

An empirical investigation of reasoning with legal cases through theory construction and application

ALISON CHORLEY and TREVOR BENCH-CAPON

Department of Computer Science, The University of Liverpool, Liverpool, UK

E-mail: A.H.Chorley@csc.liv.ac.uk

Abstract. In recent years several proposals to view reasoning with legal cases as theory construction have been advanced. The most detailed of these is that of Bench-Capon and Sartor, which uses facts, rules, values and preferences to build a theory designed to explain the decisions in a set of cases. In this paper we describe CATE (CAse Theory Editor), a tool intended to support the construction of theories as described by Bench-Capon and Sartor, and which produces executable code corresponding to a theory. CATE has been used in a series of experiments intended to explore a number of issues relating to such theories, including how the theories should be constructed, how sets of values should be compared, and the representation of cases using structured values as opposed to factors.

Key words: Case Based Reasoning, factors and dimensions, legal knowledge based systems, legal reasoning, theory construction

1. Introduction

The importance of cases in legal reasoning has been recognised throughout the development of AI and Law. Even approaches which took formalisation of legislation as their starting point, e.g. Sergot et al. (1986), rapidly came to realise that crucial questions of the interpretation and application of terms found in the legislation could be answered only by reference to cases (e.g. Bench-Capon 1991b). Cases, despite some differences in the ways in which they are used, are of considerable importance in Civil Law jurisdictions as well as Common Law jurisdictions (McCormick and Summers 1997; Ashley 2004). Given this centrality of cases, a good understanding of their contribution and use is vital. In this paper we will attempt to contribute to this understanding, taking as a starting point the theoretical approach of Bench-Capon and Sartor (2003) and reporting our empirical investigations designed to explore and test this approach.

1.1. USE OF CASES IN AI AND LAW

Despite the recognition of the importance of cases, there has been less agreement on the way in which cases should be represented and used within AI and Law systems. We may distinguish approaches which have used cases as a knowledge source, (e.g. Bench-Capon 1991a), on a par with other sources such as statutes and commentaries, and those which have placed importance on the structure and manipulation of cases as entities, as in, for example, the various systems originating in HYPO (Ashley 1990; Alevan 1997; Skalak and Rissland 1992; Brüninghaus and Ashley 2003). In the first approach, cases will be represented only implicitly, whereas in the second they must be represented explicitly. Both these approaches capture aspects of the truth. Given a body of case law, lawyers experienced in the field will be able to give rule-like advice: for example we may say that injury during a standard commute to work will not be considered as “arising out of, or in the course of, employment” and so not attract Industrial Injury compensation. Such advice will typically be based on several cases, any of which could be cited, and represent an overview of the thrust of a number of decisions. On the other hand, when it comes to forming an argument in the context of particular legal case, precedents will be explicitly deployed, in the manner of the HYPO like systems. Both approaches have their strong points: on the first approach we can examine the knowledge that the system will apply: this may be essential if we are to trust the operation of a system in administrative law. Moreover we can examine the knowledge to critique the law, identifying areas where we are dissatisfied with it and perhaps propose amendments to legislation accordingly. On the other hand, such systems do not provide very satisfactory models of legal reasoning, and also fix the theory, whereas in practice, the interpretation of cases is, at least potentially, continually open to reconsideration (e.g. Levi 1949). The second approach means that each new situation is thought through afresh on its particular merits, rather than being decided mechanically.

A middle way, which attempts to include both aspects, is to introduce the notion of theory construction: for example McCarty states:

“The task for a lawyer or a judge in a “hard case” is to construct a theory of the disputed rules that produces the desired legal result, and then to persuade the relevant audience that this theory is preferable to any theories offered by an opponent” (McCarty 1995, p. 285).

On this view, there is a body of knowledge, in the form of a theory, but these theories are always potentially constructed anew in the context of difficult cases. We therefore attain the benefits of both approaches: the theory provides the knowledge for inspection, and the process of construction can

reflect the practice of legal argument. This paper concentrates on the nature of theories: the construction of theories through argumentation is the subject of developments from the work reported here: a report of these developments can be found in Chorley and Bench-Capon (2005).

One recent account of reasoning with legal cases which takes this view of reasoning with cases as theory construction is the work of Bench-Capon and Sartor, most fully reported in Bench-Capon and Sartor (2003). Their account is entirely theoretical, and in this paper we will describe some empirical investigations based on this approach, attempting to make it more concrete and open to evaluation, and to identify and inform some key design choices that must be made when realising the approach in practice. The approach itself is summarised in the next section.

2. Bench-Capon and Sartor

Bench-Capon and Sartor take as their starting point a background comprising a set of cases (“the case background”) each described using some set of descriptors (“the factor background”). In their initial presentation the descriptors correspond to factors as used in CATO (Aleven 1997), although Bench-Capon and Sartor (2003) also presents extensions intended to allow cases to be described in terms of dimensions as used in HYPO (Ashley 1990). The factor background construes each descriptor as a reason for deciding the case for one of the parties to it, either the plaintiff or the defendant, and associates each descriptor with this party and a value, which represents the social value enhanced or promoted by deciding the case for the favoured party given the presence of this reason. Thus, for example, in Trade Secrets Law, taking security measures is a reason for deciding for the plaintiff, and promotes the value that people should take reasonable steps to protect their own interests.

The use of values is important in Bench-Capon and Sartor (2003). The idea, originally proposed in Berman and Hafner (1993) and developed in Bench-Capon (2002), Prakken (2002), and Sartor (2002) is that cases reveal preferences between conflicting reasons, and that these preferences are explained in the theory through the priority given to the values promoted by these reasons. Values, in this way, allow us to deduce preferences which have not yet been explicitly tested in any precedent case. Consistency is required of preferences between rules and values, since this is an essential condition of any coherent theory.

Against this background, the aim of theory construction when presented with some new cases is to produce a theory which provides a sufficient explanation¹ of the previous cases, and which gives an answer in the new case.

Competing theories may be constructed, and these need to be critically compared to determine which should be applied to the new case.

In Bench-Capon and Sartor (2003), a theory comprises a five-tuple:

- A set of cases selected from the case background (C);
- A set of factors selected from the factor background (F);
- A set of rules, comprising simple rules relating factors to the side favoured, and composite rules with several reasons in the antecedent (R);
- A set of preferences between these rules (RP);
- A set of preferences between the various values (or sets of values) promoted by the factors (VP).

Each of these five sets is initially empty. The theory is constructed by adding to them using one of a number of theory constructors defined in Bench-Capon and Sartor (2003). Here we give informal explanations of the theory constructors: formal definitions are given in Bench-Capon and Sartor (2003).

- *Include Case*: This adds a case from the case background to C.
- *Include Factor*: This adds a factor from the factor background to F. Note that the cases in the theory are described only using the factors which have been included. Thus adding a factor to F may modify the descriptions of the cases in C. Additionally the factor adds to R a rule expressing that the factor is a (defeasible) reason to decide for the party it favours.
- *Factors Merging*: Given a rule in R, the antecedent may be strengthened to give a new rule. Antecedents may be strengthened only by the addition of another factor favouring the *same* party to the dispute.
- *Rule Broadening*: Given a rule in R, the antecedent may be weakened to give a new rule by omitting one of the factors from the antecedent.
- *Rule Preference from Cases*: Given a case in C to which two rules in R, each favouring a different party are applicable, we may infer a preference for the party which won the case and add this to RP. Moreover, from this rule preference we may infer that the set of values promoted by following the preferred rule are preferred to those promoted by following the other rule, and add this to VP.
- *Rule Preference from Value Preference*: Given a preference in VP and two rules corresponding to the related sets of values, we may deduce that the rule relating to the preferred value is preferred to the other rule, and include this in RP.
- *Arbitrary Rule Preference*: Using this constructor a preference is added to RP, even though no case can be found to justify it.
- *Arbitrary Value Preference*: Using this constructor a preference is added to VP, even though no rule preference can be found to justify it.

Starting from this notion of theory we identified a number of issues which needed to be resolved in any practical realisation of the approach, and designed a programme of research to answer these questions. The programme of research and the questions addressed will be discussed in the following section.

3. Programme of research

Our first objective was to establish that the definitions of the background and theory constructors presented in Bench-Capon and Sartor (2003) were sufficient to provide an account of theory construction. We tested this by realising the definitions in an implemented tool, CATE (CAse Theory Editor). CATE is described in Sect. 4. In order to apply the theories constructed to cases CATE generates executable Prolog code corresponding to the theory.

While Bench-Capon and Sartor (2003) lays out a general approach to modelling reasoning with legal cases, it leaves a number of points of detail open. The experiments described in this paper are intended to cast light on how these issues should be resolved. The focus, of these experiments will not be on the *process* of theory construction, but rather on how theories are *applied* once constructed. We will wish to establish that it is possible for theories as specified in Bench-Capon and Sartor (2003) to explain a body of case law, and to understand how a given theory should be applied to yield a decision on a case. In particular, we will need to know how we can use the theory to evaluate the strengths of two sets of conflicting reasons. Our standpoint therefore will be that of a knowledge engineer attempting to construct a theory of the domain, rather than a lawyer reasoning from the facts of a particular case.

To carry out our experiments we will, however, need to construct some theories. There are three issues we need to consider in constructing these theories. First there is the question of how we should extract the rules of the theory from the past cases. We would hope that there is some systematic approach to theory construction which can be applied, perhaps even automatically. In Chorley and Bench-Capon (2003a) we described three approaches, which we call the *safe* approach, the *simple* approach and the *value driven* approach. These will be explained in more detail below. We have again used these three approaches to construct different theories. Our first question is therefore:

Q1: How should we select cases and extract rules for inclusion in the theory?

Once cases have been chosen and rules identified, a similar question arises with respect to factors. Within the general approach it still remains possible to include or exclude the factors that appear in the cases and the domain

analysis generally. We could therefore aim at a theory with few or many factors. In particular, having built a theory using some of the available factors, we have the option of including more factors, using the established value preferences to assign priorities to the rules containing them. This gives rise to the second question:

Q2: Should we be inclusive or exclusive with regard to factors?

It is a central part of the philosophy of Bench-Capon and Sartor (2003) that preferences between factors reveal preferences between values, which can then in turn determine preferences between other factors relating to those values. If this is so, we should expect to be able to include additional factors without major revisions to the value preferences of the theory, since their priority will be determined by the established value preferences. Thus we may pose a third question:

Q3: Is there evidence to suggest that values can be used to determine the relative importance of factors?

Preferences between rules are explained in terms of a comparison between the sets of values promoted by the factors contained in the rules. Since our theory does not contain every possible preference between sets of values, preferences between other sets of values must be derived from these preferences. There are, however, several plausible ways in which sets of values might be compared. The fourth question therefore is:

Q4: How should sets of values be compared?

A related question concerns what should be done when a case contains two factors relating to the same value, but favouring different sides. In such cases we may need to prefer a factor rather than a value. Our fifth question is:

Q5: Is it possible to use a general principle to pre-order factors relating to the same value?

In Bench-Capon and Sartor (2003) it is assumed that factors promote values to an equal degree: if a rule contains a factor then following that rule is held to promote the value associated with the factor. It is not impossible, however, that different factors will promote values to different degrees, e.g. Hage (2001), and that this needs to be considered. This leads to the sixth question:

Q6: Is there evidence to suggest that factors promote values to different degrees?

Again, Bench-Capon and Sartor (2003) does not allow for accumulation of factors: if following a rule promotes a value, then it is not relevant whether this is the result of the presence of one factor or several. Perhaps, however, a value is promoted to a greater extent if several factors are present. The seventh question is thus:

Q7: Is there evidence to suggest that factors relating to a particular value have a cumulative effect?

We see answers to these questions as central to the effective modelling of reasoning with cases in law in the manner of Bench-Capon and Sartor (2003). Obviously the experiments on a single domain with a limited number of cases described here cannot produce definitive answers to them: our hope, however, is that the experiments may help to move the debate forward, and indicate which lines may be worth pursuing.

The remainder of the paper is organised as follows. Sect. 4 describes the theory tool, CATE, and how it can be used to build theories and generate the corresponding executable code. Section 5 sets the scene for the detailed experiments by describing the domain in which they were performed, US Trade Secrets Law. Section 6 describes the principles used to construct theories in the experiments, and the resulting theories. Section 7 describes the different forms of comparison of unweighted factors that are used in the experiments, the different principles of weighting considered and the various experiments performed with factors, and suggests answers to the questions given earlier in the section. Section 8 extends the experiments to consider structured values rather than factors, and Section 9 describes the experiments performed using structured values. We conclude by summarising our results and giving a brief overview of our current work on the automation of theory construction.

