Procedural Arguments

Arno R. Lodder

Universiteit Maastricht, Faculty of Law - Department of Metajuridica
P.O. Box 616, 6200 MD Maastricht
arno.lodder@metajur.unimaas.nl
http://www.metajur.unimaas.nl/~arno

Abstract

Although many authors claim that argumentation is a process, in most models the procedural side of argumentation is restricted to the definition of a procedural framework that is able to compare arguments.

This paper elaborates on the procedural side of argumentation. A distinction is made between structural and procedural arguments.

Structural arguments are characterized by their specific structure. The elements recognized in the analysis of structural arguments are used to illustrate the attack, justification and construction of structural arguments in a procedural model. Subsequently the question is addressed whether structural arguments necessarily have underlying rules.

Procedural arguments are characterized by the procedure in which they are adduced. It is argued that there is more to argumentation than structure. Therefore the notion procedural arguments is introduced and elaborated. Especially in legal practice, e.g. court decisions, non-structural, procedural arguments play an important role. It is argued that procedural models of legal reasoning should take procedural arguments into account.

1. Introduction

If the product of argumentation is studied, general structures of support or justification between sets of premises and conclusions are defined. The product of argumentation is static, just like a 'check mate' position in chess. The procedure of argumentation, on the other hand, is dynamic, just as a strategy path that leads to check mate. For chess players it is interesting to know under what circumstances a position is checkmate. Even more interesting is what procedure must be followed to check mate the opponent.

Nowadays it is generally accepted in the AI and Law community that the study of legal argumentation should not only concern the product of argumentation, but that the process or procedure is an important subject too. Only five years ago Hage, Span and Lodder (1992) could open their contribution to the JURIX-conference with the following sentence:

> **"Although most legal reasoning takes place in the context of legal disputes in which two parties argue their positions in a dialogue with the other party, formal models of legal reasoning usually resemble monological proofs."**

Things have radically changed since then. At this moment this characterization of formal models of legal reasoning can hardly be defended anymore. The number of papers that reported on formal models of legal reasoning taking a procedural, in particular a dialogical approach is impressive (e.g., Gordon 1993, Loui et al. 1993,

Hage et al. 1994, Lodder and Herczog 1995, Farley and Freeman 1995, Prakken and Sartor 1996, Kowalski and Toni 1996, Nitta and Shibasaki 1997).

In argumentative dialogs arguments can appear in different ways. In this paper a distinction is made between structural and procedural arguments. The former are characterized by their specific structure, the latter by the procedure in which they are adduced. Most procedural models represent only an exchange of structural arguments. This paper analyses structural arguments and argues that there is more to argumentation than structural arguments, namely procedural arguments.

The paper is structured as follows. First, the terminology as used in the paper, viz. statements, structural and procedural arguments is introduced. Then follows an analysis of structural arguments. The elements recognized in this analysis are used to explain how arguments can be supported, how arguments can be attacked, and how the elements can be used to build structural arguments during a procedure. Subsequently it is argued that not necessarily all elements of structural arguments have to be present in the procedure, and moreover, that structural arguments are not always necessary or even desirable. This paves the way for the introduction of procedural arguments. After a characterization of the dialogical model DiaLaw (Lodder and Herczog 1995), that allows to model both procedural and structural arguments, related literature is briefly discussed.

2. Terminology

There is a difference between a statement and an argument. Suppose we have two statements: 1. Jeffrey is punishable, and; 2. Jeffrey has intentionally killed someone. An argument shows a relation between statements. Arguments indicate that a statement supports or justifies another statement. An example is the argument that Jeffrey is punishable, because he has intentionally killed someone. I call this type of argument a structural argument, because the reason it is an argument is its specific structure. Although a structural argument can be regarded as a special type of statement, namely one that relates two other statements, for reasons of clarity structural arguments are excluded if statements are mentioned.

Most models of argumentation do have structural arguments. For instance, in the Pleadings Game (Gordon 1995) arguments are represented as argument(A p), where A is a set of formulas that supports the formula p. In Reason-Based Logic (e.g. Verheij 1996, Hage 1997), as well as in DiaLaw (Lodder and Herczog 1995)[1], there are basically two structural argument types. The first type is the reason: reason(Cond, Concl)

This formula says that Cond is a reason for Concl. The following formulas are examples of reasons:

```
reason(killer(oj), punish(oj)),
reason(falsely_accused(oj), ~punish(oj)).
```

The fact that OJ is a killer is a reason for punishing him (or a reason against not punishing him). The fact that OJ is falsely accused is a reason to not punish him (or a reason against punishing him).

