)
Ought(punished(peter))

[19] From now on italic strings of italic characters denote variables of the logical language instead of metavariables (cf. note 11).

We show (somewhat informally) how this last sentence follows from the theory that consists of the first two sentences.

By means of **R1**a we can derive:

Valid(rule(theft, thief(*x*), ought(punished(*x*))))
Thief(peter)

By means of **R2**, provided that Excluded(theft) cannot be derived, we can derive:

Applicable(rule(theft), thief(peter), ought(punished(peter)))

By means of **R3** we can derive:

Reason(thief(peter), applies(rule(theft), thief(peter), ought(punished(peter))))

By means of **A6** follows:

Outweighs({ thief(peter) }, { }, applies(rule(theft), thief(peter), ought(punished(peter))))

By means of **R5**a, provided that thief(peter) is the only reason that can be derived concerning the application of the rule called theft, it follows that:

Applies(rule(theft), thief(peter), ought(punished(peter)))

By means of **R4**a we can derive:

Reason(thief(peter), ought(punished(peter)), pro)

Again, by means of **A6** we derive:

Outweighs({ thief(peter) }, { }, ought(punished(peter)))

And, finally, by means of **R6**a, provided that thief(peter) is the only reason that can be derived concerning the punishment of Peter, we derive the conclusion that we wanted:

Ought(punished(peter))

The only other sentences that can possibly be derived are FOPL consequences of the sentences above (by **R1**b). For example, because there is no formula in the theory containing Excluded(rule(theft)), Excluded(rule(theft)) cannot be derived. In the same vein, we conclude that thief(peter) is the only derivable reason. This means that the constraints for the derivation of the sentences

Applicable(rule(theft), thief(peter), ought(punished(peter)))
Applies(rule(theft), thief(peter), ought(punished(peter)))
Ought(punished(peter))

are not violated.

It can be formally checked that the deductive closure of the set containing the above mentioned sentences (plus the axiomatic information provided by **A6** and **A7**) is the only extension of the theory that consists of the sentences

Valid(rule(theft, thief(*x*), ought(punished(*x*))))
Thief(peter)

4.2. The 'derivation' of rules

Notice that in the previous example it is stated that rule called theft is a valid rule. Since rules are individuals, and not sentences, they cannot stand on their own in a theory of RBL. Moreover, only a valid rule can only be used in a derivation to create reasons. These two facts together are the reason why rules are introduced into arguments by means of the statement that they are valid.

This has the additional advantage that it becomes possible to 'derive rules', without the need of extending the inferential apparatus of RBL to allow the derivation of individuals.

Suppose that we have

Valid(rule(legislation, adopted_by_legislator(rule(*id*)), valid(rule(*id*))))
Adopted_by_legislator(rule(statutory_rule))

From these two formulas it is possible to derive (the validity of) the rule called statutory_rule.

This way to derive rules is specifically legal. (The rule called legislation is a rule of competence.) It is, however, possible to give more general rules about the derivation of rules. For instance, a theory might include the following rule-expressing sentence:

Valid(rule(transitivity_of_validity,
valid(rule(*id1*, *condition*, *conclusion1*)) & valid(rule(*id2*, *conclusion1*, *conclusion2*)),
valid(rule(*id3*, *condition*, *conclusion2*)))).

This rule would be the counterpart in RBL of what in FOPL would be the tautology:

$((P \rightarrow Q) \,\&\, (Q \rightarrow R)) \rightarrow (P \rightarrow R)$.

Since the rule called transitivity_of_validity would not be generally valid in, for instance, the legal domain, it is not included as an axiom-scheme in RBL proper.[20]

4.3. Weighing reasons

Suppose we have the following RBL-theory:

Valid(rule(theft, thief(*x*), ought(punished(*x*))))
Thief(peter)
Valid(rule(youth, had_difficult_youth(*x*), ~ought(punished(*x*))))
Had_difficult_youth(peter)

On the basis of these sentences we obtain two reasons regarding the punishment of Peter:

Reason(thief(peter), ought(punished(peter)), pro)
Reason(had_difficult_youth(peter), ought(punished(peter)), con)

On the basis of these reasons alone, neither

Ought(punished(peter))

nor

~Ought(punished(peter))

can be derived. To derive a conclusion we need extra knowledge about the relative weight of the reasons. Suppose we also have:

Outweighs({ had_difficult_youth(*x*) }, { thief(*x*) }, ought(punished(*x*))).

This formula makes it possible to derive by means of **R5**b:

~Ought(punished(peter)).

In RBL it is possible to derive the knowledge about the relative weight of sets of reasons from other information. It is completely open to a theory in RBL whether and how it specifies how reasons are to be weighed. For instance, the relative weights of sets of reasons can be made dependent on other reasons, that concern the weight of the reasons regarding a conclusion, but are not themselves reasons concerning that conclusion.

Such a situation is illustrated by the following real life example. A small supermarket had to dismiss one of its employees for financial reasons. One of the employees, called Mary, has been longer in service, and this is a reason to dismiss the other employee, called Richard:

Reason(longer_in_service(mary), dismiss(richard), pro).

The other employee, on the other hand, has better papers for the job, and this is a reason not to dismiss him:

Reason(better_papers(richard), dismiss(richard), con).

The judge decided that, although Richard had better papers for the job, Mary was still sufficiently qualified, so that the better papers did not count for much. The fact that Mary had been longer in service should therefore tip the balance of reasons:

Reason(suitable_for_job(mary),
outweighs({longer_in_service(mary)}, {better_papers(richard)}, dismiss(richard)),
pro).

Notice that the fact that Mary was suitable for the job was not considered as a reason to dismiss Richard, but only as a reason why the seniority of Mary should outweigh the better papers of Richard.