4. CATE

4.1. PURPOSE OF CATE

CATE (CAse Theory Editor) is a tool developed to provide support to the process of understanding a legal domain through theory construction. CATE is intended to be useful both to lawyers exploring their understanding of a set of cases, and to knowledge engineers desirous of building an automated system. By providing a means rapidly to develop and execute theories, the tasks of exploring alternatives and refining initial intuitions is greatly eased.

4.2. THE PROGRAM

Figure 1 gives a screen shot of CATE. It is designed to embody the set of theory constructors as described in Bench-Capon and Sartor (2003) and has been implemented in Java. There are panels to show case and factor backgrounds, and the theory under construction. The case and factor backgrounds are read from previously prepared files. Theories can be saved,

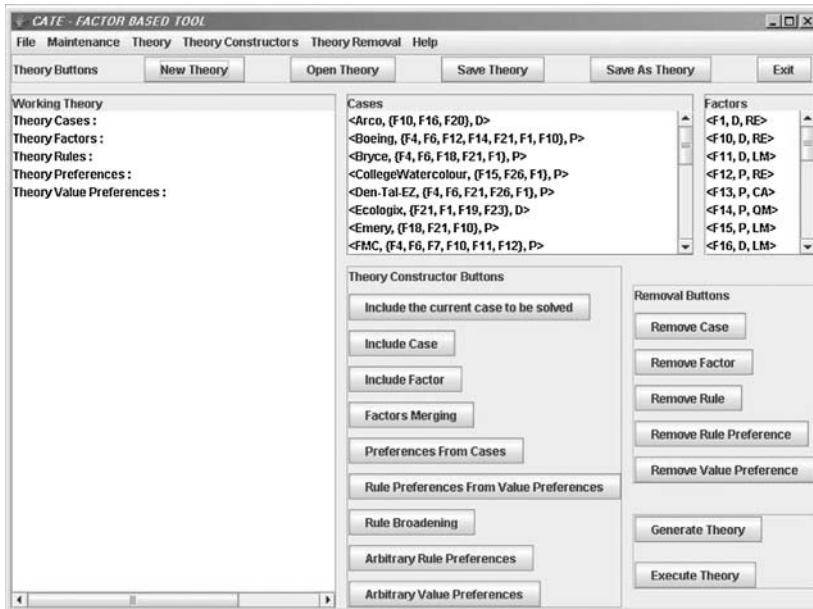


Figure 1. This figure shows CATE before a Theory is constructed.

and previously constructed theories loaded for extension and modification. The various theory constructors can be used by clicking on the appropriate buttons on the screen, each of which will invoke a dialogue box appropriate to the particular constructor. For example, to include a case into the theory, the *Include Case* button is selected and the user is prompted to choose which case they want to add. CATE also provides some checking on the legality of use of the constructors: when a user specifies preferences over rules or values, CATE checks that the resulting theory is consistent and if adding the preference would make the theory inconsistent then a warning is issued and the preference is not added. If a user still wishes to include the preference, then they must first remove an existing preference causing the conflict. CATE also tracks where the rule preferences came from, so that we can distinguish preferences derived from cases from those derived from a value preference, by labelling each rule preference to show which theory constructor was used.

Because CATE starts from a case and factor background, it requires that a domain has already been analysed to identify factors with which to describe the cases, and to provide a set of case descriptions in terms of these factors. CATE is not restricted to any particular domain and so can be used with any domain for which the analysis to supply the requisite background is available.

4.3. BUILDING THEORIES WITH CATE

The theory is created by selecting buttons to include items in the theory. To build a simple theory using CATE, the user must first select some cases to include in the theory by selecting the *Include Case* button and choosing the desired cases. Next some factors to be used in the theory are selected by selecting the *Include Factor* button and choosing the factors. Including factors will also include the simple rules associated with them. Both of these buttons have the added option of including all the cases or factors contained in the backgrounds so the user does not have to manually select everything if the whole background is to be used.

Next the user can add rule preferences by selecting the *Preference From Cases*, *Rule Preference From Value Preference* or *Arbitrary Rule Preference* buttons. If the *Preference From Cases* button is selected then the user is prompted to choose the most preferred rule, then the least preferred rule and finally the case which supports this preference. CATE checks that the case does support this rule preference and if it does, then the rule preference and associated value preference are included into the theory with the rule preference being labelled with its supporting case. However, CATE also checks that this new preference does not introduce inconsistency to the theory and if it does then a warning is issued and the preference is not added.

If the other rule preference buttons are used then for the *Rule Preference From Value Preference* constructor the preference is labelled with $\langle | \textit{From Value Preference} | \rangle$ and if the *Arbitrary Rule Preference* constructor is used then the preference is labelled with $\langle | \textit{Arbitrary Rule Preference} | \rangle$. Once a basic theory has been constructed, the theory can be modified using the theory constructors as desired. A full reconstruction of the example of Bench-Capon and Sartor (2003) can be found in Chorley and Bench-Capon (2004a).

4.4. CODE GENERATION

The *Generate Theory* button takes the constructed theory and translates it into Prolog code and the *Execute Theory* button takes the Prolog code and executes it to give the outcome which results from applying the theory for each case included in the theory. For a new theory for which the Prolog code has not yet been generated the *Execute Theory* button will also generate the Prolog code and then execute it, but if there is already Prolog code then it will simply execute the existing code. This means that if the user wishes to modify the Prolog code manually they can then execute this modified code.

The code is generated according to the theory in the following way. CATE first takes the value preferences and translates them into rule preferences, by substituting the factors related to the values for the values. It then adds the

original rule preferences and can start ordering the rules. The standard execution of Prolog assigns priority to its clauses in accordance with the order in which they appear in the program: thus the rule preferences are enforced in the program by ordering the rules themselves.

The rules forming the rule preferences are divided into three groups. The *left* group, which contains only those rules which are always most preferred, and so only appear on the left side of the rule preferences. The *right* group, which contains only those rules which are always least preferred and so only appear on the right side of the preferences. Finally the *middle* group, which contains the rules which appear on both sides of rule preferences. Figure 2 shows the process of how CATE sorts the rules. The four rules forming the three rule preferences are sorted into the correct groups. Rule A only appears on the left hand side of the preferences and so is placed in the left group. Rule D only appears on the right hand side of the preferences and so is placed in the right group. Rules B and C each appear on the left hand side of one rule reference and the right hand side of a different rule preference so these are placed in the middle group. The left and right groups are sorted alphanumerically and saved. The middle group is again sorted into three groups and this process continues until there are no rules in the middle group.

Each time the process repeats the new left rules are placed below the existing left rules and the new right rules are placed above the existing right rules. When the process is complete the right rules are placed below the left rules giving a complete ordering of the rules with the most preferred rules at the top of the list and the least preferred rules at the bottom. Any rules which do not appear in a rule preference and hence are not in the list are placed at the bottom of the sorted list.

<u>Rule Preferences</u>		
1) A > B		
2) B > C		
3) C > D		
<u>First Pass</u>		
Left Group	Middle Group	Right Group
A	B C	D
<u>Second Pass</u>		
Left Group	Middle Group	Right Group
A B		C D
<u>Finished List</u>		
A		
B		
C		
D		

Figure 2. Three rule preferences and how the rules are sorted into a finished list.

Note that the theory determines only a partial order on the rules. Each time we have, for example, a *left* group, we know that these rules have a higher priority than the rules in the other two groups. We do not, however, have any information as to the priority of the rules *within* the left group. The theory thus determines a family of programs, each consistent with the theory, but differing as to the ordering of rules with indistinguishable priorities. Initially CATE simply uses an arbitrary ordering within groups. This means that the program may need refinement, adjusting the order of rules within a group to produce the desired behaviour. This refined program will be an alternative expression of the theory, and so the theory itself will need no modification.

CATE then takes the sorted list of rules and translates them into Prolog Clauses. This is a straightforward matter of mapping from the rule syntax into Prolog syntax. For example, the rule ($\langle \{F1\}, D \rangle$) is transformed to *outcome*(*X*, *d*):- *factor*(*X*, *f1*). The more complex rule ($\langle \{F1, F10, F19\}, D \rangle$) is transformed to *outcome*(*X*, *d*) :- *factor*(*X*, *f1*), *factor*(*X*, *f10*), *factor*(*X*, *f19*). It then includes the cases and their factors, which will supply the facts for the execution of the program and saves the complete program, rules and facts.

To execute the theory, CATE takes the Prolog code and for each case searches through the sorted clauses to find the first clause to fire for the case. This clause gives an outcome to the case. These case outcomes can then be used to evaluate how the theory performs with respect to the actual decisions for the cases, so as to verify that the theory does indeed explain the selected cases.

5. Environment for the experiments

5.1. THE DOMAIN

For our experiments we chose the domain of US Trade Secrets Law. This was chosen because it has formed the basis of several of the most important and best documented legal Case Based Reasoning Systems (Ashley 1990; Alevén 1997; Brüninghaus and Ashley 2003, and numerous conference and journal papers). This means that we have a wealth of discussion, examples of system behaviour, and most important, a domain analysis which has proven a successful basis of a case based reasoning system to draw on.

We based our factor background on the CATO system as described in Alevén (1997). This work provides 26 factors, each identified as being plaintiff or pro-defendant. In this paper we use the identifiers for the factors used in Alevén (1997). The complete list of factors is given in Table I.

The final element needed for the factor background is values. CATO, however, does not use this notion, and so we will need to supply this element

Table I. Factors in CATO (NB: There is no F9 in Aleven (1997))

Pro-plaintiff factors	Pro-defendant factors
<i>F2 Bribe Employee</i>	<i>F1 Disclosure in Negotiations</i>
<i>F4 Agreed not to disclose</i>	<i>F3 Employee Sole Developer</i>
<i>F6 Security Measures</i>	<i>F5 Agreement not specific</i>
<i>F7 Brought Tools</i>	<i>F10 Secrets Disclosed Outsiders</i>
<i>F8 Competitive Advantage</i>	<i>F11 Vertical Knowledge</i>
<i>F12 Outsider Disclosures Restricted</i>	<i>F16 Info Reverse Engineerable</i>
<i>F13 Noncompetition Agreement</i>	<i>F17 Info Independently Generated</i>
<i>F14 Restricted Material Used</i>	<i>F19 No Security Measures</i>
<i>F15 Unique Product</i>	<i>F20 Info Known to Competitors</i>
<i>F18 Identical Products</i>	<i>F23 Waiver of Confidentiality (d).</i>
<i>F21 Knew Info Confidential</i>	<i>F24 Info Obtainable Elsewhere</i>
<i>F22 Invasive Techniques</i>	<i>F25 Info Reverse Engineered</i>
<i>F26 Deception</i>	<i>F27 Disclosure in Public Forum</i>

of the background ourselves. In this paper we will use the analysis developed in Chorley and Bench-Capon (2003a, b, c). We recognise that we are not legal experts and others might have chosen different values and perhaps a different distribution of factors across values. One strength of CATE as a tool is that it allows those who wish to build a theory from a different perspective the means to do so rapidly and in a way which can be quickly evaluated, and which allows different analyses to be experimented with. None the less in order to illustrate how CATE can be used to build and evaluate theories, we do need to use some analysis linking factors to values: we will return to the plausibility of our analysis at the end of this section.

So what values seem to underlie the factors? Values relate to behaviour that the law wishes to encourage or discourage. The motive for encouraging or discouraging behaviour is to promote some socially desirable end. For example, marking F1 as an important consideration would promote the social end that people act with reasonable care for their own interests: if one has a secret one has a certain responsibility to keep it to oneself. We therefore examined the factors to identify patterns of behaviour which they encouraged or discouraged.

First a number of factors relate to confidentiality agreements. It is reasonable to suppose that if all trade secret disputes were governed by a specific agreement, the task of deciding them would become a relatively straightforward matter. We would therefore expect the law to encourage such agreements to be made. Our first value then is *Confidentiality Agreement (CA)*: the side favoured will depend on the nature of the agreement. This value secures five factors:

- *F4 Agreed not to disclose (p)*
- *F5 Agreement not specific (d)*
- *F13 Noncompetition Agreement (p)*
- *F21 Knew Info Confidential (p)*
- *F23 Waiver of Confidentiality (d)*.

