Legal Knowledge and Information Systems
M. Palmirani (Ed.)

doi:10.3233/978-1-61499-935-5-11

Lessons from Implementing Factors with Magnitude

Trevor BENCH-CAPON, Katie ATKINSON
Department of Computer Science, The University of Liverpool, UK

Abstract. We discuss the lessons learned from implementing a CATO style system using factors with magnitude. In particular we identify that giving factors magnitudes enables a diversity of reasoning styles and arguments. We distinguish a variety of ways in which factors combine to determine abstract factors. We discuss several different roles for values. Finally we identify the additional value related information required to produce a working program: thresholds and weights as well as a simple preference ordering.

Keywords. legal case based reasoning, dimensions, factors, values.

1. Introduction

Reasoning with legal cases has always been a central concern of AI and Law. Much of the investigation of this topic has been based on the pioneering work of Rissland and Ashley's HYPO system [23], [6]. Subsequent development of HYPO's ideas is described in [8]. While HYPO used *dimensions*, aspects of a case which were described using a range with one end favouring the plaintiff and the other end favouring the defendant, most subsequent work has used the simpler notion of *factors* to represent cases. Factors, as introduced in CATO [5], are stereotypical fact patterns which are legally significant and which can be seen in Boolean terms, as either present or absent [8]. For a discussion of the differences between dimensions and factors, see [24]. Using the simplification enabled by representing cases as a set of Booleans, a good understanding of this kind of reasoning has been developed, as formalised in [18] and [21]. It remains a simplification, however, and many researchers have often felt that it would become necessary to return to dimensions in order to give a full account of reasoning with legal cases (e.g. [11]). Both HYPO and CATO supported argumentation in US Trade Secret law and took as their starting point section 757 of the *Restatement of Torts*:

> "Some factors to be considered in determining whether given information is one's trade secret are: the *extent* to which the information is known outside of his business; the *extent* to which it is known by employees and others involved in his business; the *extent* of measures taken by him to guard the secrecy of the information; the *value* of the information to him and to his competitors; the

amount of effort or money expended by him in developing the information; and the *ease or difficulty* with which the information could be properly acquired or duplicated by others"[1].

In CATO, we find several factors such as *InfoObtainableElsewhere*, *InfoKnownToCompetitors*, *SecurityMeasures*, *CompetitiveAdvantage* and *InfoReverseEngineerable*, all of which clearly originate in this section of the *Restatement*, but which are represented as Booleans, despite the emphasis on the *extent* to which these notions are satisfied in the *Restatement*. From this it is clear that concepts which are treated as Boolean factors in CATO, should not really be Boolean but have *magnitudes* (extents, amounts, degrees of difficulty and the like), and it should be for the court (rather than the analyst as in CATO) to decide whether the extent is sufficient, given the particular facts of the case. The simplification from dimensions to factors has proved useful, greatly facilitating understanding of several aspects of legal CBR, but the time has now come to return to the original notions of dimensions and factors with magnitude and much attention in recent AI and Law research has focussed on how this can best be done.

The persistent need for dimensions was argued in [11] and related to argumentation schemes for legal CBR in [20]. Formal representations for exploring a logic of precedent for dimensions and factors with magnitude have recently been proposed by Horty [17] and Rigoni [22]. In [7] a method for representing case law using factors with magnitude based on the methodology of [2] was described and an implementation of Aleven's CATO analysis using this representation as formalised in [1] was demonstrated at the COMMA 2018 conference [10]. In this paper we draw out the lessons learned from that implementation which can be used to guide future research on the topic.

Section 2 describes the representation used. Section 3 discusses the various ways in which factors are combined. Section 4 relates the implementation to the statement types of [3]. Section 5 discusses the additional types of reasoning enabled by using magnitudes and gives a hypothetical example case and variations to illustrate these different kinds of reasoning. Section 6 offers some concluding remarks.