The second structural argument type is the relation 'outweighs'. This structural argument indicates that concerning some formula a set of reasons pro outweighs a set of reasons con.[2] This second structural argument does not play an important role in the present paper.

In a procedure, also statements can become arguments. They are arguments not because of their structure, but because of the procedure in which they have

1) An adapted and extended version of DiaLaw can be obtained by the author. The reasons presented here are from this new version, so a little different from the reasons of (Lodder and Herczog 1995).

been put forward. I call this type of arguments procedural arguments. A statement is a procedural argument, if it is used in a procedure to support or justify. Before procedural arguments are further explained in section 5, first structural arguments are analyzed.

3. Structural arguments

Suppose that Jeffrey has killed someone. In that case an obvious structural argument is that Jeffrey is punishable, because he has killed someone. This can be represented as DiaLaw's structural argument reason:

reason(killer(jeffrey), punishable(jeffrey)).

Sometimes a structural argument is acceptable for anyone, but in cases it is not there must be an opportunity to either attack or question the argument. In the discussion of the possible attacks (3.1) and defenses (3.2) of arguments, as well in the discussion of the process of building arguments (3.3), different elements of a structural argument play a role. This paper distinguishes four elements of a structural argument (see for different, well-known analyses of arguments, e.g. Toulmin (1958) and Van Eemeren and Grootendorst (1987)):

1. the conclusion;
2. the condition;
3. the underlying rule in the present case, or, the application of the rule;
4. the underlying rule in general, or, the validity of the rule.

The four elements can be easily recognized in a representation in First Order Predicate Logic of the argument about Jeffrey's killing: [3]

(4) killer(X)→ punishable(X)

---

(3) killer(jeffrey)→ punishable(jeffrey)
(2) killer(jeffrey)

---

(1) therefore punishable(jeffrey)

The numbers refer to the different elements, so, the conclusion (1), the condition (2), the rule in this particular case, or the application of the rule (3), and the rule in general or the validity of the rule (4).

In sections 3.1-3.3 is discussed what role the four elements play in justifying an argument, attacking an argument, and building an argument, respectively. For all the three processes it is important that the procedural model is open: new information, e.g. statements, arguments, must be allowed (cf., e.g. Scholten 1974, p. 75f, Lodder 1996, 1998). If the set of premises or arguments were fixed, a consequence is that some premises/arguments are indisputable. However, most knowledge is disputable, especially in the law.

---

2) The formula, outweighs ({Condpro$_1$, ..., Condpro$_m$}, {Condcon$_1$, ..., Condcon$_n$}, Concl) (m≥1,n≥0), means the following. The set {Condpro$_1$, ..., Condpro$_m$} originates from reason(Condpro$_i$, Concl); the set {Condcon$_1$, ..., Condcon$_n$} originates from reason(Condcon$_i$, ˜Concl). The set of reasons pro must contain at least one term (m≥1), the set of reasons con may be empty (n≥0).

3) A similar scheme (Walton and Krabbe 1995, p.180), more explicitlymentioning rules, is:
(4) (Rule: n) $K_x \rightarrow P_x$
(2) $K_{jeffrey}$
(3) Rule n applies to the present case

---

(1) Therefore $P_{jeffrey}$

3.1. Justifying structural arguments

A structural argument can be justified with all but the first element: the conclusion. Since the argument itself aims at justifying the conclusion, it is not strange that the conclusion cannot justify the argument.

We consider again the argument that Jeffrey is punishable, because he has killed someone. Three elements remain that may justify the argument.

First, the condition of the argument, that is, the fact that Jeffrey has killed someone can be put forward. A discussion can start about whether Jeffrey really has killed someone.

Second, it can be claimed that a rule about punishable killers applies, or that this rule is true for this particular case.

Finally, the general rule that killers are punishable can be adduced. A discussion can start about whether the rule is valid or the rule in general is true.