Next it seems that the law does not wish to condone lax behaviour, so that it wishes people with secrets to take reasonable measures to protect them. This gives the second value *Reasonable Efforts (RE)*. Additionally this can be seen as evidence that the plaintiff regarded his knowledge as a secret. Making such efforts is encouraged if having made them favours the plaintiff, and having failed to make them favours the defendant. Six factors share this value.

- *F1 Disclosure in Negotiations (d)*
- *F6 Security Measures (p)*
- *F10 Secrets Disclosed Outsiders (d)*
- *F12 Outsider Disclosures Restricted (p)*
- *F19 No Security Measures (d)*
- *F27 Disclosure in Public Forum (d)*.

Third the law wishes to encourage competition by legitimate means. Therefore if a person can develop the product using *Legitimate Means (LM)*, this should tell in their favour. This covers eight factors. Note that one of them is pro-plaintiff; the uniqueness of a product creates a presupposition that it cannot be developed by legitimate means, and so places an extra burden of proof on the defendant.

- *F3 Employee Sole Developer (d)*
- *F11 Vertical Knowledge (d)*
- *F15 Unique Product (p)*
- *F16 Info Reverse Engineerable (d)*
- *F17 Info Independently Generated (d)*
- *F20 Info Known to Competitors (d)*
- *F24 Info Obtainable Elsewhere (d)*
- *F25 Info Reverse Engineered (d)*

The reverse of this is that illegal or immoral means should be discouraged. Five factors relate to this value, *Questionable Means (QM)*, which always favours the plaintiff:

- *F2 Bribe Employee (p)*
- *F7 Brought Tools (p)*
- *F14 Restricted Material Used (p)*
- *F22 Invasive Techniques (p)*
- *F26 Deception (p)*

The final two factors are intended to show that the secret had *Material Worth (MW)*. The law would naturally attempt to discourage litigation about

secrets of no worth, and so will favour the plaintiff if his secret had demonstrable value. Two factors, both of which favour the plaintiff, are used here:

- *F8 Competitive Advantage* (*p*)
- *F18 Identical Products* (*p*).

We have now assigned the factors to five values. Conveniently the distribution is reasonably equal, with only *Material Worth* represented by substantially fewer factors.

Although the process by which we have arrived at our five values is more common sense and knowledge engineering than legal analysis, we believe that the five values that result have plausibility, in that they have a strong correspondence to analyses carried out by other groups. The values have clear mappings to the abstract factors of Aleven (1997). Even more importantly, the logical structure of Trade Secrets Law used in IBP (Brüninghaus and Ashley (2003), produced by a direct analysis of the Uniform Trade Secrets Act and the Restatement of Torts has five leaf nodes. These are Information Valuable, Maintain Secrecy, Information Used, Confidential Relationship and Improper Means. These are, in effect, our five values of Material Worth, Reasonable Efforts to protect the secret, Legitimate Means (i.e. protected information not used), Confidentiality Agreement and Questionable Means, respectively. The crucial role in prediction of these five issues is further emphasised in Ashley and Brüninghaus (2003). We therefore believe that our analysis provides a reasonable basis on which to perform our experiments.

5.2. CASES

We must now select a set of cases. A number are described in Aleven (1997), and Brüninghaus and Ashley (2003). We chose a selection of these, although excluding some of the cases flagged as problematic by Aleven. Our initial selection, used in Chorley and Bench-Capon (2003a) included seven found for the plaintiff and seven found for the defendant. We continued to use these cases, which we call *Group 1*, as the initial data set with which to construct our theories. In order to test whether the theories generalise to cases not used in their construction we added another group of cases (*Group 2*). In Table II we show the cases, the factors present in them, split according to whether they favour the plaintiff or the defendant, the values that would be promoted by deciding in favour of the plaintiff and the values that would be promoted by finding for the defendant and the outcome of the case.

Table II. Factor based cases used in CATE

	Pro-P factors	Pro-P values	Pro-D factors	Pro-D values
<i>Group 1</i>				
Arco			F10, F16, F20	RE LM
Boeing	F4, F6, F12, F14, F21	CA, RE, QM	F1, F10	RE
Bryce	F4, F6, F18, F21	CA, RE, MW	F1	RE
College Watercolour	F15, F26	LM, QM	F1	RE
Den-Tal-Ez	F4, F6, F21, F26	CA, RE, QM	F1	RE
Ecologix	F21	CA	F1, F19, F23	CA, RE
Emery	F18, F21	CA, MW	F10	RE
Ferranti	F2	QM	F17, F19, F20, F27	RE, LM
Robinson	F18, F26	QM, MW	F1, F10, F19	RE
Sandlin			F1, F10, F16, F19, F27	RE LM
Sheets	F18	MW	F19, F27	RE
Space Aero	F8, F15, F18	MW, LM	F1, F19	RE
Televation	F6, F12, F15, F18, F21	CA, RE, LM, MW	F10, F16	RE, LM
Yokana	F7	QM	F10, F16, F27	RE, LM
<i>Group 2</i>				
CMI	F4, F6	CA, RE	F20, F17, F16, F10, F27	LM, RE
Digital Development	F6, F8, F15, F18, F21	RE, MW, LM, CA	F1	RE
FMC	F4, F6, F7, F12	CA, RE, QM	F10, F11	RE, LM
Forrest	F6, F15, F21	RE, LM, CA	F1	RE
Goldberg	F21	CA	F1, F10, F27	RE
KG	F6, F14, F15, F18, F21	RE, QM, LM, MW, CA	F16, F25	LM
Laser	F6, F12, F21	RE, CA	F1, F10	RE

Table II. Continued

	Pro-P factors	Pro-P values	Pro-D factors	Pro-D values
Lewis	F8, F21	MW, CA	F1	RE
MBL	F4, F6, F13	CA, RE	F5, F10, F20	CA, RE, LM
Mason	F6, F15, F21	CA, RE, LM	F1, F16	RE, LM
Mineral Deposits	F18	MW	F1, F16, F25	RE, LM
National Instrument	F18, F21	MW, CA	F1	RE
National Rejectors	F7, F15, F18	QM, LM, MW	F10, F16, F19, F27	RE, LM
Reinforced	F4, F6, F8, F15, F21	CA, RE, MW, LM	F1	RE
Scientology	F4, F6, F12	CA, RE	F10, F11, F20	RE, LM
Technicon	F6, F12, F14, F21	RE, QM, CA	F10, F16, F25	RE, LM
Trandes	F4, F6, F12	CA, RE	F1, F10	RE
Valco-Cincinnati	F6, F12, F15, F21	RE, LM, CA	F1, F10	RE

From this we were able to construct the files required to supply the factor and case backgrounds to CATE.

6. Constructing theories

6.1. NEED FOR A METHOD

Having established our background, we can proceed to construct theories. Often theory construction is directed towards a particular, as yet undecided, case. Here, however, we are trying to come to a theory which will explain as many of the available cases as possible, and so we need to choose our cases. To guide us we need some principles for how we will construct our theory. To explore a range of possibilities, we decided to use three approaches.

For each approach we restricted ourselves to four cases, selected from Group 1 to construct the theory, two won by the plaintiff and two won by the defendant. The other cases in Group 1 will then be used to assess the theory, and if necessary to refine it. Once the theory has been refined, it will then be tested against the Group 2 cases to see if it generalises so as to classify new cases correctly. The relative success of the three methods will help us to answer Q1 of our research questions.

6.2. THE “SAFE” METHOD

Using this method we attempt to say no more than we are strictly justified in saying from a consideration of the cases. We will not attempt to generalise beyond them, nor impose any preconceptions as to how the domain should be. On this method we are willing to include as many factors as possible, but we will produce rules which do not go beyond the minimum that we are entitled to infer. This latter effect is given by using the method of Prakken and Sartor (1998) for producing rules from cases, whereby the conjunction of all the pro-plaintiff factors present gives one rule, the conjunction of all the pro-defendant factors gives another rule, and the priority between them is determined by the decision.

For this method we select *Emery* and *College Watercolour* as plaintiff cases and *Robinson* and *Sheets* as our defendant cases, the idea being that these would generate the most powerful rules since they involve the fewest factors. Representing the rules from these four cases in the manner of Prakken and Sartor (1998) yields the rule and value preferences shown in Figure 3. Figure 3 shows the Rule Preference and Value Preference sections

```

Theory Preferences :
  pref(<{F1, F10, F19}, D>, <{F18, F26}, P>)
    <|<Robinson, {F18, F26, F1, F10, F19}, D>|>
  pref(<{F15, F26}, P>, <{F1}, D>)
    <|<CollegeWatercolour, {F15, F26, F1}, P>|>
  pref(<{F18, F21}, P>, <{F10}, D>)
    <|<Emery, {F18, F21, F10}, P>|>
  pref(<{F19, F27}, D>, <{F18}, P>)
    <|<Sheets, {F18, F19, F27}, D>|>
Theory Value Preferences :
  valpref({MW, CA}, {RE})
  valpref({RE}, {MW, QM})
  valpref({LM, QM}, {RE})
  valpref({RE}, {MW})

```

Figure 3. Rule and value preferences from “Safe” method.

from the theory as produced by CATE. The Rule Preferences are of the form $pref(<rule\ 1>, <rule\ 2>) \quad <|Case|>$ where $pref$ is a label denoting that this is a preference between rules, $rule\ 1$ is preferred over $rule\ 2$ and the preference is labelled with the case from which it is derived. Each rule consists off a set of factors forming the antecedent and the outcome, either P or D, as the consequent. The Value Preferences are of the form $valpref(\{value\ set\ 1\}, \{value\ set\ 2\})$ where $valpref$ is a label denoting that this is a preference between sets of values and $value\ set\ 1$ is preferred over $value\ set\ 2$.

6.3. THE “SIMPLE” METHOD

The second method was intended to produce the simplest theory. Here we will want to use the fewest possible factors, and will be willing to make assumptions which enable us to produce rules not strictly justified by the cases. Hence we will select a small set of factors which covers all the cases, and choose cases to establish priorities between them. The idea here is similar to the automatic induction of decision trees, which strives to produce the smallest tree capable of classifying the instances on the data available.

For this method we must first select our factors. What we need is a set of factors such that at least one pro-plaintiff factor occurs in every case decided for the plaintiff, and at least one pro-defendant factor occurs in every case decided for the defendant. F21 occurs in 6 of the pro-plaintiff cases, so we choose this together with F15 to handle *Space Aero*. For our pro-defendant factors F19, F20 and F27 will cover all defendant cases. Now we need express preferences only where we have both a pro-plaintiff and a pro-defendant factor. In only two cases do we have a conflict to resolve: *Space Aero* and *Ecologix*, so we express preferences according to the outcomes of these two

```

Theory Preferences :
  pref(<{F15}, P>, <{F19}, D>)
    <|<SpaceAero, {F8, F15, F18, F1, F19}, P>|>
  pref(<{F19}, D>, <{F21}, P>)
    <|<Ecologix, {F21, F1, F19, F23}, D>|>
Theory Value Preferences :
  valpref({LM}, {RE})
  valpref({RE}, {CA})

```

Figure 4. Preferences for the “Simple” method.

cases. For this approach, nothing is to be gained by including additional cases, so only these two are used in this theory. We thus get the following rule and value preferences shown in Figure 4:

6.4. THE VALUE DRIVEN METHOD

Our third approach will be value driven, thus embodying some pre-determined assumptions about how we believe the domain should operate. Here we will first reflect on the values and produce a ranking. We will then choose factors to represent these values, and cases to establish the desired value order. It is, of course, arguable that this is not the approach that lawyers would take, since they would want to order values in the context of the facts of a particular case, and would not wish to consider the importance of values abstractly. Lawyers might well, therefore, prefer the safe method which does rely more heavily on the facts of particular cases. On the other hand, politicians, who make law in advance of the facts, often do have social purposes in mind when they construct legislation, and are capable of prioritising these purposes. In any event, the purpose of using the method in our experiments is to provide a principle which will produce a third theory to compare with the others: other approaches to theory construction could also have been taken.

