Combining Modes of Reasoning: an Application of Abstract Argumentation

Henry Prakken

Department of Information and Computing Sciences, Faculty of Science, Utrecht University & Faculty of Law, University of Groningen, The Netherlands

Abstract. Many reasoning problems involve subproblems that can be solved in different ways. Therefore, hybrid reasoning architectures have long been a research topic in AI. However, most work in this area has either focused on particular combinations of reasoning methods or has ignored the problem of handling alternative solutions to subproblems. The present paper proposes an abstract framework for combining modes of reasoning and handling alternative solutions. It is argued that current abstract argumentation systems are either too abstract or too specific for this purpose, so that an intermediate level of abstraction is needed.

1 Introduction

Many reasoning problems involve subproblems that can be solved in different ways. Legal reasoning provides a good example. First there is the problem of determining the facts. This problem can, for instance, be modelled with statistical methods [1], as abduction [2], as a problem of explanatory coherence [3] or as argumentation [4]. Then there is the problem of classifying the facts under the conditions of legal rules. This has been modelled, for instance, as case-based reasoning [5], as argumentation [6], and with neural nets [7]. Finally, when a rule's conditions have been satisfied, it must be applied. Since legal rules can have exceptions, this is often modelled as default reasoning [8].

Hybrid reasoning architectures have therefore long been a research topic in AI. However, most work in this area has either focused on particular combinations of formalisms, such as different logics or logics and probability theory (e.g. [2]), or on general methods for combining (monotonic) logics (e.g. [9]). The present paper instead proposes an abstract framework in which any set of problem solving methods can be combined, whether logical, probabilistic, connectionist or otherwise. Also, unlike the work on general techniques for combining logics, the aim is not to combine different methods into a single new one but to define a framework for defining intput-output relations between methods, while keeping the individual methods as they are. A final distinguishing feature of the present approach is the handling of *alternative* solutions to subproblems. A main motivation for this is the observation that in practical applications of KR & R formalisms often the hardest part is the *modelling* of a problem. While traditionally this is regarded as a knowledge engineering problem, in certain cases it is fruitful to explicitly model disagreements about the modelling of a problem.

I will in particular investigate the suitability of current abstract argumentation systems. The idea is that since these systems abstract from the nature of arguments, they could be suitable for combining different modes of reasoning. The most abstract argumentation system is that of Dung [10], which assumes nothing but a set of arguments with unspecified structure and an abstract relation of attack between arguments. However, this is too abstract for our purposes, since we also need support relations between arguments, to capture dependencies between subproblems. Abstract support relations have been added to [10]'s framework by [11] but this is still insufficient, since for giving guidance on how modes of reasoning can be combined, an account is needed of the nature of support and attack relations between modes of reasoning.

Still less abstract are systems with abstract inference rules, such as OSCAR [12], Abstract Argumentation Systems [13] and Carneades [14]. These systems model reasoning as the application of inference rules defined over some logical language. The systems are abstract in that nothing else is assumed on the nature of inference rules except that they are either strict (beyond attack) or defeasible (prone to attack). Support relations between arguments are captured in inference trees as usual in logic, and conflict relations are defined with logical negation.

At first sight, it would seem that the abstraction from specific sets of inference rules makes these systems suitable for combining modes of reasoning. However, it is not obvious that all modes of reasoning can be fruitfully modelled in the format of defeasible inference rules, let alone that their combination can be modelled as the combination of such rules: inference rules relate propositions, while we need to relate applications of possibly quite complex modes of reasoning.

Yet a strong point of rule-based argumentation systems is that they offer a well-founded formal machinery for managing conflicts given dependencies between issues. The present paper can be seen as an attempt to retain these strong points without having to commit to a modelling in terms of inference rules. This will be achieved by making the formalism isomorphic to the framework of [12]. Since that framework is known to be an instance of [10]'s abstract framework, the theory developed on abstract argumentation in the last thirteen years also applies to the present formal framework. The new contribution is that the elements of Pollock's framework will be given a different interpretation than in [12]: instead of propositions connected by inference rules, we will have applications of problem solving methods connected by output/input transformers.

Below in Section 2 the running example of this paper will be introduced, after which in section 3 the formalism is defined and illustrated with the running example. A discussion and concluding remarks are provided in Section 4.