If the three ways to justify the argument have been used successfully, a firm basis for the justification of the argument is given. In DiaLaw the elements can be used to justify an argument, i.e., a reason. However, it is not possible to force the opponent to accept the reason. There are only a few situations in which an opponent can be forced. [4]

3.2. Attacking structural arguments

All elements can play a role in the attack of structural arguments, so there are four different ways to attack. As an example, again the following argument is used:

```
reason(killer(jeffrey), punishable(jeffrey)).
```

First, the conclusion of the argument can be denied:

```
~punishable(jeffrey).
```

An attacking argument supporting the denial of the conclusion is commonly referred to as a rebutter. An example of a rebutting argument is that Jeffrey is not punishable, because he is a diplomat. In DiaLaw this argument would be represented as:

```
reason(diplomat(jeffrey), ~punishable(jeffrey)).
```

Second, the condition of the argument can be denied:

```
~killer(jeffrey).
```

This denial can be supported by the claim that it was not Jeffrey who killed, but his twin brother Bert.

Third, the specific rule underlying the argument can be denied:

```
~applies(rule(killer(jeffrey), punishable(jeffrey))).
```

The argument supporting this denial is usually referred to as an undercutter. For instance, the killer is not punishable, because he acted in self-defense. In DiaLaw this argument would be represented as:

```
reason(self_defense(jeffrey),
```

4) An example of forced acceptance is the following situation. If the opponent accepts that the condition of the rule is satisfied, and that the rule is valid, he is forced to accept that there is a reason to apply the rule.

~applies(rule(killer(jeffrey), punishable(jeffrey)))).

Finally, the general rule underlying the argument can be denied:

~valid(rule(killer(Person), punishable(Person))).

An argument supporting this denial could be that the rule is not valid, since it was not announced properly. In DiaLaw this argument would be represented as:

reason(~announced_properly(rule(killer(Person),
punishable(Person))),
~valid(rule(killer(Person), punishable(Person)))).

3.3. Building structural arguments

In the previous two sections it was discussed how different elements can be used to either attack or justify an argument. Instead of adducing the elements in order to attack or justify a structural argument, the elements can also be used without there being a structural argument yet. In the latter case the argument is constructed after one or more elements have been put forward.

Only if it is clear what justifies a particular statement, a structural argument justifying the statement can be put forward. However, often a particular statement may seem justified, although it is not clear why. For instance, a lawyer intuitively solves a case, and then tries to construct justifying reasons. This process of constructing justifying reasons can be modeled as a dialog in which the proponent puts forward one of the elements, and the opponent accepts, attacks, or asks to justify this element. After one or more elements are accepted, it may be possible to construct a supporting argument. Note that this structural argument can still be attacked or that a justification of this argument may be asked. However, on some points there will be agreement already, namely about the elements that were accepted before the structural argument was constructed.

Building the argument that Jeffrey is punishable comes down to the following. All elements of arguments can be used. The discussion starts with the conclusion (the intuitive solution to the case). Any statement can be claimed and attacked; some statements may turn out to correspond to conditions of a relevant argument. Any (valid) rule can be claimed and attacked; some valid rules may turn out to be relevant for the conclusion, that is the intuitive solution of the case. Finally, the application of any rule can be claimed and attacked; some applied rules may turn out to be relevant for the conclusion. After one or more of the elements are adduced, the structural argument can be constructed.

The process that ends in the construction of a structural argument is opposite to the processes of justification and attack. In the latter processes the elements of an argument are used after the structural argument is adduced; in the former process the elements of an argument are used before the structural argument is adduced. The picture shows the relation between the three processes.

4. Structural arguments without underlying rules

In the Pleadings Game the backing of the rule, comparable to the validity of the rule in DiaLaw, always is part of the argument. The same is true for Reason-Based Logic. Each reason has a valid rule underlying it. In my opinion not all arguments (reasons) necessarily have underlying rules.

In order to make my point clear, I discuss three ways reasons and rules can be related. First, a reason can be based on the application of an already existing rule. Second, a reason can be based on the application of a new created rule. Finally, a reason can be based on no rule application at all.

4.1. Reasons based on an existing rule

An obvious example of an existing rule is one that is laid down in the statutes. But not only statutory rules, also rules taken from case law belong to the category of already existing rules. Take for example the legal rule 'if someone kills another person intentionally, then he is liable for second degree murder'. In case Oedipus killed his father intentionally, this fact is a justifying reason for the statement that Oedipus is liable for second-degree murder. Although we are dealing with an existing rule, terms in legal rules sometimes need to be interpreted, like in this case intentionally.