To use this approach we must first decide on a value order. We do not need to distinguish between *Questionable Means* and *Material Worth*, since these both always favour the same side (the plaintiff). We might suppose that the most highly rated value is *Confidentiality Agreement*, since if all the dealings were regulated by properly drafted agreements, there would be no problems to decide. We rate *Legitimate Means* next: in the absence of a specific agreement, the right to enterprise must be protected. We rate *Reasonable Efforts* third, since people must take some steps to protect themselves. This leaves *Questionable Means* and *Material Value* at the bottom. Is *Material Worth* so unimportant, when surely it is a *sine qua non* for an action? Well, it is of little importance here, since while if it is not present the action seems pointless, it does not really cast much light on

```

Theory Preferences :
  pref(<{F15}, P>, <{F19}, D>)
    <|<SpaceAero, {F8, F15, F18, F1, F19}, P>|>
  pref(<{F19}, D>, <{F18}, P>)
    <|<Sheets, {F18, F19, F27}, D>|>
  pref(<{F19}, D>, <{F26}, P>)
    <|<Robinson, {F18, F26, F1, F10, F19}, D>|>
  pref(<{F21}, P>, <{F16}, D>)
    <|<Televation, {F6, F12, F15, F18, F21, F10, F16}, P>|>
Theory Value Preferences :
  valpref({CA}, {LM})
  valpref({LM}, {RE})
  valpref({RE}, {LM})
  valpref({RE}, {MW})

```

Figure 5. Preferences from the “Value Driven” method.

whether the defendant behaved incorrectly. It does not, in fact, appear in every case. We assume that this is because it was accepted by both sides, and so is made explicit only if the matter is raised in an effort to discredit the action. Arguably also, the presence of pro-plaintiff CA and RE factors implies MW.

In order to establish this order on values we need four cases. In choosing representative factors we should have an eye mainly to coverage. First we choose a case where $CA > LM$, *Televation*, and F21 and F16 can play this role. For $LM > RE$ we choose *Space Aero* and use factors F15 and F19. We now need $RE > QM$, for which we can have *Robinson* with factors F19 and F26. Finally for $RE > MW$ we chose *Sheets* with F19 and F18. This yields the third theory shown in Figure 5.

Note that we have produced three rather different theories. The value preferences are considerably different – Theory 2 recognises only three rather than five values – and Theory 1 places more stress on RE than does Theory 3.

6.5. THEORY COMPLETION

In constructing the theory, typically not all the rule preferences determined by the value preferences will be explicitly included. Since it is part of the philosophy of Bench-Capon and Sartor (2003) that the impact of factors can be derived from the values to which they relate, we can complete the theory by adding the remaining rule preferences entailed by the value preferences of the theory. Now, when the theory is executed, the value preferences are used to create these additional rule preferences which are then used to rank all the factors present in the theory. In this way the user does not need to explicitly identify and add all the rule preferences entailed by the value preferences, because this can be done for them.

For example, for Theory 3 given above in Figure 5, the value preference of $CA > LM$ ranks all the Confidentiality Agreement factors used above all the Legitimate Means factors used.

7. Comparing factors

7.1. FACTOR COMPARISON METHODS

The initial method of code generation caused the first Prolog rule to match for each case to fire and give the case the outcome that the rule promotes. Thus we are embodying the assumption that the most preferred rule will govern the case by itself. This assumption will be relaxed in later experiments.

Even so, there are a number of ways in which we can form Prolog rules from the theory. In our experiments we considered four different methods.

The first two methods distinguish between whether we chose to consider the absence of factors as well as their presence. The intention here is to explore Q4 of our research questions.

Best factor: This method of code generation simply sorts the rules present in the theory according to the value order and rewrites the rules as Prolog clauses. Effectively this method weights a set of values according to its most highly weighted value, unless a comparison between particular sets of values has been treated explicitly. Where a case contains the same value as the most highly rated for both the pro-plaintiff and pro-defendant sets, the outcome is determined by preferences between factors related to this value, which makes the ordering of factors within a value significant. Much of the program refinement process, if this method is used, may be seen as tuning this ordering of factors with common values.

Best nonshared factor: A second possibility, following Prakken (2000), is to discount values found in both the pro-plaintiff and pro-defendant sets, giving the set importance according to its most significant value not in the other set. This *cancellation* method creates Prolog rules of the form *f1 and not f2 and not f3... and not fn* where *f2...fn* are the factors which relate to the same value as *f1* but favour the opposite side.

The next two methods address the issue of whether it is possible to order factors within a value (Q5) in advance according to some general principle. We suggest two candidate possibilities:

Exceptions: The *exception* method creates more rule preferences by ordering the factors within each value. The idea here is that more importance should be placed on exceptions than defaults. Each value typically is used to favour either the Plaintiff or the Defendant: thus *Material Worth* and *Questionable Means* always favour the Plaintiff and seven of the eight factors

for *Legitimate Means* favour the defendant. So where a factor favours the party not usually favoured by the value it is considered an exception. Whether a factor is considered an exception is determined by examining the factors that relate to each value, and considering whether pro-plaintiff or pro-defendant factors predominate. The factors promoting the less common outcome are taken to be exceptions and these are preferred to the factors promoting the more usual outcome for the value.

CATO: The second method takes information from the *CATO* system. The *CATO* method also sorts the factors within each value. The factor hierarchy from Aleven (1997) contains thin and thick arcs and these are used to represent whether the factor is strong or weak. Strong factors are those with strong/thick arcs from the factor to the abstract factor. In *CATO* strong factors block the operation of weaker factors. In *CATE* the strong factors within each value are preferred to the weak factors. When the theory is executed in Prolog, the preferred factor will, if present, decide the case before the rule using the weaker factor is reached, giving a similar blocking effect.

In our experiments we used all of these four methods to apply the theories produced by the three methods of theory construction. This was intended to cast light on Q1, by determining whether there were significant differences in the performance of the theories produced by the different methods.

Additionally, in each case, we augmented the theory by including all the factors as well as those used in the initial theory construction. The constructed theories will typically contain only a subset of the factors available from the background. Since the impact of factors can be derived from the values to which they relate, we can extend the theory by adding the remaining factors available from the background and ranking them according to the value preferences of the theory. The priority assigned to the rules containing these additional factors is determined solely by the preferences between their values in the theory. If the performance does not degrade, this can be taken as an acceptable way of establishing priorities between factors not explicitly considered, providing an answer to Q3.

7.2. RESULTS

The main results of these initial experiments are depicted in Tables III and IV (for more detailed results see Chorley and Bench-Capon 2003c). *Chosen* is for code using only the factors explicitly included and *All* is for the code which includes all the factors available from the background. The cells show the number of cases not correctly classified by the theory, including abstentions as well as misclassifications. The number in brackets show the failures to classify before refinement, for those cases where refinement was used. Recall

Table III. Results for the 14 Group 1 cases

	Theory 1		Theory 2		Theory 3	
	Chosen	All	Chosen	All	Chosen	All
Best factor	1 (3)	0 (3)	0	1 (4)	1	0 (1)
Cancellation	0 (1)	0 (1)	0	1	0	0
Exceptions	1 (3)	0	0	1	1	0
CATO	1 (3)	0 (1)	0	1 (3)	1	0 (1)

Table IV. Results for the 18 Group 2 cases

	Theory 1		Theory 2		Theory 3	
	Chosen	All	Chosen	All	Chosen	All
Best factor	3 (11)	4 (11)	5	6 (9)	6	4 (5)
Cancellation	8	5 (4)	4	6	5	3
Exceptions	3 (11)	5	5	7	6	4
CATO	3 (11)	6 (3)	5	6 (7)	6	4

that Group 1 cases are used to generate the programs, and Group 2 used as a test set.

From these results we can answer some of our original questions.

Q1: How should we select cases and extract rules for inclusion in the theory?

With regard to the Group 1 cases, we can see that all three methods can produce theories which account for the body of cases reasonably well, especially if we can refine our theories by ordering factors within a value. Note also that for Theories 1 and 3, performance improves if more factors are included. This is not true, however, of Theory 2, suggesting that the method tends to overfit the data. None of the theories generalise particularly well, failing to give the correct decision in at least three, and typically more of the eighteen cases. The cases misclassified varied somewhat for the different theories.

Q2: Should we be inclusive or exclusive with regard to factors?

Theories 1 and 3 actually improve their performance on the Group 1 cases if we include all the available factors. For the group 2 cases, abstentions are eliminated, often with the correct decision, except for the overfitting Theory 2. For Theories 1 and 3, we would recommend using all available factors.

Q3: Is there evidence to suggest that values can be used to determine the relative importance of factors?

The benefits of including all available factors, and the fact that these new factors are able to make a positive contribution using the value preferences determined by the other factors sharing their values, offers evidence that values are significant in accounting for the importance to be placed on factors.

Q4: How should sets of values be compared?

The evidence is by no means clear cut here. Theories 1 and 3 perform better with cancellation on the training set, but this improvement does not generalise to the Group 2 cases. An alternative view is that comparison should use weighted factors, and this will be explored in the next set of experiments described in Sect. 7.3.

Q5: Is it possible to use a general principle to pre-order factors relating to the same value?

This question is addressed by the exceptions and CATO methods. The key test, because involving the most factors to order, is when the complete set of factors is used. These methods perform about the same, and perform the same as when no pre-ordering is used. Note, however, that the non pre-ordered results are for the refined theory: before refinement both Theories 1 and 2 made more errors. The suggestion here, therefore, is that pre-ordering works to a certain extent, and can remove at least some of the need for refinement – which amounts to a manual ordering of factors within a value. It does not, however, improve the capacity to generalise.

We next performed experiments involving the weighting of factors. In these experiments instead of the decision being determined by a single rule, all applicable rules are able to contribute to the decision. These experiments will address Q6 and Q7 of our original questions in Sect. 3.

7.3. WEIGHTING FACTORS

In order to allow consideration of all values in a set we assign its members a weight relative to the importance of the value to which they relate. Preferences are now reflected by weights rather than by rule order, allowing for the possibility that several weak factors may collectively outweigh a strong factor. There are two possibilities here, according to whether we count the values represented once, or increase the weight according to how many factors representing each of the values are present (that is, we use “bags” of values rather than sets). To explore this possibility we must assign weights to values.

The *numeric* method assigns weights to each value and hence to each factor within the value. The weight for each value is decided by the value preferences, a value which is preferred being given a larger weight than the

value to which it is preferred. The weight is positive for a plaintiff factor and negative for a defendant factor.

For Theory 3 the value preferences are $CA > LM > RE > (MW, QM)$. The least preferred values of MW and QM are given the weight of 0.1. The next value of RE is given the weight of double the previous weight plus 0.1 which is 0.3. The next value of LM is again given the weight of double the previous weight plus 0.1 which is 0.7. Finally the value of CA is given the weight of double the previous weight plus 0.1 which is 1.5.

Assigning the weights for Theory 3 is easy because the value preference ordering is very simple. Assigning the weights for a more complicated set of value preferences is more difficult because the weights must still ensure that the value preferences work. For Theory 1 possible value weights are: $(MW, QM) = 0.1$, $RE = 0.3$ and $(LM, CA) = 0.7$. Looking at the value preferences for Theory 1 there are several complicated ones. For the value preference of $(MW, CA) > RE$ we obtain a joint weight of 0.8 for (MW, CA) which is larger than the weight of 0.3 for the RE value. For the value preference of $RE > (MW, QM)$ we obtain a joint weight of 0.2 for (MW, QM) which is less than the weight of 0.3 for RE.

Instead of relying on the order of the execution of the rules to enforce priorities, we now consider every factor present in order to determine the relative strength of the plaintiff's and defendant's cases. The size of the numbers produced may be an indicator of confidence in the predicted decision, but we lay no stress on this here.

We adopted four methods of accumulating weights.

Value weights: Values are assigned weights in accordance with the value preferences determined by the theory. The weights for each value present as a pro-plaintiff factor are summed as are the weights for each value present as a pro-defendant factor. The latter is then subtracted from the former to give the outcome for the case. A positive number is pro-plaintiff and a negative number is pro-defendant. This method considers sets of values.

Factor weights: Factors are assigned weights according to the preferences given to the value they represent. The weights for each pro-plaintiff factor present are summed as are the weights for each pro-defendant factor. The latter is then subtracted from the former to give the outcome for the case. A positive number is pro-plaintiff and a negative number is pro-defendant. This method considers bags of values.

Exceptions: Exceptions, as defined above, are considered strong factors, and the weights given to these factors are a multiple of the weight from the value of the factor. These weights are then used as in the factor weights method.

IBP: In Brüninghaus and Ashley (2003) and Ashley and Brüninghaus (2003) a predictive program, IBP, based on CATO is described. In IBP

factors are of three types: *KO Factors*, which are typically sufficient to determine the outcome on their own; *Weak Factors*, which typically have no significant impact on the outcome, but are important for contextualising the case, and *Normal Factors*, which have an influence, but not a determinant influence. We used this classification in our method by multiplying (by 10) the weight from the value for the KO Factors and dividing (by 10) the weight from the value for the weak factors.