2. Representation

The knowledge in the implemented program is represented following the ANGELIC methodology [2], specifically using the 2-regular structure of [1], in which statements (issues, intermediate concepts and factors) are represented as nodes and non-leaf nodes have exactly two children (see Table 1). Essentially this is used as the design documentation on which the implementation of [10] is based.

Each non-leaf node is associated with acceptance conditions, expressed in terms of its two children. Like [9] the statements have *degrees of acceptance*, in range [0,1]: 0 representing total rejection and 1 total acceptance. Some statements are genuinely Boolean and they use only 0 and 1. The acceptance conditions are then a set of conditions for attributing particular degrees of acceptance, together

[1] http://www.lrdc.pitt.edu/ashley/RESTATEM.HTM *italics ours.*

Table 1. 2-Regular ADF for CATO.

Parent	Child 1	Child 2
TradeSecretMisappropriation	SecretMisappropriated	EmployeeSoleDeveloper
SecretMisappropriated	InfoMiasappropriated	Info Trade Secret
Info Miasappropriated	BreachOfConfidence	ImproperMeans
Info Trade Secret	InfoValuable	EffortstoMaintainSecrecy
InfoValuable	InfoUseful	KnownOrAvailable
EffortstoMaintainSecrecy	AdequateEfforts	SecurityFailures
InfoUseful	CompetitiveAdvantage	UniqueProduct
KnownOrAvailable	Known	Available
Known	KnownOutside	Limitations
InfoAvailableElsewhere	InfoReverseEngineerable	InfoObtainableElsewhere
KnownOutside	InfoKnownToCompetitors	DisclosureInPublicForum
Limitations	UniqueProduct	MaintainSecrecyOutsiders
MaintainSecrecyOutsiders	SecretsDisclosedOutsiders	OutsiderDisclosuresRestricted
AdequateEfforts	SecurityMeasures	MaintainSecrecyDefendant
SecurityFailures	Reckless	WaiverOfConfidentiality
MaintainSecrecyDefendant	AgreedNotToDisclose	DisclosureInNegotiations
Reckless	NoSecurityMeasures	SecretsDisclosedOutsiders
Disclosed	DisclosureInPublicForum	MaintainSecrecyOutsiders
BreachOfConfidence	InfoUsed	ConfidentialRelationship
ImproperMeans	QuestionableMeans*	LegitimatelyObtainable
ConfidentialRelationship	NoticeofConfidentiality	ConfidentialityAgreement
ConfidentialityAgreement	AgreedNotToDisclose	WaiverOfConfidentiality
InfoUsed	GaveHelp	InfoIndependentlyGenerated
NoticeofConfidentiality	ValidAgreement	AwareConfidential
QuestionableMeans	IllegalMethods	DEfOKMethods
LegitimatelyObtainable	InfoKnownorAvailable	InfoAvailableElsewhere
GaveHelp	IdenticalProducts	GaveAdvantage
GaveAdvantage	BroughtTools	CompetitiveAdvantage
IllegalMethods	Criminal	Dubious
DefOKMethods	DisclosureInNegotiation	DefendantDiscovered
Criminal	BribeEmployee	InvasiveTechniques
Dubious	RestrictedMaterialsUsed	Deception
DefendantDiscovered	InfoIndependentlyGenerated	InfoReverseEngineered
AwareConfidential	RestrictedMaterialsUsed	KnewInfoConfidential
ValidAgreement	AgreementMade	AgreementInForce
AgreementMade	AgreedNotToDisclose	OutsiderDisclosuresRestricted
AgreementInForce	AgreementNotSpecific	WaiverOfConfidentiality
DisclosureInNegotiations		
BribeEmployee		
EmployeeSoleDeveloper		
AgreedNotToDisclose		
AgreementNotSpecific		
SecurityMeasures	Magnitude	
BroughtTools	Magnitude	
CompetitiveAdvantage		
SecretsDisclosedOutsiders	Magnitude	
VerticalKnowledge		
OutsiderDisclosuresRestricted		
NoncompetitionAgreement		
RestrictedMaterialsUsed		
UniqueProduct		
InfoReverseEngineerable		
InfoIndependentlyGenerated	Magnitude	
IdenticalProducts	Magnitude	
NoSecurityMeasures	Magnitude	
InfoKnownToCompetitors		
KnewInfoConfidential		
InvasiveTechniques		
WaiverOfConfidentiality		
InfoObtainableElsewhere	Magnitude	
InfoReverseEngineered		
Deception		
DisclosureInPublicForum		