Another interesting way to argue about the relative weight of sets of reasons is based on cases. The decision in a case can be considered as a reason why the reasons in a new case should be weighed in the same way as they were weighed in the decided case [Hage 1993].

4.4. Exclusionary reasons

Exclusionary reasons form the basis for an essential mechanism in RBL. The exclusion of rules plays a role in various important types of legal reasoning. We first discuss the simple

---

[20] Delgrande's conditional logic (Delgrande, 1988) contains a weaker version of rule 4 as a rule of inference:
From $B \rightarrow C$, infer $(A \Rightarrow B) \rightarrow (A \Rightarrow C)$.
The RBL-version of this rule would run:
Valid(rule(4', valid(rule(*id1*, *cond*, *concl1*)) & *concl1* $\rightarrow$ *concl2*, valid(rule(*id2*, *cond*, *concl2*)))

example with which we started this article. Then follows an illustration of how conflict rules are modeled. We close the discussion of exclusionary reasons with an example that shows how the mechanism of exclusion can be applied on more than one level of reasoning.

a. The scope of rules

Rules have a scope in time. In the case of criminal law, this means that a criminal law may not be applicable anymore, if the crime occurred sufficiently long ago. This situation is illustrated in our old example about John, the thief. Although John was a thief, he was not punishable, because his crime was prescribed. Let us formulate the relevant data in an RBL theory:

Valid(rule(theft2, thief(*x*), punishable(*x*)))
Valid(rule(prescription, prescribed_theft, excluded(rule(theft2))))
Thief(john)
Prescribed_theft

From

Prescribed_theft

it can be derived that

Excluded(rule(theft2))

and as a consequence it is not anymore possible to derive that

Reason(thief(john), punishable(john), pro)

and as a further consequence, the sentence

Punishable(john)

cannot be derived anymore.

b. General conflict rules

The law knows at least three general conflict rules that deal with the conflict of rules. These three are Lex Specialis, Lex Superior, and Lex Posterior. These rules can conflict amongst themselves too. Let us see how RBL can deal with these conflict rules and with conflicts between conflict rules.

If two rules of law are in conflict, one of the two prevails over the other one. The latter rule may be said to be excluded:

Valid(rule(prevalence, prevails_over(rule(*id1*), rule(*id2*)), excluded(rule(*id2*))))

Exactly when does one rule prevail over another one? When they are in conflict, and the one rule is identified by a conflict rule as the prevailing one:

Valid(rule(lex_specialis,
in_conflict(rule(*id1*), rule(*id2*))) & more_specific(rule(*id1*), rule(*id2*))),
prevails_over(rule(*id1*), rule(*id2*))))
Valid(rule(lex_superior,
in_conflict(rule(*id1*), rule(*id2*))) & superior(rule(*id1*), rule(*id2*))),
prevails_over(rule(*id1*), rule(*id2*))))
Valid(rule(lex_posterior,
in_conflict(rule(*id1*), rule(*id2*))) & more_recent(rule(*id1*), rule(*id2*))),
prevails_over(rule(*id1*), rule(*id2*))))

It remains to be specified when two rules are in conflict. One reason why rules might be in conflict is that they have opposite conclusions.

c. Conflicting conflict rules

Since the conflict rules as formalized here are RBL-rules, they generate reasons why one rule prevails over another rule. This means that if two conflict rules have incompatible conclusions, they generate reasons that must be weighed. (Unless, of course, one of the two is excluded by the application of the other one.)

Such a situation of conflicting conflict rules occurs in the example about the freedom of religion, in which a constitutional right collides with a municipal traffic regulation. The situation discussed in that example might be described as follows:

Valid(rule(freedom_of_religion, exercise_of_religion(*act*), permitted(*act*)))
Valid(rule(traffic, procession_on_weekday(*act*), ~permitted(*act*)))
Superior(rule(freedom_of_religion), rule(traffic))
More_specific(rule(traffic), rule(freedom_of_religion))

Procession_on_weekday(procession)
Exercise_of_religion(procession)

Because the municipal traffic rule is more specific than the constitutional rule about freedom of religion, there is a reason why the traffic rule prevails over the constitutional rule. On the other hand, the constitutional rule is superior, and that is a reason why this rule should prevail. There is a conflict of reasons which must be decided by additional weighing knowledge.[21] Suppose we have

Outweighs({superior(rule(*id1*), rule(*id2*))}, {more_specific(rule(*id2*), rule(*id1*))},
prevails_over(rule(*id1*), rule(*id2*)))

On the basis of this weighing knowledge, we can derive that

Prevails_over(rule(freedom_of_religion), rule(traffic))

and consequently also

Excluded(rule(traffic))

and

Permitted(procession)

4.5. Reasons against the application of a rule

An exclusionary reason makes that the fact that the conditions of a rule are satisfied is not a reason for the application of the rule. The situation is different in the case that there are reasons not to apply a rule. If there are such reasons, they must be weighed against reasons that plead for the rule's application. RBL proper does not say anything about reasons against the application of a rule. Yet, because Applies(rule(*id*)) is an ordinary sentence that can be derived on the basis of reasons, RBL allows the weighing of reasons for and against application.

In the law, the fact that application of a rule is against its purpose is a reason not to apply the rule. A legal theory in RBL might therefore contain:

Valid(rule(purpose_rule,
application_against_purpose(rule(*id*)),
~applies(rule(*id*), *condition*, *conclusion*)))

Let us illustrate the use of this rule by means of an example borrowed from Fuller [1958]. There is a rule that forbids to sleep in the railway station, with as its purpose to prevent tramps from occupying the station as their place to spend the night. An old lady that wants to meet a friend at the station dozes off when the evening trains turns out to be retarded. Should the prohibition apply to this lady?