7.4. RESULTS

Results are summarised in Table V, which shows the number of cases (Groups 1 and 2, 32 cases in all) misclassified by the theories. The full results, with the numerical outcome for each case was given in Chorley and Bench-Capon (2003c).

From this we can again see the problems with over fitting in Theory 2. For Theory 1 we can see the benefits of including all factors: this is not, however, marked in Theory 3. In comparison with the results of using unweighted factors, we have a clear improvement: suggesting in answer to Q4 that all the elements of a set do need to be considered. The exception method of identifying strong factors is outperformed by the IBP method, indicating that expert knowledge derived from an analysis of cases is superior to the simple principle. No differences between counting all the factors relating to a value rather than simply all the values were found.

Some observations on the mistakes relating to particular cases may be made. Only the results marked with a “*” in Table V classified *Mineral Deposits* correctly. Our reading of the decision was that the Court placed considerable stress on a specific feature of this case which is not adequately represented by any of the factors used. Lipinski and Britz (2000) says “compare *Mineral deposits Ltd. v. Zigan*, 773 P. 2d 606 (Colo. App. 1988) (reverse engineering not allowed when product loaned in confidence)”, although we do not have any factors relating to confidentiality for *Mineral*

Table V. Results for different methods of weighting factors

	Theory 1		Theory 2		Theory 3	
	Chosen	All	Chosen	All	Chosen	All
Weighted values	5	1	1*	5	2	2
Weighted factors	4	1	1*	5	2	2
Exceptions	4	3	1*	4	3	3
IBP	3*	2	1*	2	1	2

Deposits in our background. We might therefore suggest that this case is not suitable for use with the existing factor background. Another case which is consistently misclassified is *Space Aero*, which is a mistake in all IBP methods. This may be explained by the presence of two KO Factors in that case, one for each side. If we disregard these two cases, when using the complete set of factors Theory 1 performs perfectly for all methods except the exception method. Theory 3, however, misclassifies Scientology (albeit with a very small weight) unless IBP is used.

To return to our original questions:

Q6: Is there evidence to suggest that factors promote values to different degrees?

Q7: Is there evidence to suggest that factors relating to a particular value have a cumulative effect?

On the basis of our experiments, there appears to be little difference between accumulating weights from values, from all the factors, and from giving differential weights to factors within values. For Q7 and both Theories 1 and 3 the same cases are misclassified whichever method is used. With regard to Q6, the experience of Brüninghaus and Ashley (2003), however, which reports IBP, which does use differential weights, as significantly outperforming programs that do not, suggests that we should investigate this further. We certainly have no evidence to deny this, and would need to run our experiment on a larger data set before coming to any firm conclusions.

8. Dimensions and structured values

The above experiments were all conducted on cases represented using factors. Factors were used in the CATO system Aleven (1997) and represent a simplification of the original notion of dimensions used in Ashley (1990). A discussion of the differences between factors and dimensions can be found in Rissland and Ashley (2002), and a case for the importance of dimensions is made in Bench-Capon and Rissland (2001).

Factors represent features of cases which can be inferred from the case facts, and which are either present or absent. If present, factors strengthen the case for one of the plaintiff or the defendant. Dimensions allow for a finer grained consideration. On the facts of a case a dimension may be applicable or inapplicable. An applicable dimension represents a range of possibilities, with a direction. One end of the range is a pro-plaintiff extreme, and points along the range represent positions which are increasingly less favourable to the plaintiff and more favourable to the defendant until the other end of the range, the pro-defendant extreme, is reached. The case facts determine the point on the range that applies in that case. Note that dimensions do not

themselves favour either party, although a point on a dimension is more or less favourable to a party. Incorporation of dimensions into theories was considered in Bench-Capon and Sartor (2003), and in this paper we will make those ideas concrete.

Dimensions may be converted to factors in a number of ways:

- (a) a dimension may have a cross over point at which it ceases to favour the plaintiff and starts to favour the defendant. This may be mapped into two factors, one pro-plaintiff and one pro-defendant.
- (b) it may be that one end of the dimension favours, say, the plaintiff, and becomes less favourable as we move away from an extreme, but then becomes inapplicable rather than favouring the defendant. In such cases the dimension maps to a single factor.
- (c) it may be that we wish to map the dimension into a number of factors of differing strength, and possibly favouring different parties.

Examples of all three can be found in the transition from HYPO to CATO. Examples are:

- (a) HYPO has a dimension *Security-Measures-Adopted*. CATO has a pro-plaintiff factor F6 *Security-Measures* and a pro-defendant factor F19 *No-Security-Measures*. Only one of F6 and F19 can be present in a CATO case.
- (b) HYPO has *Brought-Tools*. CATO has the pro-plaintiff factor F7 *Brought-Tools*, but no corresponding pro-defendant factor.
- (c) HYPO has *Secrets-Disclosed-Outsiders*. CATO has a pro-defendant factor F10 *Secrets-Disclosed-Outsiders*, and also the stronger F27 *Disclosure-in-Public-Forum*.

Thus a dimension can be seen as structuring a collection of factors relating to a given issue. Case descriptions can either be constructed in terms of the factors present, or in terms of points on the applicable dimensions.

Factors also have different degrees of importance (represented by thick and thin lines in Alevén (1997) which indicate that a factor may exclude some other factors, and as normal, weak and KO factors indicating predictive power in Brüninghaus and Ashley (2003)). If we compare factors F10 *Secrets-Disclosed-Outsiders* and F27-*Disclosure in Public Forum* we see that they are both defendant factors but F10 is weaker for the defendant because the plaintiff only disclosed the trade secret to some outsiders, whereas F27 is stronger for the defendant because if the plaintiff disclosed the trade secret in a public forum then it can hardly be considered a secret anymore. F27 is a KO factor in Brüninghaus and Ashley (2003). As indicated in (c) above, such factors can be thought of as points on the same dimension with F27 placed at

the stronger part of the dimension and F10 placed at the weaker part of the dimension.

8.1. RELATION TO VALUES

Values bring together groups of factors which promote the same value and dimensions give rise to groups of factors with similar features in common. This similarity is the basis of the extension to values in Bench-Capon and Sartor (2003), and we will follow their account and arrange our factors according to the extent to which they promote the value to which they relate. For example, F10 and F27 can be seen as points on the value scale of taking Reasonable Efforts, with F27 more important than F10. In making this move, however, we are departing considerably from the HYPO conception of dimension, and changing the focus from how the facts of a case are represented to a *measure of the contribution* to an issue made by a factor. For this reason, although the approach is inspired by the notion of dimension, and follows Bench-Capon and Sartor's attempt to accommodate the role of dimensions in their notion of a theory, we will not use the term "dimension" here. What we are doing is providing structure to our values, by identifying the different extents to which the factors promote their values. In what follows, therefore, we will therefore use the term "structured value" rather than dimension.

8.2. COMPARISON WITH IBP MODEL

In Brüninghaus and Ashley's program IBP (Issue-Based Prediction) (Brüninghaus and Ashley 2003) the factors are grouped in five issues and the factors may be present in more than one issue. These issues are similar to our values although for our values, the factors can as yet only be present in one value (This limitation will be addressed when we consider complex structures later in this paper). IBP also has factors of different strengths, KO, normal and weak which can be seen as corresponding to points on dimensions. However IBP does not explicitly order the factors.

8.3. AS A JUSTIFICATION OF WEIGHTS

Each structured value, like each unstructured value, can have a different weight because some are considered more important than others. Also, because a structured value consists of a range of points of differing strengths, the weight can be varied to give each point a different proportion of the weight associated with the structured value.

For example, the strong factor of F27-*Disclosure in a Public Forum* would be given a large proportion of the weight for the defendant from Reasonable Efforts and the weak factor of F10-*Disclosure in Negotiation* would be given a smaller proportion of the weight for that structured value.

In our experiments we wanted to explore three main issues:

- How should we map factors back into structured values?
- Can we identify a plausible relationship between values and structured values?
- Can we use the notion of structured values to produce a principled means of assigning weights to factors?

Our experiments with structured values are described below.

9. Structured value environment

In this section we use two types of Structured Values. On the *Simple Structured Values* a factor can only be present as a point on a single Structured Value, whereas for the *Complex Structured Values* each factor can be present as a point on more than one Structured Value.

9.1. SIMPLE STRUCTURED VALUES

We began with a notion of *Simple Structured Values*, where each factor can only be present on a single structured value. These *Simple Structured Values* were created by using the value groupings from the experiments using factors, as described in Sect. 5.1. Figure 6 shows the grouping of the factors into the structured values.

The structured value is arranged with the strong pro-P factors first, running through the weaker pro-P factors, then changing to the weaker pro-D factors and finally the strong pro-D factors. The structured values do not need to be arranged like this as they can have the defendant factors first; we have, however, chosen a consistent direction for ease of reading. If the structured value only consists of factors which promote one outcome then only this part of the structured value is used. Table VI shows the factors present in each structured value sorted into the correct range for the structured value. The Type column indicates the type of factor as defined in IBP, showing that our ordering is compatible with the partial order implied by the IBP categories.

Because we wish the stronger factors to subsume the weaker factors to avoid double counting, the case descriptions have to reflect this. The weaker factors are removed from the description, leaving only the strongest pro-plaintiff factor and the strongest pro-defendant factor on each structured

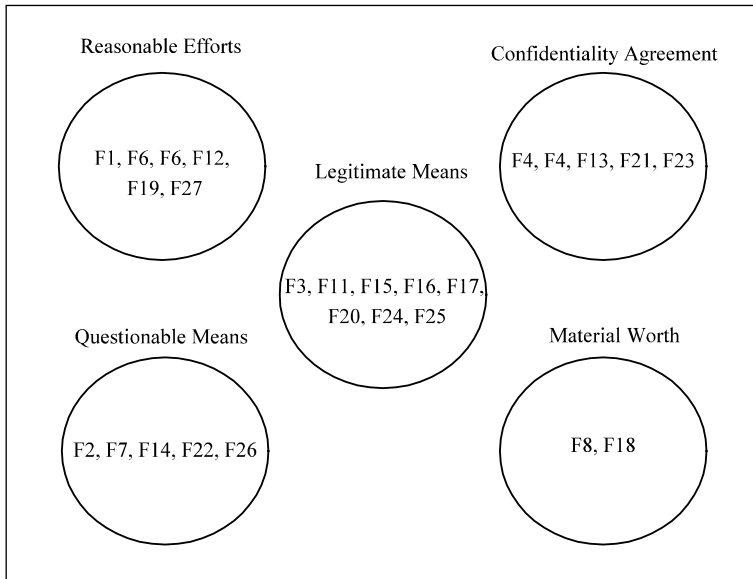


Figure 6. Factors divided into simple structured values and their relationships.

value. Where there are Plaintiff and Defendant factors from the same structured value in the case description then the strongest Plaintiff factor and strongest Defendant factor remain while the weaker points are removed.

For example, in the Arco case, factors F16 and F20 are both on the *Legitimate Means* structured value and because F20 is a stronger factor, it subsumes the weaker F16. Figure 7 demonstrates this.

In the Boeing case, factors F6 and F12 are Plaintiff factors on the *Reasonable Efforts* structured value and factors F1 and F10 are Defendant factors on that structured value. F6 subsumes F12, F10 subsumes F1 and so both of these factors remain in the case description. Also factors F4 and F21 are on the *Confidentiality Agreement* structured value and F4 subsumes F21. Figure 8 demonstrates this.

Table VII shows the new case descriptions with the factor and values separated into Plaintiff and Defendant (cf. Table II which gives all factors).

9.2. COMPLEX STRUCTURED VALUES

When we use complex structured values, these will relate factors which may promote several of our original values. Also factors may appear in several complex structured values. This means that the complex structured values will need to represent new values.