with a default degree. Not all factors need to be ascribed magnitudes. Of the 26 factors in [5], 17 turned out to be adequately modelled as Boolean, but 9 were modelled with magnitudes. Even in HYPO, 10 of the 13 dimensions can adequately be modelled as Booleans [8], and Rigoni [22] recognises that there are these two types of factor and accordingly represents cases with a set of Booleans factors, as well as a set of factors with magnitude. Our program was implemented using SWI Prolog (Windows version).

It may be for some nodes (especially those with disjunctive children) that the acceptance conditions do not give an answer. For example there may be no evidence of competitive advantage offered, and hence there are no relevant facts for either side. Equally, there may be no mention of any disclosures. In such cases a default must be used. This is chosen to reflect the burden of proof. For example, it is generally uncontested that there was some competitive advantage from using the information, and this is usually simply accepted: thus it defaults to the plaintiff. Here the onus is on the defendant to show that the information provided no competitive advantage. In contrast, the existence of a confidential relationship must be demonstrated by the plaintiff, and so the relevant factors default to the defendant. In this way, the defaults can be used to assign the burden of proof, as revealed in the precedent cases.

3. Combining Factors

Examination of the structure used to represent the domain knowledge reveals that although each non-leaf node has two children, these children play a variety of roles for the different nodes. Consider for example the top of the tree, which represents the issues identified in the *Restatement*, which form the "logical model" used in [14]. At this level, we are dealing mainly with issues, such as *was the information misappropriated?* and *was the information a Trade Secret?*. These are Boolean (as discussed in [3]), found either for the plaintiff or the defendant, and the children are there either because the parent requires the satisfaction of multiple conditions (e.g. *SecretMisappropriated*), or because there are several conditions which enable the parent to be satisfied (e.g. *InfoMiasappropriated*). The very top level, however, represents a general rule with an exception: *TradeSecretMisappropriation* is normally satisfied if *SecretMisappropriated* but has an exception: if the case has the factor *EmployeeSoleDeveloper*, then *TradeSecretMisappropriation* is not satisfied. This represents a preference: the presence of factor *EmployeeSoleDeveloper* casts doubt on the ownership of the information and is taken to outweigh the considerations leading to *TradeSecretMisappropriation*: if the courts had decided otherwise in the relevant precedents, the exception would not appear.

As well as such preference based exceptions, we find exceptions which are motivated not by preferences and precedents, but by the very meanings of the terms. Consider *MaintainSecrecyOutsiders*. This is normally not satisfied, as one would expect, if we have *SecretsDisclosedOutsiders*. But there is an exception if we also have *OutsiderDisclosuresRestricted*. This is not a matter of preference: by making any disclosures to outsiders subject to restrictions, the plaintiff must be regarded as having made efforts to maintain secrecy with respect to outsiders, as a consequence of the meanings of the words involved.

Another kind of node is where a balance is struck between two competing factors [19]. This is illustrated by the node *ImproperMeans*, which requires that a balance be struck between the information having been obtained by *QuestionableMeans* and being *LegitimatelyObtainable*. It is, of course, possible that information which was obtainable legitimately was in fact obtained using questionable means. Thus the extent to which questionable means were used has to be weighed against the ease with which the information could have been obtained legitimately. This involves weighting the values concerned: do we want to encourage enterprise or strict adherence to ethical principles? In practice there has been a preference given to finding for the plaintiff if questionable means have been used, but this cannot be seen as absolute. If the information had been readily available on, say, Wikipedia, then some mild deception in obtaining the secret might be overlooked. In order to represent this relationship we ascribe weights to the values of the children involved. The values are taken from [15], which have also been used in later work such as [1]. An alternative set of values and ways of handling balances and trade offs using Boolean factors, can be found in [16].