The following sentences describe the situation:

Valid(rule(sleeping_prohibition, sleep_in_railway_station(*act*), forbidden(*act*)))
Sleep_in_railway_station(lady's_act)
Application_against_purpose(rule(sleeping_prohibition))

In this situation we can derive:

Reason(sleep_in_railway_station(lady's_act),
applies(rule(sleeping_prohibition),
sleep_in_railway_station(lady's_act),
forbidden(lady's_act)), pro)

and

Reason(application_against_purpose(rule(sleeping_prohibition)),
applies(rule(sleeping_prohibition),
sleep_in_railway_station(lady's_act),
forbidden(lady's_act)), con)

Suppose that we also have the weighing information

Outweighs({application_against_purpose(rule(sleeping_prohibition))},
{sleep_in_railway_station(lady's_act)},
applies(rule(sleeping_prohibition),

[21] To make it clear that the reasons why rule x prevails over rule y is in conflict with the reasons why rule y prevails over rule x, the scheme should be added that Reason(prevails_over(rule(*x*), rule(*y*)), prevails_over(rule(*y*), rule(*x*)), con).

sleep_in_railway_station(lady's_act),
forbidden(lady's_act)))

On the basis of this additional information, we can derive

~Applies(rule(sleeping_prohibition), sleep_in_railway_station(lady's_act),
forbidden(lady's_act))

Because the rule called sleeping_prohibition is not applied, there is no reason why the old lady's sleeping was forbidden.

5. CHARACTERISTIC FEATURES OF REASON-BASED LOGIC

In this section we discuss a number of characteristics of RBL and globally compare RBL with other nonmonotonic logics. First we treat the situations in which the conclusion of a rule is not derivable, even though the conditions of the rule are satisfied. Section 5.2 deals with the role of consistency checking, both in RBL and other nonmonotonic logics. The comparison of arguments forms the topic of section 5.3. The section 5.4, finally, deals with some forms of reasoning about rules.

5.1. When does the conclusion of a rule not follow?

Normally, if the conditions of a valid rule are satisfied, the conclusion of the rule follows. Only additional information should be able to break the chain between the conditions and the conclusion of a rule. In this section we raise the question under which circumstances it is in RBL possible that the conditions of a valid rule are satisfied, but the conclusion of the rule can nevertheless not be derived.

The kind of situation we have in mind can as follows be specified in terms of the extension E of an RBL-theory:

- Valid(rule(*id*, *condition*, *conclusion*) ε E.
- *Facts* ε E, where *Facts* is an instance of one of the disjuncts of the formula *Condition* under some substitution σ (cf. **R2** in section 3.2).
- It is not the case that *Instance-of-conclusion* ε E, where *Instance-of-conclusion* is the instance of *Conclusion* under substitution σ.

This situation can occur for three reasons. The first possibility is that the rule applies and generates a reason for its conclusion, but that there is no information that the reasons for the rule conclusion outweigh the reasons against the rule conclusion. In fact, it can even be that the reasons against the rule conclusion outweigh the reasons for the rule conclusion. This possibility is as follows characterized in terms of the extension E:

- Reason(*facts*, *instance-of-conclusion*, pro) ε E.
- It is not the case that Outweighs(*reasons_pro*(*instance-of-conclusion*), *reasons_con*(*instance-of-conclusion*), *instance-of-conclusion*) ε E.

The second and third possibility have in common that the rule does not apply and consequently does not generate a reason for its conclusion. In theory the conclusion might still follow from the theory because of other reasons, but we will leave this possibility out of consideration.

One situation in which the conditions of a rule are satisfied and the rule nevertheless does not apply is when the facts that satisfy the rule conditions are a reason to apply the rule, but that there is no information that the reasons for the application of the rule conclusion outweigh the reasons against the application. This situation is analogous to the situation described above, with this important difference that in the present case the application of the rule and not the conclusion of the rule is at stake. This second possibility is as follows characterized in terms of the extension E:

- Applicable(rule(*id*), *facts*, *instance-of-conclusion*) ε E.
- Reason(*facts,* applies(rule(*id*), *facts*, *instance-of-conclusion)*, pro) ε E.
- It is not the case that
  Outweighs(reasons_pro(applies(rule(*id*), *facts, instance-of-conclusion*)),
  reasons_con(applies(rule(*id*), *facts, instance-of-conclusion*)),
  applies(rule(*id*)*, facts, instance-of-conclusion*)) ε E.

The second situation in which the conditions of a rule are satisfied and the rule nevertheless does not apply is when the application of the rule is excluded. In this case, the facts that satisfy the rule conditions do not even generate a reason to apply the rule. In theory the rule might still apply because of other reasons, but we will leave this possibility out of consideration. This third possibility in which the conclusion of a valid rule does not follow, although the rule conditions are satisfied, is as follows characterized in terms of the extension E:

- Excluded(rule(*id*)) ε E.
- It is not the case that Applicable(rule(*id*), *facts*, *instance-of-conclusion*) ε E.
- It is not the case that
  Reason(*facts*, applies(rule(*id*), *facts, instance-of-conclusion*), pro) ε E.

5.2. The role of consistency checking

Many nonmonotonic logics treat defeasible reasoning as a kind of reasoning with inconsistent knowledge.[22] We will globally characterize the way in which these logics deal with defeasible reasoning, necessarily skipping many details that make these logics different from each other.[23]

A theory T is considered as consisting of two parts (each of which can theoretically be empty), that is a consistent part of 'hard' knowledge and a part of defeasible knowledge in the form of so-called defaults. Let us call the hard knowledge K and the defeasible knowledge D. T = <K, D>.