Suppose Oedipus intentionally shot three feet above his mother's head, and that accidentally this shot fatally wounded his father. Since in case law intentionally is also explained as 'knowingly and willingly accepting the ... chance that ...', it may be concluded here that Oedipus is liable for second-degree murder. Since the interpretation is based on case law, the conclusion is justified on the basis of an existing rule. However, the first time intentionally was interpreted as above by the Dutch Supreme Court[5], the justifying reason was not based on an existing rule.

4.2. Reasons based on a newly created rule

In the Dutch Road Traffic Act it is regulated that in case of an accident between a motorist and other non-motoring road users, e.g., pedestrians or bikers, the motorist has to pay for the damages, except if he acted in force majeure[6]. In the beginning of the nineties the Dutch Supreme Court decided that in case a child not older than fifteen is involved in the accident, the motorist always has to pay the damages, except if the child acted recklessly or intentionally. This narrowing of force-majeure can be considered as a rule, although the court did not explicitly present it as such.

In a later case the Dutch Supreme Court had to decide about an accident in which an elderly man was involved. The Dutch Supreme Court explicitly formulated a new rule: it is only fair that a motorist has to pay at least 50% of the damages in case he cannot demonstrate that he acted in force-majeure, and the mistake of the pedestrian or biker did not constitute intent or recklessness.

4.3. Reasons based on no rule at all

Finally, reasons can be put forward, based on no rule at all.

In the above-mentioned case in which a new rule was created, another decision dealt with the question whether the rule about young children should also be applied to elderly people. The Dutch Supreme Court decided that the rule was not applicable, based on the following reasons:

5) Decided on November 9, 1954 (NJ 1955,55).

6) The concerning rule was at that time laid down in section 31 of the Road Traffic Act (art. 31 Wegenverkeerswet). In the current Road Traffic Act the regarding section is 185 (art. 185 Wegenverkeerswet 1994).

1. the arguments used to define the rule for young people are less striking for elderly people;
2. law knows no fixed age criterion for elderly people (for young people it does);
3. elderly people are less recognizable.

This collection of reasons constituted the decision of the Dutch Supreme Court. No rule was applied here, the decision was justified by these reasons alone. An objection may be that there is a rule after all, which can be created after the decision is made. First, this objection does not change the fact that the reasons were in the first place not based on a rule, and therefore is an example of an argument without an underlying rule. Second, if reference is made to this case, probably the collection of reasons will be mentioned, and not some –'rather artificial'– rule.[7] An alternative representation of this verdict is proposed in the next section on procedural arguments.

Summarizing, in a procedure structural arguments are related to rules as follows: structural arguments can have either an existing or newly created underlying rule, or structural arguments can have no underlying rule at all.

5. Statements as procedural arguments

In section 3 it was described how structural arguments can be justified, attacked, and built in a procedure. In the previous section it was claimed that not all structural arguments have underlying rules. In this section it is claimed that not all arguments are structural arguments, and the notion of a procedural argument is elaborated.

Structural arguments can be characterized as logical arguments, procedural arguments as psychological arguments. The latter arguments are not logically compelling, but are nevertheless arguments in a procedure. The idea of procedural arguments is inspired by the work of Stevenson. He claims that the relation between reasons, adduced for or against a normative statement, and the normative statement itself is not a logical relation but rather only a psychological one (Stevenson 1979, p. 113).

A structural argument indicates an explicit relation between two statements: A, therefore B. In procedures, relations between statements can remain implicit. If a statement B is adduced in order to support or justify an other statement A, the statement B is a procedural argument, that is, the statement B is a procedural argument for the player who claimed B. For the opponent the statement B only becomes a procedural argument if he accepts the statement A because the statement B has been put forward.

In general, a procedural argument is a statement that contributes to the acceptance of another statement, without a structural argument being used. It is a statement that itself or in combination with other non-structural arguments, leads to the acceptance of the other statement.[8]

A trivial procedural argument is a statement that is accepted immediately after it has been put forward, so without any structural or procedural argument being adduced to support it. Such a statement is accepted, because of its intrinsic cogency.

Non-trivial procedural arguments are statements that contribute to the accep-

7) I do not claim that it is never possible to model the ratio decendi of a case as a rule (see on this topic Loui and Norman (1995) and Prakken and Sartor (1997)). For instance, the interpretation of intentionally as given in section 4.1 is easily modeled as a rule.