2 A running example

Throughout this paper an imaginary legal case will be used as a running example. A criminal court has to decide whether a suspect is guilty of murder. The main rule of the criminal code is that someone who kills with intent is guilty of murder. This rule has two main statutory exceptions, namely, that there was some ground of justification for the killing (for instance, self-defence) and that the killer cannot be held responsible for the killing (for instance, because he was insane). Therefore, application of the main rule can arguably best be modelled in a logic for reasoning with default rules. Let us, since it is so well known, use default logic as this logic. It divides input information into facts F, which are propositional or first-order formulas, and defaults D, which are domain-specific inference rules of the form $P{:}Q/R$, in which P, Q and R are propositional or first-order formulas. P is the prerequisite and R the consequent of the default, while Q is its justification. The informal meaning of a default is that R is derivable if P is derivable and Q is consistent with what is derivable. If defaults conflict, then the derivation process branches into alternative conclusion sets ('extensions'). The formal definition of extensions is rather involved but need not be explained here since our example is quite simple and intuitive. A formula is skeptically implied by a default theory if it is in all its extensions and it is credulously implied by the theory if it is in some but not all of its extensions. Exceptions to a default can be modelled as defaults whose consequents contradict its justification.

To apply the default theory to the issue of murder, information is needed about whether the suspect was the killer. The reasoning about this issue will be modelled as an application of Bayesian probability theory. The evidence is that the DNA of the suspect matches the DNA of the killer found at the crime scene. The conditional probability of a random match with that DNA given that a person is not the killer is given as 1 in 2 million. What we want to compute is the posterior conditional probability that the suspect is the killer given the match of his DNA with that of the killer.

This is not all, since to transform a posterior probability into an element of a default theory we need to apply proof standards. For simplicity this will be modelled as logical reasoning with a simple consistent first-order theory, saying that for criminal issues the proof standard is 0.99 (making precise 'beyond reasonable doubt') while for civil issues it is 0.51 (making precise 'on the balance of probabilities'). In fact there is an ongoing debate on how in legal applications of Bayesian statistics these legal proof standards should be defined [1]: such a debate could be modelled as a problem using some argumentation method.

We are still not done. Bayesian statistics requires prior probabilities to derive posterior probabilities from conditional ones and in our example there happens to be disagreement between the experts on what is the correct prior probability that the suspect is the killer. This disagreement will be modelled as an application of the method of first-order default logic (first-order since we want to talk about numerical values of variables). Discussions about prior probabilities are in fact very common in legal applications of Bayesian statistics [1]. Here the example will for reasons of space be kept very simple. The main disagreement is about the population of the potential suspects (let us assume that each of them has equal probability of being the murderer). One point of view is: 'the group of potential suspects consists of all male inhabitants over 16 of town X, which is 50.000 people', while another position is 'the group of potential suspects consists of all male inhabitants over 16 of quarter Y of town X, which is 10.000 people'.

With this information, it is convenient to use the odds formulation of Bayes' theorem:

$$\frac{\Pr(h|e)}{\Pr(\neg h|e)} = \frac{\Pr(h)}{\Pr(\neg h)} \times \frac{\Pr(e|h)}{\Pr(e|\neg h)}$$

Now we are interested in the prior odds that the suspect is the killer (k), in

$$\frac{\Pr(k)}{\Pr(\neg k)}$$

3 The formal framework

As illustrated by our running example, the main idea is that an overall problem is decomposed into several related subproblems, each solved in so-called problem treatments with possibly different methods. The input of a problem treatment is partly given and partly obtained from the output of one or more other problem treatments by so-called output-input transformers. A problem decomposition can also have conflict relations between its problem treatments, namely when they are alternative ways to model the same 'informal' problem, reflected by mutually contradictory input.

An important distinction of the framework is that between the object and metalanguages of problem solving methods. Different subproblems may be formulated in different object languages, such as propositional, first-order or modal logical languages, the language of probability theory, and so on. Yet the formalism defined below assumes that the input and output of all problems is described in a first-order language. This is since the output/input transformers operate on and produce first-order metalevel descriptions of the output and input of problem solving methods.

Let us now formalise these ideas.

Definition 1. *A* problem solving method M *is a tuple* $(\mathcal{L}^I_M, \mathcal{L}^O_M, R_M)$ *where*

- $\mathcal{L}^I_M$ *and* $\mathcal{L}^O_M$*, both first-order languages, are the input language and output language of* M*;*
- R_M *is a function from the powerset of* $\mathcal{L}^I_M$ *into the powerset of* $\mathcal{L}^O_M$*.*

The function R_M is the heart of a method M, specifying how M relates input to output. A method is applied in a 'problem treatment', which applies R_M to a set of problem input statements to produce a set of problem solutions:

Definition 2. *A* problem treatment *is a tuple* $\langle M, I^g, I^d, S\rangle$ *where*

- M *is a problem solving method;*
- $I^g \subseteq \mathcal{L}^I_M$ *is the given problem input;*
- $I^d \subseteq \mathcal{L}^I_M$ *is the derived problem input;*
- $I^g \cup I^d$ *is consistent;*
- $S \subseteq \mathcal{L}^O_M$ *are the problem solutions, such that* $S \subseteq R_M(I^g \cup I^d)$*.*

The elements of a problem treatment P will also be denoted by $M(P)$, $I^g(P)$, $I^d(P)$ and $S(P)$.