If T as a whole is consistent, reasoning with T is dealt with by classical logic or in a very similar way. If T is inconsistent, a new consistent theory T' is generated by adding a subset D' of D to K, so that T' = <K, D'>. This new consistent theory T' is then dealt with by classical logic or similarly.

Often it will be possible to combine many different subsets of D with K so that the result is consistent. For each of these subsets there is a corresponding set of sentences that can be derived from them. These different sets of derivable sentences are called the extensions of T. Cf. figure 2.

---

22 Although the following description does not match all of these theories in all details, we think that the description captures the essence of the proposals made in e.g. Reiter 1980, McCarthy 1980, Moore 1985, Delgrande 1988, Poole 1988, Brewka 1990, Geffner and Pearl 1992, Sartor 1993 and Prakken 1993.

23 The strive to capture a number of theories in one abstract description made it necessary to force these theories in the same terminological framework, which means that the terms employed here are not necessarily those that were used in the original descriptions.

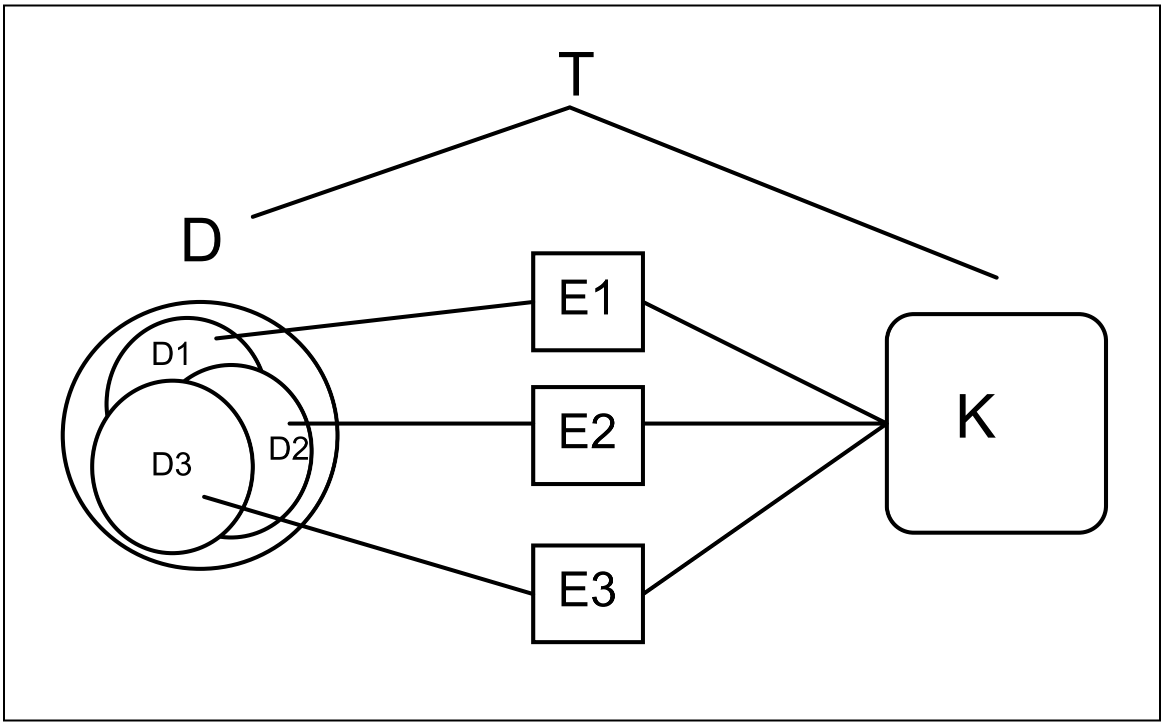

Figure 2. The parts and the extensions of a theory T.

A sentence can be derived from a theory T if this sentence occurs in one or all of the extension, or in all or one of a preferred subset of the extensions of T. We will deal with the topic of preferred subsets of extensions in section 5.3. For the present purposes it is important to notice that a consistency check is involved in the construction of every extension, and consequently in the determination of whether a sentence is derivable from T.

RBL does not use such consistency checks. The reason for this is that RBL does not consider rules as statements, and consequently does not consider a theory in which rules with incompatible conclusions can be applied as inconsistent. These rules only generate reasons that plead in different directions. To distinguish this situation from inconsistencies, we use the expression 'collision' to denote entities that 'work' into different directions, but which do not create an inconsistency. Colliding reasons do not generate an inconsistency.

If a rule is treated like a statement, the conclusion of the rule must follow if the rule is applied. This means that if rules with incompatible conclusions are applied, the result becomes inconsistent. Let us call a theory that contains rules with incompatible conclusions and with satisfied conditions a theory with colliding rules. There are in principle two possibilities to withhold a theory with colliding rules from being inconsistent. One possibility is to leave a number of rules out of consideration, so that the result does not contain colliding rules anymore. This option is chosen by the nonmonotonic logics that were described above.

The other option, that is chosen by RBL, is to 'weaken' the conclusion that follows from a rule, so that the results of rule application are not incompatible anymore. In the case of RBL this weakening consists of making the consequence of the application of a rule the existence of a *reason* for the rule conclusion.[24] The reasons that result from the application of colliding rules still have to be 'weighed' to determine which conclusion follows.

As was already discussed in section 5.1, weighing the reasons that are generated by colliding rules is not the only way in which RBL deals with colliding rules, although it is the way that most highlights the difference with other nonmonotonic logics. If a theory contains colliding rules, it will often be the case that not all of these rules are applied. There may be exclusionary reasons against application of some of these rules, and these exclusionary reasons will often

[24] An alternative form of weakening is to assign the conclusion of the rule a probability which is relativized to the rule conditions. If the probability of C, given A, is 0.5, and the probability of C, given B, is 0.01, this is not inconsistent. This form of weakening would, however, ask for a completely different interpretation of rules. Such an intepretation would certainly not do for most rules of law.

be generated by rules that collide with the rule that is excluded (cf. section 4.4b). Moreover, it is also possible that a rule is not applied, even though it is applicable (cf. section 4.5).