8) Because of the leveled structure of dialogs in DiaLaw, the concept procedural arguments can easily be defined. Structural arguments can only be claimed one level deeper (after a question) than the statements they are arguments for. A possible definition of structural arguments is the following. If a level n statement is accepted, and on the level n+1 were no structural arguments claimed, then all statements claimed and not withdrawn on the level n+1 are procedural arguments.

tance of other statements. I will give an example that illustrates the difference between a structural and a procedural argument.

punishable(jeffrey)	punishable(jeffrey)
?	?
killer(jeffrey)	reason(killer(jeffrey), punishable(jeffrey))

In the left side mini-dialog a procedural argument is adduced, in the right side mini-dialog a structural argument. The relation between the two statements in the left dialog remains implicit in structure, but is explicit in the procedure. The statement that Jeffrey killed someone else is a procedural argument that supports the statement that Jeffrey is punishable.

The importance of allowing procedural arguments is primarily that if structural arguments cannot be adduced, it may still be possible to put forward arguments, namely procedural arguments. This means that in order to support a statement other statements (procedural arguments) can be claimed, without the need to explicitly indicate relations (structural arguments). For instance, in one of the above examples the Dutch Supreme Court used three statements to justify why a particular rule about young people should not be applied to elderly people (see section 4.3.). It was claimed that not all reasons are the result of rule application. The representation of the decision with procedural arguments goes even beyond the representation that was proposed in section 4.3. The structural arguments (reasons) are replaced by procedural arguments (statements).

Another point in favor of procedural arguments is that they fit in with actual legal reasoning. Not only academics or rhetorically well equipped attorneys, but also courts reason with procedural arguments. The justification of a decision is often of the following type. A couple of statements is adduced and then the magic phrase "in view of the above" (in Dutch: gezien het bovenstaande) connects the conclusion with the previous statements. An appropriate characterization of these statements is to qualify them as procedural arguments.[9]

Finally, procedural arguments are necessary if structural arguments are justified, attacked or constructed, because the elements (condition, conclusion, etc.) are not of the structural argument type.

Summarizing, it is not always possible or necessary to use structural arguments in order to justify a statement in a procedure, or in terms of a dialog, to convince the opponent. Often procedural arguments are a better means. In many cases procedural arguments are the only possible arguments, because structural arguments cannot, or not immediately, be put forward.

6. DiaLaw, allowing structural and procedural arguments

A procedural model of argumentation must allow both structural and procedural arguments. Structural arguments to explicitly indicate relations between statements, and procedural arguments merely to convince without indicating an explicit relation. The importance of modeling structural and procedural arguments was explained in the previous sections. It is not claimed that it is inferior to use structural arguments, just that it is not always possible or necessary.

The model DiaLaw is addressed only briefly. For precise details of the game I refer to (Lodder and Herczog 1995).

DiaLaw is a two person dialog game, in which both players make moves alter-

9) Another example of the way courts argue is a decision by the Arnhem Court of Justice (NJ 1994, 107). The court decided that the noise of cocks annoyed a neighbor. The ground for this decision was the following. Some facts were mentioned (so some statements were put forward) and these facts were connected to the conclusion in the following way: "considered in mutual connection and coherence it is sufficiently clear that "(in Dutch: bezien in onderling verband en samenhang is voldoende komen vast te staan dat").

nately. The purpose of the game is to justify statements in a dialog. The sentences put forward by one player, become justified because of acceptation by the other player. The dialogue game is a rhetorical procedure (See e.g. Perelman and Olbrechts-Tyteca (1971), Witteveen (1988)). Characteristic for such a procedure is that there is no fixed outcome, the procedure is non-deterministic. By presenting arguments, either structural or procedural ones, each party tries to draw the outcome in his direction, but the final result cannot be determined in advance.

The key idea of DiaLaw is that justification of a statement can solely be based on agreement of participants in a dialog. This agreement can be reached by:

1. immediate acceptance (trivial procedural arguments);
2. acceptance after only procedural arguments were adduced;
3. acceptance after (also) structural arguments have been used.

The question may arise whether anything can be justified, solely by acceptance of opponents in a dialog. Acceptation in a dialog as criterion for justification seems rather shallow. However, because an external criterion of justification, i.e. a criterion outside the procedure, is impossible to define, it is essential to define the criterion inside the procedure, i.e. a procedural criterion. (see e.g. Hage et al. 1994, Lodder 1996). Regarding all interests, procedural and structural arguments lead to justified statements.