Table VI. Simple structured values

<i>Reasonable Efforts</i>	<i>Type</i>	<i>Questionable Means</i>	<i>Type</i>
F6 Security Measures (p)	Normal	F26 Deception (p)	KO
F12 Outsider Disclosures Restricted (p)	Normal	F22 Invasive Techniques (p)	Normal
F1 Disclosure In Negotiations (d)	Weak	F2 Bribe Employee (p)	Normal
F10 Secrets Disclosed Outsiders (d)	Weak	F14 Restricted Materials Used (p)	Normal
F27 Disclosure In Public Forum (d)	KO	F7 Brought Tools (p)	Normal
F19 No Security Measures (d)	KO		
<i>Confidentiality Agreement</i>	<i>Type</i>	<i>Legitimate Means</i>	<i>Type</i>
F13 Noncompetition Agreement (p)	Normal	F15 Unique Product (p)	Normal
F4 Agreed Not To Disclose (p)	Normal	F16 Info Reverse Engineerable (d)	Weak
F21 Knew Info Confidential (p)	Normal	F25 Info Reverse Engineered (d)	Normal
F5 Agreement Not Specific (d)	Normal	F3 Employee Sole Developer (d)	Normal
F23 Waiver Of Confidentiality (d)	Normal	F11 Vertical Knowledge (d)	Normal
		F24 Info Obtainable Elsewhere (d)	Normal
<i>Material Worth</i>	<i>Type</i>	F17 Info Independently Generated (d)	Normal
F8 Competitive Advantage (p)	KO	F20 Info Known To Competitors (d)	KO
F18 Identical Products (p)	Normal		

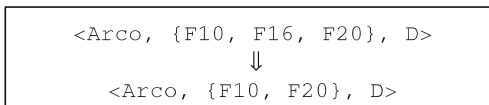


Figure 7. Modification of the Arco Case.



Figure 8. Modification of the Boeing Case.

Table VII. Simple structured value based cases used in CATE

	Pro-P factors	Pro-P Values	Pro-D factors	Pro-D Values	
<i>Group 1</i>					
Arco			F10, F20	RE, LM	D
Boeing	F4, F6, F14	CA, RE, QM	F10	RE	P
Bryce	F4, F6, F18	CA, RE, MW	F1	RE	P
College	F15, F26	LM, QM	F1	RE	P
<i>Watercolour</i>					
Den-Tal-Ez	F4, F6, F26	CA, RE, QM	F1	RE	P
Ecologix	F21	CA	F19, F23	RE, CA	D
Emery	F18, F21	MW, CA	F10	RE	P
Ferranti	F2	QM	F19, F20	RE, LM	D
Robinson	F18, F26	MW, QM	F19	RE	D
Sandlin			F16, F19	LM, RE	D
Sheets	F18	MW	F19	RE	D
Space Aero	F8, F15	MW, LM	F19	RE	P
Televation	F6, F15, F18, F21	RE, MW, CA, LM	F10, F16	LM, RE	P
Yokana	F7	QM	F16, F27	LM, RE	D
<i>Group 2</i>					
CMI	F4, F6	CA, RE	F20, F27	LM, RE	D
Digital	F6, F8, F15, F21	RE, MW, LM, CA	F1	RE	P
<i>Development</i>					
FMC	F4, F6, F7	CA, RE, QM	F10, F11	LM, RE	P
Forrest	F6, F15, F21	RE, LM, CA	F1	RE	P
Goldberg	F21	CA	F27	RE	P
KG	F6, F14, F16, F18, F21	RE, QM, MW, CA, LM	F25	LM	P
Laser	F6, F21	RE, CA	F10	RE	P
Lewis	F8, F21	MW, CA	F1	RE	P
MBL	F6, F13	RE, CA	F5, F10, F20	LM, RE, CA	D
Mason	F6, F15, F21	RE, CA, LM	F1, F16	LM, RE	P
Mineral	F18	MW	F1, F25	RE, LM	P
<i>Deposits</i>					
National	F18, F21	MW, CA	F1	RE	P
<i>Instrument</i>					
National Rejectors	F7, F15, F18	QM, MW, LM	F16, F19	LM, RE	D
Reinforced	F4, F6, F8, F15	CA, RE, MW, LM	F1	RE	P
Scientology	F4, F6	CA, RE	F10, F20	LM, RE	D
Technicon	F6, F14, F21	RE, QM, CA	F10, F25	LM, RE	P
Trandes	F4, F6	CA, RE	F10	RE	P
Valco-Cincinnati	F6, F15, F21	RE, LM, CA	F10	RE	P

We choose four such new values. Honouring Agreements is desirable since if all dealings were regulated by properly drafted agreements, there would be no conflicts for the courts to resolve. A second structured value can relate to interests in security, to enforce the notion that a person with a secret should take Reasonable Efforts to maintain it. Thirdly we wish to avoid litigation where possible, and if information is known generally, there should be no case to answer and hence Less Litigation. Finally we wish to promote Fair Competition, and this gives rise to two structured values, one relating to questionable means taken to obtain the secret, and one to fair methods having been used to develop the product, We thus have five complex structured values with which to organise our factors.

With *Complex Structured Values*, the factors can occur as points on several of these structured values. Factors were assigned to these structured values by analysing the factor descriptions from Aleven (1997). Many factors seem to have several different characteristics and so can be placed on different structured values.

Figure 9 shows the description for F10-*Secrets-Disclosed-Outsiders* as given in Aleven (1997). It has a *Security* characteristic, namely that the plaintiff told outsiders and so failed to show concern for his secret, an *Information Known* characteristic because people outside the plaintiff know the information and a *Fair Method* characteristic because the defendant can obtain the information from the outsiders.

Figure 10 shows how the factors can be grouped into *Complex Structured Values* and also how the structured values overlap and relate to each other.

The structured values are again arranged with strong pro-P factors first, running through the normal and weak pro-P factors, then changing to the weak pro-D factors, then the normal pro-D factors and finally the strong Pro-D factors. If the structured value only consists of factors which promote one outcome then again only this part of the structured value is used.

The factors for each structured value are arranged into their order of strength with the strong factors at the end of the structured value and the

<p>F10 Secrets-Disclosed-Outsiders (d)</p> <p>Description: Plaintiff disclosed its product information to outsiders. This factor shows that plaintiff's information was known in the industry or available from sources outside plaintiff's business. Also, it shows that plaintiff showed a lack of interest in maintaining the secrecy of its information.</p> <p>The factor applies if: Plaintiff disclosed its product information for example to licensees, customers, suppliers, subcontractors, etc.</p> <p>The factor does not apply if: Plaintiff published the information in a public forum. (In that situation, F27 applies.) All we know is that plaintiff marketed a product from which the information could be ascertained by reverse engineering.</p>
--

Figure 9. Factor description of F10 from Aleven (1997).

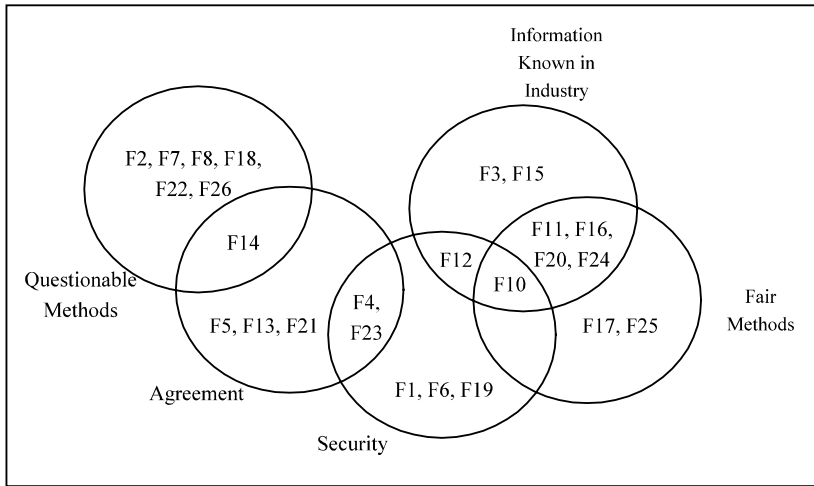


Figure 10. Factors divided into complex structured values and their relationships.

weaker factors towards the centre. The relative ordering of the factors stays the same on all the structured values, so that if a factor is stronger than another factor on one structured value then it will be stronger on all the other structured values on which they both occur. Factors F11 and F16 occur on two structured values and F11 is stronger than F16. In the *Information Known in Industry* structured value they are next to each other, but in the *Fair Methods* structured value they are separated by the inclusion of F25. Table VIII shows the factors present in each structured value sorted into the correct range for the structured value, and the type in Brüningshaus and Ashley (2003). Again our ordering is consistent with the partial order implied by IBP.

The factor based case descriptions need to be modified again to reflect these Complex Structured Values. First, the factors which are always together on all the structured values are compared and the weaker factors subsumed by the stronger factors. Second, the remaining factors are transformed into all the possible structured value points because F10 is on three structured values it is replaced by three structured value points, namely F10sec, F10inf and F10fme. The structured value point is labelled with three extra letters to reflect which structured value it is located on. Third, the points from each structured value are compared and the strong factors subsume the weaker factors and remain in the case description.

In the Arco case F16 and F20 are always together on all the structured values in which they feature because they share the same characteristics. Because F20 is stronger it subsumes F16, leaving F10 and F20. Next they are replaced by the relevant structured value points. F10 is replaced by three

Table VIII. Complex structured values

<i>Interests in Security</i>	<i>Type</i>	<i>Info Known in Industry</i>	<i>Type</i>
F6 Security Measures (p)	Normal	F15 Unique Product (p)	Normal
F4 Agreed Not To Disclose (p)	Normal	F12 Outsider Disclosures Restricted (p)	Normal
F12 Outsider Disclosures Restricted (p)	Normal	F10 Secrets Disclosed Outsiders (d)	Weak
F1 Disclosure In Negotiations (d)	Weak	F16 Info Reverse Engineerable (d)	Weak
F10 Secrets Disclosed Outsiders (d)	Weak	F11 Vertical Knowledge (d)	Normal
F23 Waiver Of Confidentiality (d)	Normal	F24 Info Obtainable Elsewhere (d)	Normal
F27 Disclosure In Public Forum (d)	KO	F3 Employee Sole Developer (d)	Normal
F19 No Security Measures (d)	KO	F20 Info Known To Competitors (d)	KO
		F27 Disclosure In Public Forum (d)	KO
<i>Questionable Methods</i>	<i>Type</i>	<i>Fair Methods</i>	<i>Type</i>
F8 Competitive Advantage (p)	KO	F10 Secrets Disclosed Outsiders (d)	Weak
F26 Deception (p)	KO	F16 Info Reverse Engineerable (d)	Weak
F22 Invasive Techniques (p)	Normal	F25 Info Reverse Engineered (d)	Normal
F2 Bribe Employee (p)	Normal	F11 Vertical Knowledge (d)	Normal
F14 Restricted Materials Used (p)	Normal	F24 Info Obtainable Elsewhere (d)	Normal
F7 Brought Tools (p)	Normal	F17 Info Independently Generated (d)	Normal
F18 Identical Products (p)	Normal	F20 Info Known To Competitors (d)	KO
		F27 Disclosure In Public Forum (d)	KO
<i>Agreements</i>			
F13 Noncompetition Agreement (p)	Normal		
F4 Agreed Not To Disclose (p)	Normal		
F21 Knew Info Confidential (p)	Normal		
F14 Restricted Materials Used (p)	Normal		
F5 Agreement Not Specific (d)	Normal		
F23 Waiver Of Confidentiality (d)	Normal		

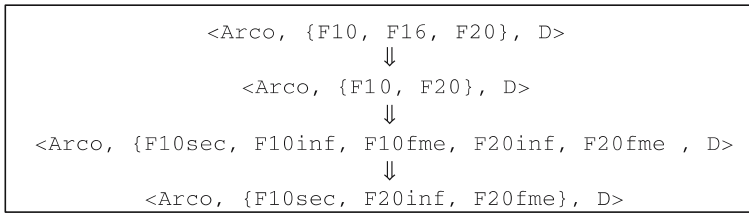


Figure 11. Modification of the Arco Case.

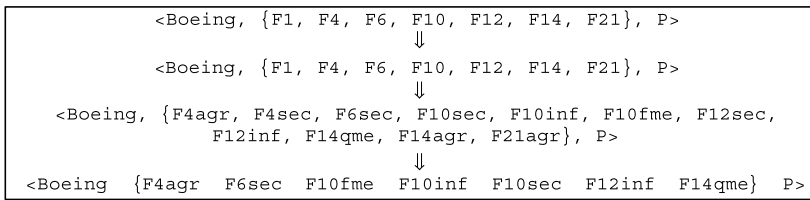


Figure 12. Modification of the Boeing Case.

points and F20 by two points. Finally the stronger F20 points subsume the weaker F10 points on both the *Information Known in Industry* and *Fair Methods* structured values so we are left with the strongest factors from each structured value. Figure 11 shows this process.