The reasons that have to be weighed are the result of the application of colliding rules. The rules that are actually applied will normally be a subset of the rules that are applicable. The applicable rules will, in turn, be a subset of the rules of which the conditions are satisfied, and these are in their turn a subset of the valid rules with incompatible conclusions. Cf. figure 3.

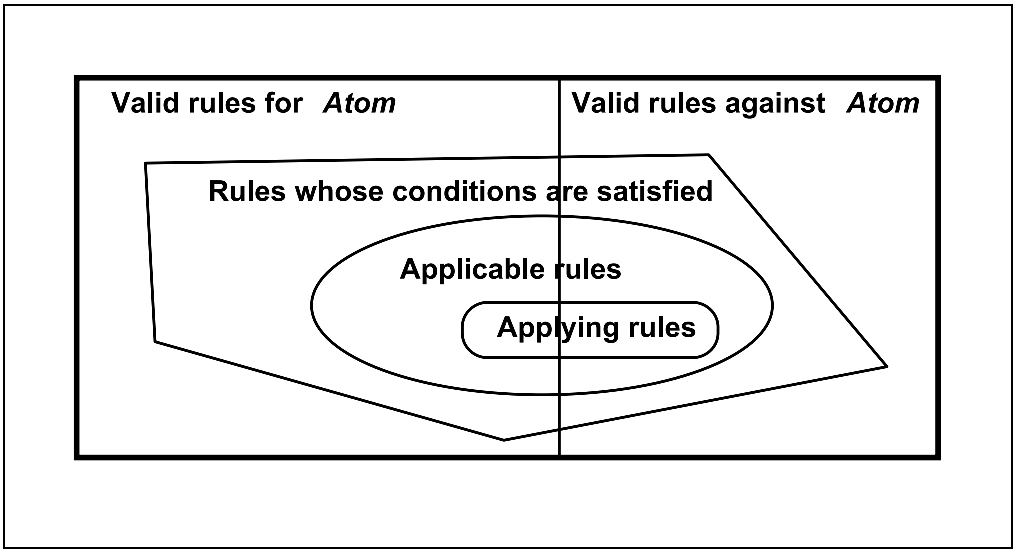

Figure 3. Colliding, applicable, and applying rules.

An important characteristic of RBL is that the 'problem' of colliding rules must completely be solved on the basis of domain knowledge. This knowledge must explicitly be represented in an RBL-theory. It determines whether a rule is excluded, whether there are reasons against its application in case the rule is applicable, and whether the reason generated by a rule that is applied also leads to the conclusion of the rule. RBL only provides handles for domain theories to deal with collisions. Dealing with rule collisions is taken away from the logic and placed into the hands of the domain theory. This heavily burdens the domain theory, but has the advantage that a richer domain theory can exclude counterintuitive conclusions that may obtain if a domain independent logic deals with the collision of rules.

Although in RBL colliding rules do not directly lead to inconsistency, inconsistency can still be the result of the interaction of facts and rules. For instance, the following set of premises is inconsistent in RBL:

P & Q
Valid(rule(1, p, ~q))

The counterpart of these sentences in, for example, normal default logic, would be consistent, because a consistency check would prevent the application of rule 1.

5.3. The comparison of arguments

Let us again consider the nonmonotonic logics described in the beginning of section 5.2. If a theory T has more than one extension, there is a question concerning the derivability of a sentence s if this sentence does occur in some, but not in all of these extensions. Is s derivable from T or not? In the so-called credulous approach to the multiple extension problem, a sentence is derivable from a theory if it occurs in at least one of its extensions. In the skeptical approach a sentence is only derivable from a theory if it occurs in all of its extensions.

Both the credulous and the skeptical approach are over-simplified. Often not all extensions of a theory are equally important. It seems unwarranted that a sentence is derivable if it only occurs in an unimportant extension of a theory, and it seems equally unwarranted if a sentence is not derivable only because it does not occur in some unimportant extension. This insight has led to approaches in which a subset of all the extensions is given a special status. A sentence is derivable from a theory if it occurs in all the preferred extensions of this theory.

To define the preferred extensions, often an ordering on the defaults of the theory is used. If an extension is based on a default that is, according to the ordering, less than all the defaults on which other extensions are based, this extension is not preferred. The ordering of the defaults can in its turn also be based on some criterion. In this connection specificity is a popular standard.

A refinement is still possible. To determine whether a sentence follows form a theory, it is not necessary to determine whether it occurs in all preferred *extensions* of this theory. Another possibility is to compare only the *sets of premises* from which a sentence is derived, on the basis of a preference relation. Such a set of premises is called the argument for this sentence. A sentence is derivable from a theory if the argument that leads to it is preferred to all arguments that lead to a conclusion that is inconsistent with the sentence.

Let us consider a example, in which we use the RBL notation for valid rules to denote defaults.