7. Related research on the discussed topics

Of all topics discussed in this paper related research is briefly evaluated.

7.1. Justifying structural arguments

Justification of arguments is part of the Pleadings Game (Gordon 1995). Each structural argument has a condition part, and a part in which it is expressed that the rule applied in the argument is valid (called backing, after Toulmin). To my knowledge, besides DiaLaw and the Pleadings Game, there are no other procedural models of argumentation that model the justification of arguments.

7.2. Attacking structural arguments

Except for attacking the underlying rule in general, procedural models allow all other types of attack (e.g. Prakken and Sartor (1996), Verheij (1996), Nitta and Shibasaki (1997)).

An interesting addition to extending the ways to attack an argument is the notion of the rationale (Loui and Norman 1995). A rationale explains why a particular conclusion of a case is correct or why a particular rule is assumed to be valid. Rationales are comparable to the backing of warrants in Toulmin's terminology. One of the five rationale types is the compression rationale. An argument of the form 'a therefore b', can be the result of two chained rules, e.g. 'a therefore c' and 'c therefore b'. If the opponent knows that c is not the case, he cannot directly attack the original argument 'a therefore b'. The use of a c-rationale will help in this situation. Informally the use of the rationale can be described as "you introduced your argument '...', but it actually comes down to '...'" Once the rationale is introduced, an argument supporting the negation of c can be put forward as a counterargument.

7.3. Building structural arguments

No model allows to build an argument procedurally as it was described in this paper. Closest to the present description are Verheij (1996) and Vreeswijk (1993). They build, in a procedure, so-called argument trees.

7.4. Structural arguments without underlying rules

In most procedural models arguments are based on rules (e.g. Gordon 1995, Prakken and Sartor 1996). The Zeno Argumentation framework of Gordon and Karacapilidis (1997) has arguments without underlying rules. To my knowledge, DiaLaw is the only model in which arguments can have underlying rules, but do not have them necessarily.

7.5. Procedural arguments

Most models only model structural arguments. Gordon's Pleadings Game (1995) models trivial procedural arguments: statements that are part of a structural argument, e.g. the condition, can be conceded. However, it is not possible to support statements by other than structural arguments. The idea of procedural arguments is implicitly present in (Gordon and Karacapilidis 1997).

The idea of procedural arguments can be recognized in the work of Hamblin (1971) and MacKenzie (1979). Their games know no structural arguments.[10] In the game the players alternately adduce a statement. After a statement is introduced both players become committed to this statement. If the other player disagrees, he has to let it know explicitly. In case a player does disagree, the player who introduced the statement will try to convince his opponent by adducing other statements, i.e. procedural arguments.

8. Conclusion

To summarize the results of this paper, a distinction has been made between structural and procedural arguments, and it was illustrated why both argument types are important in a procedural model of legal argumentation. Most procedural models use structural arguments (e.g. Gordon (1995), Prakken and Sartor (1996), Nitta and Shibasaki (1997)). However, not all features described in this paper are represented in these models. Some models implicitly use procedural arguments (e.g. MacKenzie 1979). In DiaLaw both aspects of argumentation are represented.

There are many procedural models; all have their own merits. Although it depends on the purpose of a model what should be contained in it, I think that all aspects of arguments described in this paper are important and necessary in general. However, I do not intend to disqualify procedural models that lack one or more of the features described in this paper.

I hope not only that this paper contributes to the theory of legal argumentation, but also that the recognition of procedural arguments helps to bridge the gap between theory and practice (Oskamp et al. 1995, Verheij et al. 1997). The more theory and practise are tuned, the more probable it is that at some time in the future sophisticated systems will be developed that are easy to use by lawyers. However, the road is still long and winding.

10) Some statements closely resemble structural arguments. For instance, by the move resolution demand an opponent is asked to solve an inconsistency in his commitment, e.g. 'Resolve whether (If P then Q and P) and not Q'.

Acknowledgements

I want to thank Bart Verheij, Jaap Hage and two anonymous referees for their suggestions. Numerous discussions with the first two about our work have been valuable.

9. References

Alexy, R. (1989). A theory of legal argumentation. Clarendon Press, Oxford.

Eemeren, F.H. van, R. Grootendorst and T. Kruiger (1987). Handbook of Argumentation Theory. Foris Publications, Dordrecht.