Finally we have nodes at which conversion from children with magnitude to Boolean parents takes place. Here we are considering whether the extent of satisfaction represented by the children is *sufficient* to allow the parent to be *considered* satisfied. To achieve this, each value is associated with a threshold. Note that the thresholds can be set independently, since they are only required to be used in this way, not for comparison with one another. These thresholds receive a good deal of emphasis in the formal approaches of both [17] and [22]. The former divides the range into two factors, one a pro-plaintiff and one pro-defendant, with the point of division determined by precedents. Rigoni divides the range into a number of factors with a precedent determined "switching point" where the factors cease to be pro-plaintiff and become pro-defendant. These points correspond to our thresholds.

Acceptance conditions can thus take a variety of forms, depending on whether the children have magnitude, and how they are combined. Our program [10] uses the following acceptance condition types:

- **Conjunctions**: Used to provide multiple conditions all of which must be satisfied for the acceptability of a node. One or both conditions may have magnitude (represent degrees of acceptability). As in Fuzzy Logic [25], the minimum of the individual conjuncts is ascribed to the conjunction as a whole, enabling uniform treatment of Booleans and factors with magnitude.
- **Disjunctions**: Used to provide multiple conditions at least one of which must be satisfied for the acceptability of a node. One or both conditions may have magnitude (represent degrees of acceptability). As in Fuzzy Logic, the maximum of the individual disjuncts is ascribed to the disjunction.
- **Preference Based Exceptions**: Used to represent exceptions based on precedents expressing a preference between values. Preferences are based on values, taken from [15]. Should the value preference be changed, the exception disappears, since it is no longer sufficient to defeat the general case.
- **Definition Based Exceptions**: Used to represent exceptions based on precedents relying on word meanings. The exception depends on the meanings of the terms involved: no preference is required.

- **Comparison with a threshold**: This converts factors with magnitudes to Booleans. The result of conjoining (or disjoining) the children is compared to a threshold, and 1 or 0 is returned accordingly.
- **Balancing two factors using weights**. This is used to strike a balance between two features. It uses weights based on the values associated with the children and the extents to which the children are satisfied. This is similar to the method used in [15] to handle comparison between factors which may promote values to different extents.

In the program there were 37 nodes, of which 19 were disjunctions and 4 conjunctions. We had 4 preference based exceptions, 2 definitional exceptions and 7 nodes employing thresholds. 1 node expressed a trade off.

3.1. Values and Their Roles

What is particularly interesting about the above characterisation of the ways in which factors can combine, is the clarification it gives to the roles of values. Originally, as in [12], factors all promoted some value, and conflicts were resolved according to a value structure applicable to the whole domain. This uniformity was, however, shown to be inappropriate in [26], where it was shown that values could motivate not only whole rules (and the preferences between them), but also the inclusion of particular terms in the antecedents of rules, so as to enable the value to be given proper consideration. We now see that the picture is a little more complicated. The role of values is certainly to ensure that the various concerns they represent are properly considered. But their use to establish preferences is limited to the relatively few rules in which a balance must be struck to express a trade off, or a preference requires an exception to a general rule. In the first case we use weights established by precedents in the manner of [15], whereas in the second the preference is in the existence of the exception. A third role is to establish thresholds for determining whether a factor is satisfied to a sufficient extent. These thresholds do not use a preference ordering, since they are considered independently: they are, however, justified in terms of precedents[2] and may be set higher or lower to reflect the switching points manifest in the precedents. Thus, not only do values play several roles, but preferences between them are limited to a few nodes, and need not be consistent across the whole structure. This makes the reasoning more akin to Branting's reasoning with portions of precedent [13] than the holistic view taken by CATO and [12].