In the Boeing case, none of the factors are always together so all the factors are replaced by their relevant structured value points. Now the weaker factors can be subsumed by the stronger factors. Figure 12 shows the process for the Boeing case.

Table IX shows the new case descriptions with the factors and values separated into Plaintiff and Defendant.

10. Constructing structured value theories

Due to the poor results for the simple method (Theory 2) in the experiments based on factors, this method was not used in the Structured Value experiments leaving only two approaches to be studied.

10.1. THE SAFE METHOD

The first of these is the *safe* theory described earlier in Sect. 5.2. For the simple structured values, the same cases can be used as in 5.2, although the actual theory will be different due to the changes in the case description. The Plaintiff cases used are *Emery* and *College Watercolour* and the Defendant cases are *Robinson* and *Sheets*. Representing the rules from these four

Table IX. Complex structured value based cases used in CATE

	Pro-P factors	Pro-P Values	Pro-D factors	Pro-D Values
<i>Group 1</i>				
Arco			F10sec, F20inf, F20fme	RE, LL, FC
Boeing	F4agr, F6sec, F12inf, F14qme	HA, RE, LL, FC	F10fme, F10inf, F10sec	FC, LL, RE
Bryce	F4agr, F6sec, F18qme	HA, RE, FC	F1sec	RE
College Watercolour	F15inf, F26qme	LL, FC	F1sec	RE
Den-Tal-Ez	F4agr, F6sec, F26qme	HA, RE, FC	F1sec	RE
Ecologix	F21agr	HA	F19sec, F23agr	RE, HA
Emery	F18qme, F21agr	FC, HA	F10sec, F10inf, F10fme	RE, LL, FC
Ferranti	F2qme	FC	F19sec, F27inf, F27fme	RE, LL, FC
Robinson	F26qme	FC	F10inf, F10fme, F19sec	LL, FC, RE
Sandlin			F19sec, F27inf, F27fme	RE, LL, FC
Sheets	F18qme	FC	F19sec, F27inf, F27fme	RE, LL, FC
Space Aero	F8qme, F15inf	FC, LL	F19sec	RE
Television	F6sec, F15inf, F18qme, F21agr	RE, FC, HA, LL	F10sec, F16inf, F16fme	LL, FC, RE
Yokana	F7qme	FC	F27sec, F27inf, F27fme	RE, LL, FC

<i>Group 2</i>					
CMI	F4agr, F6sec F6sec, F8qme, F15inf, F21agr F4agr, F6sec, F7qme, F12inf F6sec, F15inf, F21agr F21agr F6sec, F14qme, F15inf, F21agr F6sec, F12inf, F21agr F8qme, F21agr F6sec, F13agr F6sec, F15inf, F21agr F18qme F18qme, F21agr F7qme, F15inf F4agr, F6sec, F8qme, F15inf F4agr, F6sec, F12inf F6sec, F12inf, F14qme, F21agr F4agr, F6sec, F12inf F6sec, F15inf, F21agr	HA, RE RE, FC, LL, HA HA, RE, FC, LL RE, LL, HA HA RE, FC, HA, LL RE, LL, HA FC, HA RE, HA RE, HA, LL FC FC, HA FC, LL HA, RE, FC, LL HA, RE, LL RE, FC, HA, LL HA, RE, LL RE, LL, HA	F27fme, F27inf, F27sec F1sec F11inf, F11fme, F10sec F1sec F27fme, F27inf, F27sec F16inf, F25fme F10fme, F10inf, F10sec F1sec F5agr, F10sec, F20fme, F20inf F1sec, F16inf, F16fme F1sec, F16inf, F25fme F1sec F19sec, F27inf, F27fme F1sec F10sec, F20inf, F20fme F10sec, F16inf, F25fme F10fme, F10inf, F10sec F10fme, F10inf, F10sec	FC, LL, RE RE LL, FC, RE RE FC, LL, RE LL, FC FC, LL, RE RE FC, LL, HA, RE LL, FC, RE RE, LL, FC RE LL, FC, RE LL, FC, RE FC, LL, RE FC, LL, RE	D P P P P P P P P D P P P D P P P P

```

Theory Preferences :
  pref(<{F15, F26}, P>, <{F1}, D>)
    <|<CollegeWatercolour, {F1, F15, F26}, P>|>
  pref(<{F18, F21}, P>, <{F10}, D>)
    <|<Emery, {F10, F18, F21}, P>|>
  pref(<{F19}, D>, <{F18, F26}, P>)
    <|<Robinson, {F18, F19, F26}, D>|>
  pref(<{F19}, D>, <{F18}, P>)
    <|<Sheets, {F18, F19}, D>|>
Theory Value Preferences :
  valpref({CA, MW}, {RE})
  valpref({LM, QM}, {RE})
  valpref({RE}, {MW, QM})
  valpref({RE}, {MW})

```

Figure 13. Rule and value preferences for the “Safe” method using the Simple Structured Values.

cases in the manner of Prakken and Sartor (1998) yields the rule and value preferences shown in Figure 13.

For the complex structured values, it is more difficult to choose cases. Our aim is to get the simplest possible value preferences. Now that factors can appear related to several values in many cases, incoherence frequently arises, with a value being required to be preferred to itself. This means there are only two cases suitable to construct the theory, *College Watercolour* and *Space Aero*. Using these two cases yields the rule and value preferences shown in Figure 14.

10.2. THE VALUE DRIVEN METHOD

For *Simple Structured Values*, because they use the same values as were used with factors, the same cases can be used, yielding the same rule and value preferences. *Televation* is used to represent the value preference of CA > LM, *Space Aero* is used to represent the value preference of LM > RE, *Robinson* is used to represent the value preference of RE > QM and finally *Sheets* is used

```

Theory Preferences :
  pref(<{F15inf, F26qme}, P>, <{F1sec}, D>)
    <|<CollegeWatercolour, {F15inf, F1sec,
F26qme}, P>|>
  pref(<{F15inf, F8qme}, P>, <{F19sec}, D>)
    <|<SpaceAero, {F15inf, F19sec, F8qme}, P>|>
Theory Value Preferences :
  valpref({FC, LL}, {RE})

```

Figure 14. Rule and value preferences for the “Safe” method using the Complex Structured Values.


```

Theory Preferences :
  pref(<{F15}, P>, <{F19}, D>
    <|<SpaceAero, {F15, F19, F8}, P>|>
  pref(<{F19}, D>, <{F18}, P>)
    <|<Sheets, {F18, F19}, D>|>
  pref(<{F19}, D>, <{F26}, P>)
    <|<Robinson, {F18, F19, F26}, D>|>
  pref(<{F21}, P>, <{F16}, D>)
    <|<Televation, {F10, F15, F16, F18, F21, F6}, P>|>
Theory Value Preferences :
  valpref({CA}, {LM})
  valpref({LM}, {RE})
  valpref({RE}, {MW})
  valpref({RE}, {QM})

```

Figure 15. Rule and value preferences for the “Value Driven” method using the Simple Structured Values.

to represent the value preference of $RE > MW$. The rule and value preferences are given in Figure 15.

However, due the change in the values for the complex structured values, we must change the theory to reflect this. Let us suppose that the most highly rated value is *Honouring Agreements*, since if all dealings were regulated by properly drafted agreements, there would be no problem to decide. Let us rate the value of *Less Litigation* next; we want to stop frivolous court cases wasting time and money. We rate the value of *Reasonable Efforts* third, since people must take some steps to protect themselves. This leaves the value of *Fair Competition* last.

Because there are only four values, only three cases are needed to represent the value preferences. For the value preference of $HA > LL$ *Emery* is used with F10inf and F21agr. For the value preference of $LL > RE$ *College Watercolour* is used with F1sec and F15inf. Finally for $RE > FC$ *Robinson* is used with F19sec and F26qme. These cases yield the rule and value preferences shown in Figure 16.

```

Theory Preferences :
  pref(<{F15inf}, P>, <{F1sec}, D>
    <|<CollegeWatercolour, {F15inf, F1sec, F26qme}, P>|>
  pref(<{F19sec}, D>, <{F26qme}, P>)
    <|<Robinson, {F10fme, F10inf, F19sec, F26qme}, D>|>
  pref(<{F21agr}, P>, <{F10inf}, D>)
    <|<Emery, {F10fme, F10inf, F10sec, F18qme, F21agr}, P>|>
Theory Value Preferences :
  valpref({HA}, {LL})
  valpref({LL}, {RE})
  valpref({RE}, {FC})

```

Figure 16. Rule and Value Preferences for the “Value Driven” method using the Complex Structured Values

11. Comparing and weighting factors

11.1. COMPARING FACTORS

The four methods for comparing factors described in Sect. 6 are again used in our Structured Value experiments. These are Best Factor, Best NonShared Factor, Exceptions and CATO.

11.2. WEIGHTING FACTORS

The four methods of weighting factors used in Sect. 6 are again used with an additional method using the structured values to adjust the weight given to the factor.

For this new method the structured value is divided into 20 slots, 10 plaintiff slots and 10 defendant slots and the factors contained on the structured value can be placed in any of the slots. Where they are placed depends on how strong the factor is. Knockout factors are placed at the ends of the structured value and receive the largest proportion of the weight. Weak factors are placed at the centre of the structured value and receive the smallest proportion of the weight, while normal factors are placed midway and receive intermediate weights. The factors can be moved along the structured value until the best position is found. We allow gaps, respecting the possibility that more factors could be introduced if finer grained points of discrimination are thought necessary. A factor in slot 10 receives one tenth of the weight, in slot 9 it receives two tenths and so on until slot 1 where it receives the whole weight. Table X shows the *Reasonable Efforts* structured

Table X. Structured value weighting for the reasonable efforts structured value

Plaintiff end	Weight	Change over point	Weight	
Slot 1	0.3	Slot 10	F1	-0.03
Slot 2	0.27	Slot 9	F10	-0.06
Slot 3	F6	Slot 8		-0.09
Slot 4	F12	Slot 7		-0.12
Slot 5	0.18	Slot 6		-0.15
Slot 6	0.15	Slot 5		-0.18
Slot 7	0.12	Slot 4		-0.21
Slot 8	0.09	Slot 3		-0.24
Slot 9	0.06	Slot 2	F27	-0.27
Slot 10	0.03	Slot 1	F19	-0.3
Change over point		Defendant end		

value with its factors placed into their slots. *Reasonable Efforts* has a maximum weight of 0.3 to reflect its importance relative to the other structured values. Here we are able to reflect that F6 and F12 are normal factors, F1 and F10 weak factors and F19 and F27 are KO factors, by positioning them in different parts of the range. The factor F6 receives eight tenths of the weight and as the weight given to Reasonable Effort is 0.3, F6 has a weight of 0.24. F12 is slightly weaker and only receives a weight of seven tenths or 0.21. Were we to discover a KO plaintiff factor for this structured value it could be placed in Slots 1 or 2.

12. Experiments with structured values

12.1. COMPARISON OF THEORY CONSTRUCTION METHODS

The main results of these initial experiments are depicted in Tables XI and XII. The theories here use all the factors available in the background. The comparison is therefore with the “All” columns of Tables III and IV. The tables show the number of cases not correctly classified by the theory, including abstentions as well as misclassifications. The number in brackets show the failures to classify before refinement, for those cases where refinement was used.

Table XI. Results for the 14 Group 1 cases

	Simple		Complex	
	Theory 1	Theory 3	Theory 1	Theory 3
Best factor	0 (3)	0 (1)	0 (5)	0 (1)
Cancellation	0 (1)	0	0 (1)	0
Exceptions	0	0	0 (4)	0
CATO	0 (1)	0	0 (2)	0 (1)

Table XII. Results for the 18 Group 2 cases.

	Simple		Complex	
	Theory 1	Theory 3	Theory 1	Theory 3
Best factor	5 (12)	5 (5)	4 (11)	13 (13)
Cancellation	5 (4)	3	3 (8)	3
Exceptions	5	4	4 (3)	4
CATO	3 (2)	4	4 (4)	4 (4)

As with factors it is possible, perhaps with some refinement, to produce theories to explain the training set of cases using either method of theory construction and any of the methods of comparison. For the test set, when using *Simple Structured Values*, the structured values method performs worse for Theory 1 than the corresponding methods using factors, except when weights are determined using the information from CATO. This suggests that the subsumption of factors has had some negative effect. Cases represented with *Simple Structured Values* typically contain fewer factors than when represented using factors directly. Since here weights are not used, it may be that removing the subsumed factor understates the contribution of that structured value. Theory 3 performs as well or better than Theory 1 except for the CATO comparison method, and performs comparably to using factors rather than structured values. When using *Complex Structured Values*, Theory 1 and Theory 3 perform equally well except for the Best Factor comparison method where Theory 3 gets very bad results. This might be explained because there is a mismatch between the finer grained representation and the broad brush comparison technique. For the remaining comparison methods, *Complex Structured Values* never perform worse than factors, and both Theories obtain better results than when using factors if the cancellation method is used.