Valid(rule(murder, murderer(*x*), ought(punished(*x*))))
Valid(rule(youth, had_difficult_youth(*x*), ~ought(punished(*x*))))
Murderer(peter)
Had_difficult_youth(peter)
rule(murder) > rule(youth)

The last sentence indicates that in the case of conflict, the use of the default named murder is to be preferred to the use of the default named youth. On the basis of this theory, there is the argument

{Valid(rule(murder, murderer(*x*), ought(punished(*x*)))), Murderer(peter)}

that leads to the conclusion

Ought(punished(peter))

and the argument

{Valid(rule(youth, had_difficult_youth(*x*), ~ought(punished(*x*)))), Had_difficult_youth(peter)}

that leads to the conclusion

~Ought(punished(peter))

Because of the ordering on the defaults, the fist argument is preferred to the second, and the conclusion that is to be drawn from the theory as a whole is

Ought(punished(peter))

In RBL, a similar result would be obtained by replacing the sentence

rule(murder) > rule(youth)

by the weighing information

Outweighs({murderer(*x*)}, {had_difficult_youth(*x*)}, ought(punished(*x*)))

The weighing knowledge in RBL performs a role that is similar to the ordering on defaults. There are however, two important differences. One has to do with the order of steps in arguments, the other with the accrual of reasons. We discuss these differences in turn.

a. The order of steps in arguments

The comparison of arguments where arguments are considered as unstructured sets of premises runs the risks of overseeing that real life arguments are structured, and that this structure is relevant for defeat. An example can make this clear. Consider the theory:

Valid(rule(murder, murderer(*x*), ought(punished(*x*))))
Valid(rule(not-murder, ~murderer(*x*), ~ought(punished(*x*))))
Valid(rule(poverty, poor_youth(*x*), had_difficult_youth(*x*)))
Valid(rule(youth, had_difficult_youth(*x*), ~ought(punished(*x*))))
Valid(rule(proof_murder, motive_to_murder(*x*) & occasion_to_murder(*x*), murderer(*x*)))
Valid(rule(alibi_murder, alibi_murder(*x*), ~murderer(*x*)))
Poor_youth(peter)
Motive_to_murder(peter)
Occasion_to_murder(peter)
Alibi_murder(peter)
rule(murder) > rule(alibi_murder) > rule(proof_murder) > rule(youth) > rule(poverty) > rule(not_murder)

In this case there are three arguments to compare:

A1: {Valid(rule(murder, murderer(*x*), ought(punished(*x*)))),
Valid(rule(proof_murder, motive_to_murder(*x*) & occasion_to_murder(*x*), murderer(*x*))),
Motive_to_murder(peter),
Occasion_to_murder(peter)}
A2: {Valid(rule(not_murder, ~murderer(*x*), ~ought(punished(*x*)))),
Valid(rule(alibi_murder, alibi_murder(*x*), ~murderer(*x*))),
Alibi_murder(peter)}
A3: {Valid(rule(youth, had_difficult_youth(*x*), ~ought(punished(*x*)))),
Valid(rule(poverty, poor_youth(*x*), had_difficult_youth(*x*))),
Had_difficult_youth(peter)}

Only the first of these arguments leads to the conclusion that Peter ought to be punished; the latter two lead to the conclusion that Peter ought not to be punished.

If we take the order on the defaults into account, A2 is the weakest argument because of its use of the rule called not_murder. Therefore A2 cannot provide the conclusion. The second weakest argument is A3, because it uses the rule called poverty. Consequently, argument A1 is the preferred one, and the conclusion is that Peter ought to be punished.[25]

This is the wrong conclusion, because the 'real' argument should run as follows: Because Peter had an alibi, he did not commit the murder. (The rule called alibi_murder ranks higher than the rule called proof_murder.) If he did not commit the murder, he ought not to be punished. (The rule called not_murder ranks higher than the rule called murder).

The reason why the comparison of arguments goes wrong is that the comparison works with complete sets of arguments, neglecting the order of the steps in the argument. If a subargument at the beginning of an argument chain is rejected, the remainder of the argument should be rejected too. Comparison of arguments as complete sets of premises cannot take this into account.[26]

In RBL this problem does not occur, because the definition of the extension of a theory closely follows the rules of inference **R1** to **R5** (cf. section 3.2), and these rules take the order of the steps in an vargument into account. To see this, let us assume that we have the following weighing information, that corresponds to (part of) the above given order on the defaults:

Outweighs({alibi_murder(*x*)}, {motive_to_murder(*x*) & occasion_to_murder(*x*)}, murderer(*x*))
Outweighs({~murderer(*x*)}, {murderer(*x*)}, ought(punished(*x*)))

As a consequence of this weighing information, the extension of the theory contains

~Murderer(peter), and
~Ought(punished(peter)).

There are no reasons why the extension should contain

Murderer(peter)

and

Ought(punished(peter))

and because of the minimality condition on extensions (cf. section 3.3), the extension does not contain these sentences. In other words, RBL allows it to derive that Peter is not a murderer and should not be punished, and does not allow to derive that Peter is a murderer and should be punished.

b. The accrual of reasons

If the comparison of arguments is based on an ordering of the defaults that are used in these arguments, it is only possible to compare pairs of defaults. This leads to problems in the case of accrual of reasons. Let us again consider an example.

Valid(rule(murder, murderer(*x*), ought(punished(*x*))))
Valid(rule(youth, had_difficult_youth(*x*), ~ought(punished(*x*))))
Valid(rule(first_offender, first_offender(*x*), ~ought(punished(*x*))))
Murderer(peter)

---

[25] The example in this section presupposes a particular way of handling preferences. The example can be adapted to other ways of dealing with preferences.

[26] Pollock 1987, Simari and Loui (1992), Prakken 1993 and Vreeswijk 1993 propose ways to compare arguments which also take the hierarchical structure of arguments into account.