4. Statement Types

In [3] a number of different statement types were identified. Obviously the verdict must be Boolean: the court must decide for one party or the other. As we saw from the Restatement of Torts, the base level factors can have magnitude. The

[2]Some factors, like *SecretsDisclosedOutsiders* have a natural mapping to numbers, while others do not. For the latter, however an ordering on fact situations can be established (as with SecurityMeasures in [6]) and this order reflected in the magnitudes assigned.

implementation starts with the base level factors as input, so at some point the factors with magnitude have to be transformed to a Boolean. This must be done at, or before, the issues are reached.

The extents ascribed to the base level factors are decided by the court, and courts may disagree. Thus we find parts of the opinions discussing whether the lower court had correctly determined, for example, the extent to which the information was reverse engineerable, or was available elsewhere[3]. As we propagate the magnitudes assigned to base level factors up the tree, at some point the parent factor will need to become a Boolean, since issues are held to favour either the plaintiff, or the defendant. This means that if we have a branch ending in one or more base level factors with magnitude, at some point a threshold will need to be applied. This transition will not always occur at the border between abstract factors and issues, since some of the abstract factors, like some of the base level factors, may themselves be Boolean. However, the thresholds must be used to transform the factors with magnitude to Booleans when, or before, the issue level is reached. The use of thresholds is illustrated in the example given in section 5.

Note that there are two distinct points for the court to address: the *extent* to which the base level factor is satisfied, and the *threshold* that must be reached if the abstract factor is to be satisfied. If we do not allow factors to have magnitudes, these two points are conflated and both are decided by the analyst representing the cases: whereas they should be transparent and subject to explicit argumentation. They should kept distinct and both aspects decided by the court. In particular we often find minority opinions expressing disagreement as to the extent to which a factor is satisfied and, separately, as to whether the degree to which a factor is satisfied is sufficient to resolve the issue for a given party.

Transforming factors with magnitudes to Booleans is the primary role of thresholds, and it is essential. Examination of our program [10] reveals the importance of thresholds: they determine whether a factor is satisfied to a sufficient extent to enable an exception to be implied: without magnitudes any extent whatsoever could be considered sufficient. In the current implementation the threshold is dependent only on the value to which it relates, and the same threshold is used for all such tests relating to a given value. It remains possible, however, that different thresholds should be used to determine which party is favoured by an issue, and whether an exception should be applied. We leave this for future investigation, which will involve a careful analysis of actual opinions.

5. Reasoning with Factors with Magnitude

With the Boolean factors of CATO we can challenge a decision only by adding or removing a factor, or by changing a preference. With magnitudes we have the further options of:

[3] Of course, courts do not assign numbers to these extents, but they do order them, and this ordering can be mapped to numbers for computation. For example, Mr Justice Stevens says in *California v Carney* "It is perfectly obvious that the citizen has a much greater expectation of privacy concerning the interior of a mobile home than of a piece of luggage such as a footlocker". It is, however, clear that the majority disagreed with Stevens when the mobile home was in use as a vehicle. This can be reflected in the magnitudes assigned in the different opinions.

- Increasing the degree of presence of a factor
- Decreasing the degree of presence of a factor
- Raising the threshold for factors relating to a particular value
- Lowering the threshold for factors relating to a particular value
- Adjusting the relative weights of factors

This provides a number of different ways in which decisions can be contested, whereas CATO allowed only for different preferences between the factors involved, and approaches such as [12] reduced all disputes to disagreements about value preferences.