Remember, however, that we are here using unweighted factors, and the weighting implied by the position within a structured value is an important part of the rationale of using structured values. We might therefore expect problems to arise where we have the possibility of cases with a point near the middle of a strongly valued structured value and at one extreme of a weakly valued structured value. We therefore conclude, that if no account of weighting is to be taken, there is little to be gained from thinking in terms of structured values rather than values. We turn to experiments with weights in the next section.

12.2. COMPARISON OF WEIGHTING

Results for the different methods of using weights are summarised in Table XIII. The cells show the number of cases misclassified by the theory using all 32 cases, groups 1 and 2.

For *Simple Structured Values* and Theory 1 the Weighted Values and Weighted Factors methods and the Structured Value Weights method perform the best. When using Theory 3, the Structured Value Weights method performs best. With only a single misclassified case, these represent the best performance from any of our experiments. For *Complex Structured Values*, both Theory 1 and Theory 3 perform identically, and the Structured Value Weights and IBP methods are the best performing comparison methods. For

Table XIII. Results for different methods of weighting factors

	Simple		Complex
	Theory 1	Theory 3	Theory 1 and 3
Weighted values	1	3	3
Weighted factors	1	3	3
Exceptions	4	4	4
IBP	2	2	2
Structured value	1	1	2

all the different versions the Exceptions method always performs worst of all the methods. We should therefore discount this as a method of assigning weights.

Note, however, the Structured Value Weights method, using the technique described in Sect. 11.2 is always one of the best performing methods. This may partly result from the ability to tune the weights more accurately, and clearly to differentiate between normal, weak and KO factors, but it does suggest that structured values offer a sensible way to structure factors so as to assign weights.

12.3. COMPARISON OF SIMPLE AND COMPLEX STRUCTURED VALUES

When using unweighted factors, better performance can typically be obtained from *complex structured values*, but more refinement is required. When using Theory 3 *Simple Structured Values* for the training set of Group 1 cases, only the Best Factor method needs refining whereas *Complex Structured Values* need the CATO method refining as well. Apart from the Best Factor method where the *Complex Structured Values* performs badly, both versions perform equally. As stated above, however, there seems little gain for unweighted factors in moving to structured values.

For methods using weights, *Simple Structured Values* and Theory 1 perform best, with all the other versions performing equally well. One theory as to why *Simple Structured Values* perform better than *Complex Structured Values*, is that while a factor may relate to several values, only one of these is germane to a particular case. Thus although F27 *Disclosure-in-a-Public-Forum* contains elements of disregard for the secret, and fair methods in that the defendant is making use of information in the Public Domain, in the context of a particular case only one of these

may matter. And since F27 is a KO factor, allowing it to appear on more than one structured value may distort its impact. Alternatively it could be that the loss of the connection to the five issues identified by IBP from the Restatement of Torts is important, supporting the argument of Ashley and Brüninghaus (2003) that the intermediate concepts representing these issues are of vital importance for prediction. That the Structured Value Weights method provides best results indicates that the position of factors on the structured values can provide a sensible basis for assigning weights. Moreover the results show that the placing of the factors on the structured value, using information from IBP, seems to be broadly correct.

We wanted to explore three questions with respect to structured values. Obviously the smallness of the sample size precludes firm conclusions, but we offer the following tentative answers.

- *How should we map factors back into structured values?* We have described two methods of performing this mapping. Our results suggest that using simple structured values is enough.
- *Can we identify a plausible relationship between values and structured values?* Since *Simple Structured Values*, which correspond directly to the values identified in Sect. 5.1 perform at least as well as the complex structured values, the suggestion is that we can indeed use our values to supply the structured values.
- *Can we use the notion of structured values to produce a principled means of assigning weights to factors?* The result that the use of structured values to determine weights of factors produced the best results whatever the method of theory construction, suggests that this is so.

13. Conclusions

In this paper we have first described CATE, a tool which supports the rapid construction, evaluation and comparison of theories intended to explain bodies of case law. We have illustrated the use of CATE with a number of experiments designed to explore the theoretical account of reasoning with legal cases as theory construction described in Bench-Capon and Sartor (2003). We summarise our findings below, always remembering that they must be tentative given the smallness of the size of the sample available, and that our analysis in terms of the values chosen and the assignment of factors to values, whilst plausible and with clear mappings to other analyses such as that used in IBP, is not beyond dispute.

- We found that including all factors available from the background produced better performance than being selective as to factors. Since we are starting from the analysis of Alevén (1997), we can expect all factors

to be relevant. In a less well analysed domain it could prove useful to select factors.

- We were able to improve performance using additional factors and determining the priority to be given to these new factors by reference to value priorities established using different factors related to that value. This supports the suggestion of Bench-Capon and Sartor (2003) that the importance of factors does relate to their motivating values.
- On comparison of sets of factors we found it best to take all the factors present in the sets into account. Factors appear to have some cumulative effect
- The improvements obtainable by weighting values and factors, suggest that factors do support the case of the party they favour to different degrees.
- We found that structuring the factors into structured values corresponding to values provided an effective and principled way to assign weights to factors.

We did not, however, come to any conclusions as to the best method for the construction of theories. Quite possibly there is no general principle which can be applied.

This issue is being explored in our current work, in which we are seeking ways to automate the construction of theories which do not rely on the application of some general principle but which are driven by particular case situations. A preliminary report of this work was given in Chorley and Bench-Capon (2004b). Our idea there is to construct theories using a sequence of argument moves of the sort found in systems such as HYPO and CATO. Each move is associated with a set of theory constructors, and thus as the moves are made, the theory is constructed as a side effect. Now, by modelling the process of reasoning as a two player game in which the plaintiff and defendant alternately make argument moves, we can construct a game tree, which will also correspond to the space of possible theories which can be constructed using these moves. Since this space rapidly becomes very large, we use a heuristic search based on A* and an evaluation function for theories. We have obtained some very promising results using this method, and, as reported in Chorley and Bench-Capon (2005), theories giving performance comparable to that of IBP can be attained. We will next look at adversarial search techniques to see whether this offers any differences or improvements over the A* like search. A further important issue which will be explored in future work is whether the quality of the explanation produced matches the quality of the prediction, since in case law it is often possible to get good prediction with methods that produce bad legal explanations.

Note

¹ Here and elsewhere in this paper we use typically use “explanation” in the scientific sense whereby a theory explains those phenomena which are predicted by the theory. This is in contrast to the notion of explanation common in law which requires a reasoned argument based on statute, cases and doctrine.

References

- Aleven, V. (1997). *Teaching Case Based Argumentation Through an Example and Models*, PhD Thesis, The University of Pittsburgh.
- Ashley, K. D. (1990). *Modelling Legal Argument*. Bradford Books, MIT Press: Cambridge, Mass.
- Ashley, K. and Brüninghaus, S. (2003). A Predictive Role for Intermediate Legal Concepts. In *Proceedings Jurix 03*, 153–162. IOS Press: Amsterdam.
- Ashley, K. D. (2004). Case-Based Models of Legal Reasoning in a Civil Law Context. Invited paper. International Congress of Comparative Cultures and Legal Systems of the Instituto de Investigaciones Jurídicas, Universidad Nacional Autónoma de México, Mexico City.
- Bench-Capon, T. J. M. (1991a). Practical Legal Expert Systems: the Relation Between a Formalisation of Law and Expert Knowledge. In Bennun and Narayanan (eds.), *Computers*, 191–201. Law and AI, Ablex.
- Bench-Capon, T. J. M. (1991b). Knowledge Based Systems Applied To Law: A Framework for Discussion, In Bench-Capon, T. J. M. (ed.), *Knowledge Based Systems and Legal Applications*, 329–342, Academic Press.
- Bench-Capon, T. J. M. (2002). The Missing Link Revisited: The Role of Teleology in Representing Legal Argument. *Artificial Intelligence and Law* 10(2–3): 79–94.
- Bench-Capon, T. J. M. and Rissland, E. L. (2001). Back to the Future: Dimensions Revisited: In Verheij, B., Lodder, A. R., Loui, R. P. and Muntjewerff, A. J. (eds.), *Legal Knowledge and Information Systems*, 41–52. IOS Press: Amsterdam.
- Bench-Capon, T. and Sartor, G. (2003). A model of legal reasoning with cases incorporating theories and values. *Artificial Intelligence* 150(1–2): 97–143.
- Berman, D. H. and Hafner, C. D. (1993). Representing Teleological Structure in Case Based Reasoning: The Missing Link. In *Proceedings of the Fourth International Conference on AI and Law*, 50–59, ACM Press, New York.
- Brüninghaus, S and Ashley, K. D. (2003). Predicting Outcomes of Case-based Legal Arguments. *Proceedings of the Ninth International Conference on AI and Law*, 233–242. ACM Press: New York.
- Chorley, A. and Bench-Capon, T. (2003a). Developing Legal Knowledge Based Systems Through Theory Construction. Technical Report ULCS-03–013, Department of Computer Science, The University of Liverpool, <http://www.csc.liv.ac.uk/research/techreports/tr2003/ulcs-03–013.pdf>.
- Chorley, A. and Bench-Capon, T. (2003b). Developing Legal Knowledge Based Systems Through Theory Construction. *Proceedings of the Ninth International Conference on AI and Law*, 85–86. ACM Press: New York.
- Chorley, A. and Bench-Capon, T. (2003c). Reasoning with Legal Cases as Theory Construction: Some Experimental Results: In Bourcier, D. (ed.), *Proceedings of Jurix 2003*, 173–182. IOS Press: Amsterdam.

- Chorley, A. and Bench-Capon, T. (2004a). Support for Constructing Theories in Case Law Domains: In Galindo, F., Takizawa, M. and Traunmuller, R. (eds.), *Proceedings of DEXA 2004*. LNCS 3180, 508–517. Springer Verlag: Berlin.
- Chorley, A. and Bench-Capon, T. (2004b). AGATHA: Automation of the Construction of Theories in Case Law Domains. In *Proceedings of Jurix 2004*, 89–98. IOS Press Amsterdam.
- Chorley, A. and Bench-Capon, T. (2005). AGATHA: Automated construction of Case Law Theories Through Heuristic search. In *Proceedings of the Tenth International Conference on AI and Law*, 45–54. ACM Press: New York.
- Hage, J. C. (2001). Formalizing legal coherence. *Proceedings of the 8th International Conference on Artificial Intelligence and Law*, 22–31. ACM: New York.
- Levi, E. H. (1949). *An Introduction to Legal Reasoning*. University of Chicago Press: Chicago.
- Lipinski, T. A. and Britz, J. J. (2000). Rethinking the ownership of information in the 21st century: Ethical implications, 2: 49–71.
- MacCormick, D. N. and Summers, R. S. (eds.), (1997). *Interpreting Precedents: A Comparative Study*. Dartmouth Publishing: Aldershot, U.K.
- McCarty, L.T. (1995). An Implementation of Eisner v Macomber. In *Proceedings of the Fifth International Conference on AI and Law*, 276–286. ACM Press: New York.
- Prakken, H. (2000). An Exercise in Formalising Teleological Case Based Reasoning: In Breuker, J., Leenes, R. and Winkels, R. (eds.), *Legal Knowledge and Information Systems: Jurix 2000*, 49–57. IOS Press: Amsterdam.
- Prakken, H. (2002). An Exercise in Formalising Teleological Case-Based Reasoning, 10(1–3): 113–133.
- Prakken, H. and Sartor, G. (1998). Modelling Reasoning with Precedents in a Formal Dialogue Game, 6: 231–287.
- Rissland, E. and Ashley, K. (2002). A Note on Dimensions and Factors, 10(1–3): 65–77.
- Sartor, G. (2002). Teleological Arguments and Theory-Based Dialectics, 10(2–3): 95–112.
- Sergot, M. J., Sadri, F., Kowalski, R. A., Krivaczek, K., Hammond, P. and Cory, H. T. (1986). The British Nationality Act as A Logic Program, 29(5): 370–386.
- Skalak, D. and Rissland, E. (1992). Arguments and Cases: An Inevitable Intertwining, 1(1): 3–44.