Had_difficult_youth(peter)
First_offender(peter)
rule(murder) > rule(youth) > rule(first_offender)

There are three arguments to be compared:

A1: {Valid(rule(murder, murderer(*x*), ought(punished(*x*)))), Murderer(peter)}
A2: {Valid(rule(youth, had_difficult_youth(*x*), ~ought(punished(*x*)))), Had_difficult_youth(peter)}
A3: {Valid(rule(first_offender, first_offender(*x*), ~ought(punished(*x*)))), First_offender(peter)}

Given the order on the defaults, A1 wins. This is the right conclusion given the information expressed in the theory. But suppose that we want a theory that holds that if somebody both had a difficult youth and is a first offender, he should not be punished, even though he is a murderer. This information cannot be expressed by means of an order on individual defaults. RBL offers the possibility to express the relevant information, with the correct consequences for the derivable conclusions:

Outweighs({had_difficult_youth(*x*) & first_offender(*x*)}, {murderer(*x*)}, ought(punished(*x*)))
Outweighs({murderer(*x*)}, {had_difficult_youth(*x*)}, ought(punished(*x*)))
Outweighs({murderer(*x*)}, {first_offender(*x*)}, ought(punished(*x*)))

Given this weighing information, Peter, being a murderer, ought to be punished if he is only a first offender, or only had a difficult youth, but he ought not to be punished if he both is a first offender and had a difficult youth.

This example illustrates the so-called accrual of reasons. Reasons that by themselves are not sufficient to outweigh other reasons can strengthen each other, so that their combination has the necessary weight. It is impossible to express the accrual of reasons by means of an ordering of defaults.[27]

5.4. Meta-reasoning about rules

In RBL a lot of information about rules is dealt with in the object language: whether rules are excluded, whether they are applicable or whether they apply, whether the reasons based on them are outweighed, all of these issues are dealt with on object language level. The important advantage of this is that RBL is very flexible in its use of rules. A disadvantage is that paradoxes of self-reference can occur (cf. the end of section 3.3).

An example of this flexibility is that RBL makes it possible to add rules to the domain theory which in another logic should be included in the rules of inference. For rules in RBL no analog of Modus Tollens holds. However, in RBL it is possible to include an analog of Modus Tollens in the domain theory. It would look like this:

Valid(rule(*id1*, *condition*, *conclusion*)) → Valid(rule(*id2*, *~conclusion*, *~condition*))

Of course this rule can also be incorporated as a *defeasible* RBL rule, instead of as a material implication[28]:

Valid(rule(modus_tollens, valid(rule(*id1*, *condition*, *conclusion*)),
valid(rule(*id2*, *~conclusion*, *~condition*))))

It is also possible to restrict the class of rules for which Modus Tollens holds by constraints that are specified by reference to the identifiers of the rules:

Valid(rule(modus_tollens,
valid(rule(*id1*, *condition*, *conclusion*)) & ~rule_of_law(rule(*id1*)),
valid(rule(*id2*, *~conclusion*, *~condition*))))

---

27 It is possible to express the accrual of reason by means of an ordering on *sets* of defaults. An attempt in this direction is made in Brewka and Gordon 1994.

Another possibility would be to use defaults with more complex conditions that distinguish different sets of reasons, and a preference on the basis of specificity. E.g.

Valid(rule(100, murderer(x) & first_offender(x), ~ought(punished(x))))
Valid(rule(101, murderer(x) & had_difficult_youth(x), ~ought(punished(x))))
Valid(rule(102, murderer(x) & had_difficult_youth(x) & first_offender(x), ought(punished(x))))
rule(102) > rule(101)
rule(102) > rule(100)

This is, however, comparable to, and just as undesirable as, the explicit mentioning of all the exceptions to a defeasible rule in its condition, as in 'If something is a bird, but not a penguin, and not an ostrich, and not ... , then it can fly'. Cf. also the criticism of the theory of hidden rule conditions in section 1.2, which deals with a related issue.

28 In this case it must hold that both *condition* and *conclusion* are literals of RBL.

Another example of how the flexibility of RBL can be used is the following definition of a form of specificity:

((A → B) & ~(B → A)) → More_specific(rule(*id1*, a, *conclusion1*), rule(*id2*, b, *conclusion2*))

This rule can be combined with the rule called lex_specialis (cf. section 4.4b), that says a rule with more specific conditions prevails over a rule with less specific conditions.

Finally, it is also possible to add abstract weighing information to an RBL theory:

Valid(rule(strengthening_reasons,
outweighs(*reasons1*, *reasons2*, *conclusion*) & (*reasons1* ⊆ *reasons3*),
outweighs(*reasons3*, *reasons2*, *conclusion*)))[29]

If a set of reasons outweighs another set, this is a reason why a superset of the first set will also outweigh the last set.

Valid(rule(weakening_reasons,
outweighs(*reasons1*, *reasons2*, *conclusion*) & (*reasons3* ⊆ *reasons2*),
outweighs(*reasons1*, *reasons3*, *conclusion*)))

If a set of reasons outweighs another set, this is a reason why a subset of the last set will also be outweighed by the first set.

6. CONCLUSION

There is no clear border between rules of inference that belong to logic and rules that belong to domain theories (Quine 1986, p. 95 f.). This does not mean that there is no difference at all, but only that the transition is gradual. For instance, some would consider 'Squares are rectangular' to be a logical truth, while others would maintain that its necessary truth depends on a domain theory governing the inclusion of squares in the class of rectangles.[30] This presumption about the gradual transition between logical rules and domain rules has influenced the design of RBL.

The consequence is that RBL has been kept minimal in several respects. Only those reasoning rules that appear to be valid in every domain have been included in the logic. As a consequence, no general principles for the defeat of arguments, such as consistency checking or specificity, have been incorporated in RBL. RBL only provides handles that can be used by domain theories to deal with colliding rules.

In an RBL theory reasoning with rules and the defeat of arguments must be guided by *explicit knowledge*. The reasons for this are first that explicit knowledge can be *adapted* to the domain at hand, and second that the consequences of explicitly represented knowledge can better be *controlled* than general implicit principles. Counterintuitive consequences are less likely and can easily be overcome by adapting the knowledge.