We illustrate these options with the following hypothetical Trade Secrets case. Plaintiff in US produced a widget. This was a distinctive product, but something similar was manufactured in China (uniqueProduct, but less than 1). Drawings of the widget were kept in an unlocked drawer in the plaintiff's office (securityMeasures but less than 1). The defendant was in the plaintiff's office and was left alone. He searched the desk and looked at the drawings and photographed them on his phone (invasiveTechniques). Defendant claimed that the drawings confirmed his view that the product was reverse engineerable (infoReverseEngineerable but less than 1). Defendant also claimed that the information could have been obtained from the Chinese company (infoObtainableElsewhere but less than 1). Suppose we start with some initial parameters and facts (only non-zero base level factors are given). Relevant values are legitimate means (lm), questionable means (qm) and material worth (mw). Note that two values used in [15] (reasonable efforts and confidential agreement) do not need weights or thresholds since they appear only in exceptions:

```
weight(qm,1).  % Questionable methods and
weight(lm,1).  % Legitimate methods have equal weights
% Thresholds are all neutral between parties
threshold(lm,0.5). threshold(re,0.5). threshold(mw,0.5).
invasiveTechniques(1).
securityMeasures(0.6). % Security measures somewhat lax,
                       % but intended to keep info secret
uniqueProduct(0.8).    % Similar product, but not in US
infoReverseEngineerable(0.3). % Product considered rather
                              % hard to reverse  engineer
infoObtainableElsewhere(0.4). % Obtainable from Chinese firm
```

These facts will find for the plaintiff. We can now suggest arguments for modifying them:

- Perhaps the reverse engineerability should be considered more straightforward (after all the information had been independently developed in China). Increasing the magnitude to 0.6 (or greater: precision is not crucial) will enable a finding for the defendant
- Perhaps the threshold for reverse engineerability is too low: if the information is really readily ascertainable, why steal it? Raising the threshold to 0.8 will find again for the plaintiff.

- The security measures were rather poor: the drawer was not even locked. Decreasing the magnitude to 0.4 (or lower) will find for the defendant.
- But perhaps inventors should be given more protection: the defendant should not be rifling through the plaintiff's drawers. Lowering the threshold to 0.3 will restore the case for the plaintiff.
- Perhaps the balance between legitimate and questionable means is wrong: perhaps "all's fair in love, war and business". Raising the weight given to legitimate methods to 3 will find for the defendant.

6. Concluding Remarks

In this paper we have consolidated our current understanding of reasoning with legal cases using factors with magnitudes, by drawing out lessons we have learned by implementing the approach in the standard domain for factor based systems, US Trade Secrets as modelled in HYPO and CATO. In particular we have learned:

- That magnitudes are important to reflect the reasoning found in opinions. In particular they enable a variety of arguments which go beyond simply preferring one set of factors or values to another. These additional types of arguments have been identified in section 5. This requires that each value is given a threshold to represent the switching points of [22].
- We have confirmed the importance of the "switching points" as identified by the formal approaches of [17] and [22]. However, we have also identified another kind of argument not catered for in the papers: namely trade offs as discussed in [7] and [16]. This requires that values also be assigned weights to be used to strike an appropriate balance in the trade offs. Thus we require two sets of value related parameters: thresholds and weights.
- The key reason for values is to enable due consideration of various aspects the law is meant to address. We have identified three separate roles for values: establishing thresholds, motivating weights and justifying preference based exceptions. However, we have shown that values affect relatively few nodes (about one third in our particular application).
- We have shown that children in the factor hierarchy (ADF) do not contribute homogeneously to the acceptability of their parents: we have identified six different ways in which children are combined.

In sum, we have identified a diversity of types of reasoning, and arguments, required when considering legal cases, moving beyond the rather uniform reasoning found in previous factor and value based approaches. It shows the different ways in which factors combine, the roles played by values, and various argumentative options not available in a Boolean setting. It shows what is (e.g. thresholds), and what is not (e.g. trade offs), captured in the formalisations of [17] and [22]. We intend to use magnitudes in our on-going work with legal firms [4] and to use these lessons to inform and improve the planned future application of our approach to further domains.

References

[1] L Al-Abdulkarim, K Atkinson, and T Bench-Capon. Factors, issues and values: Revisiting reasoning with cases. In *Proceedings of the 15th ICAIL*, pages 3–12. ACM, 2015.

[2] L Al-Abdulkarim, K Atkinson, and T Bench-Capon. A methodology for designing systems to reason with legal cases using ADFs. *AI and Law*, 24(1):1–49, 2016.