Gordon, Thomas F. (1993/1995). The Pleadings Game - An artificial intelligence model of procedural justice. PHD-thesis, Technische Hochschule Darmstadt, Germany, 1993. Also published by Kluwer, The Netherlands, 1995.

Gordon, Thomas F. and Nikos Karacapilidis (1997). The Zeno Argumentation Framework. Proceedings of the Sixth International Conference on Artificial Intelligence and Law, pp. 10-18. ACM, New York.

Hage, J.C., G.P.J. Span, and A.R. Lodder (1992), A Dialogical Model of Legal Reasoning, in C.A.F.M. Grütters et al. (eds.), Legal Knowledge Based Systems: Information Technology and Law, JURIX '92,Koninklijke Vermande, Lelystad.

Hage, J.C, R. Leenes, A.R. Lodder (1994). Hard cases: a procedural approach. Artificial Intelligence and Law. 2: 113-167.

Hage, Jaap (1997). Reasoning with rules. An essay on legal reasoning and its underlying logic. Kluwer, Dordrecht. Hamblin, C.L. (1971). Mathematical models of dialogue. Theoria 2, pp. 130-155.

Kowalski and Toni (1996). Abstract Argumentation. Artificial Intelligence and Law. 4: 275-296.

Lodder, Arno R, Aim ée Herczog. (1995) DiaLaw - A dialogical framework for modeling legal reasoning. Proceedings of the fifth International Conference on Artificial Intelligence and Law, ACM, New York, pp. 146-155.

Lodder, Arno R. (1996). The ideas behind DiaLaw. Proceedings of the Fifth National/ First European Conference on Law, Computers and Artificial Intelligenc, EUCLID, Exeter, pp. 93-103.

Lodder, Arno R. (1998). From law to DiaLaw: why legal justification should be modeled as a dialog. To appear in Information and CommunicationTechnology Law.

Loui, R.P. and J. Norman (1995). Rationales and argument moves. Artificial Intelligence and Law. 3: 159-189.

Loui, Ronald. P., Jeff Norman, Jon Olson and Andrew Merill (1993). A design for reasoning with policies, precedents and rationales. Proceedings of the Fourth International Conference on Artificial Intelligence and Law, pp. 202-211. ACM, New York.

Mackenzie, J.D., Question-Begging in non-cumulative systems, in: Journal of Philosophical Logic, 8 (1979) 117-133.

Nitta, Katsumi, and Masato Shibasaki (1997). Defeasible reasoning in Japanese criminal jurisprudence. Artificial Intelligence and Law. 5: 139-159.

Oskamp, A., M. Tragter, C. Groendijk (1995). AI and Law: What about the future? Artificial Intelligence and Law. 3. pp.209-215.

Perelman, Ch. and L. Olbrechts-Tyteca (1971). The New Rhetoric A Treatise on Argumention. University of Notre Dame Press, London.

Prakken, H. and G. Sartor (1996). A dialectical model of assessing in conflicting arguments in legal reasoning. Artificial Intelligence and Law. 4: 331-368.

Prakken, H. and G. Sartor (1997). Reasoning with precedents in a dialogue game. Proceedings of the Sixth International Conference on Artificial Intelligence and Law, pp. 1-9. ACM, New York.

Scholten, Paul (1974). Asser-Scholten (Algemeen deel), Tjeenk Willink, Zwolle.

Stevenson (1979), Ethics and language. Yale University Press, New Haven and London, Reprint of the 1944 ed.

Toulmin, S.E. (1958). The uses of argument. University press, Cambridge.

Verheij, Bart (1996). Rules, Reasons, Arguments. Formal studies of argumentation and defeat. Dissertation Universiteit Maastricht.

Verheij, Bart, Jaap Hage and Arno R. Lodder (1997). Logical tools for legal argument: a practical assessment in the domain of tort. Proceedings of the Sixth International Conference on Artificial Intelligence and Law, pp. 243-249. ACM, New York.

Vreeswijk, G.A.W. (1993). Studies in Defeasible Argumentation. Doctoral thesis, Free University, Amsterdam.

Walton Douglas N. and Erik C.W. Krabbe (1995). Commitment in dialogue. State University of New York Press, Albany.

Witteveen, W.J. (1988). De retoriek in het recht, W.E.J. Tjeenk Willink, Zwolle.