In the end this approach means that many characteristics of the ways of reasoning are determined by the domains concerned, and should be established by empirical research. Meanwhile we have made some choices based on our knowledge of the legal domain where many 'natural' logical rules are not accepted, and on the heuristic that everything that can be left out of the logic should be left out.

This approach has led to the absence from the logic of, amongst others, consistency checking, preferring the more specific rule, and definitions of what counts as more specific. All of these can be included in a theory about reasoning for specific domains, but in that case the corresponding rules should be included in the theory as domain knowledge. We have given several examples of how this can be accomplished in RBL.

Another consequence is that the inference rules of RBL employ propositions, the truth conditions of which are not (completely) defined in the logic itself, but in domain theories as well. Examples of these propositions are Excluded(*x*), Applicable(*x*), Applies(*x*), Reason(*x*, *y*, pro/con), and Outweighs(*x*, *y*, *z*). The resulting fusion of logic and domain theory helps to complement the relative weakness of RBL proper.

---

[29] Here ⊆ is a function symbol that represents set inclusion. Additional axioms are needed to give it the expected properties.

[30] Logical truths are often considered to rely on meaning relations between *logical* terms, such as 'all', 'some', 'and', 'or', etc. But how are logical terms to be distinguished from non-logical terms such as square and rectangle? And what about non-truth-functional terms such as 'necessary', 'possible', 'ought', 'forbidden', and 'permitted'?

ACKNOWLEDGMENT

The research for this article was partly financed by the Foundation for Knowledge Based Systems (SKBS), which seeks to improve the level of expertise in the Netherlands in the field of knowledge based systems, and to promote the transfer of knowledge in this field between universities and business companies.

REFERENCES

Alexy, R. (1979). Zum Begriff des Rechtprinzips, *Rechtstheorie*, Beiheft 1, p. 59-87.

Baker, G.P. (1977). Defeasibility and Meaning. P.M.S. Hacker and J. Raz (eds.) *Law, Morality and Society*. Clarendon Press, Oxford, p. 26-57.

Brewka, G. (1990). Cumulative Default Logic: in defense of nonmonotonic inference rules. *Artificial Intelligence* 50, pp. 183-205.

Brewka, G. and Th. Gordon (1994). How to Buy a Porsche: An Approach to Defeasible Decision Making. Extended Abstract submitted to the AAAI-workshop on Computational Dialectics.

Delgrande, J. (1988). An approach to default reasoning based on a first-order conditional logic: revised report. *Artificial Intelligence 36,* p; 63-90.

Dworkin, R. (1978). *Taking Rights Seriously*, 2nd ed., Duckworth, London.

Fuller, L.L. (1958). Positivism and Fidelity to law: A Reply to Professor Hart, *Harvard Law Review* 71, p. 630-672.

Geffner, H. and J. Pearl (1992). Conditional entailment: bridging two approaches to default reasoning, *Artificial Intelligence* 53, p.209-244.

Hage, J.C. (1993). Monological Reason Based Logic. *Proceedings of the fourth International Conference on Law and Artificial Intelligence*, ACM-press, Amsterdam, p. 30-39.

Hart, H.L.A. (1949). The Ascription of Responsibility and Rights. *Proc. of the Aristotelian Society*, 49, p. 171-194.

Lukaszewicz, W. (1990). *Non-monotonic reasoning. Formalization of commonsense reasoning*. Ellis Horwood, New York.

McCarthy, J. (1980). Circumscription-a form of non-monotonic reasoning. *Artificial Intelligence* 13, p. 89-116.

Moore, R.C. (1985). Semantical considerations on nonmonotonic logic. *Artificial Intelligence* vol. 25, p. 75-94.

Pollock, J.L. (1987). Defeasible Reasoning. *Cognitive Science* 11, p. 481-518.

Poole, D. (1985). On the comparison of theories: Preferring the most specific explanation. *Proceedings IJCAI-1985*, p. 144-147.

Poole, D. (1988). A logical framework for default reasoning. *Artificial Intelligence* 36, p. 27-47.

Prakken, H. (1993). *Logical tools for modelling legal argument*, Ph.-D.-thesis Amsterdam.

Quine, W.V. (1986). *Philosophy of Logic*, 2nd ed., Harvard University Press, Cambridge Mass, London.

Raz, J. (1975). *Practical Reason and Norms*, Hutchinson, London.

Reiter, R. (1980). A logic for default reasoning. *Artificial Intelligence* 13, p. 81-132.

Sartor (1993). A Simple Computatinal Model for Nonmonotonic and Adversarial legal Reasoning. *Proceedings of the fourth International Conference on Law and Artificial Intelligence*, ACM-press, Amsterdam, p. 192-201.

Schwartz, S.P. ed. (1977). *Naming, Necessity, and Natural Kinds*. Cornell University Press, Ithaca.

Simari, G.R. and R.P. Loui (1992). A mathematical treatment of defeasible reasoning and its implementation. *Artificial Intelligence* 53, p.125-157.

Smith, E.E. (1990). Categorization. D.N. Osherson and E.E. Smith (eds.) *Thinking. An Invitation to Cognitive Science, vol. 3.* Mit Press, Cambridge Mass. and London, p. 33-54.

Toulmin, S.E. (1953). *The philosophy of science; an introduction*, Hutchinson & Co., London.

Verheij, H.B. (1994). Reason Based Logic and legal knowledge representation. *Proceedings of the Fourth National Conference on Law, Computers and Artificial Intelligence* (eds. I. Carr and A. Narayanan). University of Exeter.

Vreeswijk, G.A.W. (1993). *Studies in Defeasible Argumentation*. Ph.-D.-thesis, Amsterdam.