[3] L Al-Abdulkarim, K Atkinson, and T Bench-Capon. Statement types in legal argument. In *Proceedings of JURIX 2016*, pages 3–12. IOS Press, 2016.

[4] L Al-Abdulkarim, K Atkinson, T Bench-Capon, S Whittle, R Williams, and C Wolfenden. Noise Induced Hearing Loss: An application of the Angelic methodology. In *Proceedings of JURIX 2017*, pages 79–88, 2017.

[5] V. Aleven. *Teaching case-based argumentation through a model and examples*. PhD thesis, University of Pittsburgh, 1997.

[6] K Ashley. *Modeling legal arguments: Reasoning with cases and hypotheticals*. MIT press, Cambridge, Mass., 1990.

[7] K Atkinson and T Bench-Capon. Dimensions and values for reasoning with legal cases. *Tech Report ULCS 17-004, Department of Computer Science, U of Liverpool*, 2017.

[8] T Bench-Capon. HYPO's legacy: introduction to the virtual special issue. *Artificial Intelligence and Law*, 25(2):205–250, 2017.

[9] T Bench-Capon and K Atkinson. Dimensions and values for legal CBR. In *Proceedings of JURIX 2017*, pages 27–32, 2017.

[10] T Bench-Capon and K Atkinson. Implementing factors with magnitude. In *Proceedings of COMMA 2018*, pages 449–450, 2018.

[11] T Bench-Capon and E Rissland. Back to the future: Dimensions revisited. In *Proceedings of JURIX 2001*, pages 41–52. IOS Press, 2001.

[12] T Bench-Capon and G Sartor. A model of legal reasoning with cases incorporating theories and values. *Artificial Intelligence*, 150(1-2):97–143, 2003.

[13] L Karl Branting. Reasoning with portions of precedents. In *Proceedings of the 3rd ICAIL*, pages 145–154. ACM, 1991.

[14] S Bruninghaus and K Ashley. Predicting outcomes of case based legal arguments. In *Proceedings of the 9th ICAIL*, pages 233–242. ACM, 2003.

[15] A Chorley and T Bench-Capon. An empirical investigation of reasoning with legal cases through theory construction and application. *AI and Law*, 13(3):323–371, 2005.

[16] Matthias Grabmair. *Modeling Purposive Legal Argumentation and Case Outcome Prediction using Argument Schemes in the Value Judgment Formalism*. PhD thesis, University of Pittsburgh, 2016.

[17] J Horty. Reasoning with dimensions and magnitudes. In *Proceedings of the 16th ICAIL*, pages 109–118. ACM, 2017.

[18] J Horty and T Bench-Capon. A factor-based definition of precedential constraint. *AI and Law*, 20(2):181–214, 2012.

[19] M Lauritsen. On balance. *Artificial Intelligence and Law*, 23(1):23–42, 2015.

[20] H Prakken, A Wyner, T Bench-Capon, and K Atkinson. A formalization of argumentation schemes for legal case-based reasoning in ASPIC+. *Journal of Logic and Computation*, 25(5):1141–1166, 2015.

[21] A Rigoni. An improved factor based approach to precedential constraint. *Artificial Intelligence and Law*, 23(2):133–160, 2015.

[22] Adam Rigoni. Representing dimensions within the reason model of precedent. *Artificial Intelligence and Law*, 26(1):1–22, 2018.

[23] E Rissland and K Ashley. A case-based system for trade secrets law. In *Proceedings of the 1st ICAIL*, pages 60–66. ACM, 1987.

[24] E Rissland and K Ashley. A note on dimensions and factors. *Artificial Intelligence and Law*, 10(1-3):65–77, 2002.

[25] L Zadeh. Fuzzy logic and approximate reasoning. *Synthese*, 30(3-4):407–428, 1975.

[26] T Zurek and M Araszkiewicz. Modeling teleological interpretation. In *Proceedings of the 14th ICAIL*, pages 160–168. ACM, 2013.