In applications of this definition the given input should include general constraints that ensure that the problem-specific input is of the required type. For example, if R is the reasoning of probability theory, then the constraints should ensure that the input is a probability distribution according to the laws of probability theory; this ensures, for instance, that two probability statements $\Pr(A) = x$ and $\Pr(A) = y$ (or $\Pr(A|B) = x$ and $\Pr(A|B) = y$) are inconsistent if $x \neq y$. In the rest of this paper such constraints will be left implicit if there is no danger for confusion.

Note that since the input language of a method is regarded as its metalanguage, consistency of a problem's input does not imply that it specifies a consistent theory. For example, the statement 'T contains p and $\neg p$' is a consistent first-order description of an inconsistent propositional theory T.

The four problem treatments of our running example are as follows. First the ultimate problem treatment P_1 on the issue of murder is given.

- $M(P_1)$ is propositional default logic (*MPropDL*) with the constraint that if $\varphi \in F$, then $\top : \varphi/\varphi \notin D$. It takes as input the specification of a propositional default theory $\Delta = (F, D)$.
- $I^g(P_1) =$
  $k \ \wedge \ i : \neg e_1 \ / \ m \in D$
  $k : \neg e_2 \ / \ i \in D$
  $(k \ \wedge \ i \ \wedge \ j) \supset e_1 \in F$
  $(k \ \wedge \ i \ \wedge \ \neg r) \supset e_1 \in F$
- We are interested in solutions that are ground instances of the formula 'm is x-implied by Δ', where x can take the values 'credulously', 'skeptically' or 'not', and Δ is the default theory specified by $I^g(P_1) \cup I^d(P_1)$.

The first default says that someone who kills (k) with intent (i) is guilty of murder (m) unless there is an exception to this statutory rule (e_1). The second default expresses the commonsense generalisation that killing is normally ($\neg e_2$) done with intent (for simplicity, exceptions to this rule are not listed). The facts in F contain the two exceptions to the rule on murder.

The problem treatment P_2 about the likelihood that the suspect is the killer is specified as follows.

- $M(P_2)$ is Bayesian probability theory (*MBayes*), taking as input probability distributions Π and sets E of evidence.
- $I^g(P_2) =$
  $E = \{d\}$
  $\Pr(d|k) = 1 \in \Pi$
  $\Pr(d|\neg k) = 0.0000005 \in \Pi$
- We are interested in solutions that are ground instances of the formula '$\Pr(k|E) \otimes p$ is implied by Π', where $\otimes$ ranges over $\{<, \leq, =, \geq, >\}$, and Π is the probability distribution specified by $I^g(P_2) \cup I^d(P_2)$.

Here d stands for 'DNA of the suspect matches DNA of the killer found at the crime scene' and k stands for 'The suspect is the killer'. The first conditional probability statement in $I^g(P_2)$ expresses that the DNA of the killer will certainly match with the DNA found at the murder scene, while the second one says that the probability of a random match of someone's DNA profile with any sample of DNA (so also with the DNA found at the murder scene) is one in two million.

The third problem treatment P_3 on what is the proof standard for the issue of killing is as follows.

- $M(P_3)$ = first-order predicate logic (*MFOL*), relating first-order theories T to output of the form 'φ is/is not implied by T'.
- $I^g(P_3) =$
  $\forall\varphi(\mathit{CriminalIssue}(\varphi) \supset \mathit{Standard}(\varphi) = 0.99) \in T$
  $\forall\varphi(\mathit{CivilIssue}(\varphi) \supset \mathit{Standard}(\varphi) = 0.51) \in T$
  $\mathit{CriminalIssue}(k) \in T$.
- We are interested in solutions that are ground instances of the formula '$\mathit{Standard}(k) = x$ is implied by T', where $0 \leq x \leq 1$, and T is the first-order theory specified by $I^g(P_3) \cup I^d(P_3)$.

The final problem treatment P_4 on what is the prior probability that the suspect is the killer is defined as follows.

- $M(P_4)$ is first-order default logic (*MFolDL*) with the constraint that if $\varphi \in F$, then $\top : \varphi/\varphi \notin D$, taking as input first-order default theories Δ and producing as output formulas of the form 'φ is x-implied by Δ' (where x can take the values 'credulously', 'skeptically' or 'not').
- $I^g(P_4) =$
  $male16X(k) : \neg e_1 \ / \ N = 50.000 \in D$
  $male16YX(k) : \neg e_2 \ / \ N = 10.000 \in D$
  $N = x : \neg e_3 \ / \ \mathit{priorodds}(k) = \frac{1}{x-1} \in D$
  $male16X(k) \in F$
  $male16YX(k) \in F$
- We are interested in solutions that are ground instances of the formula '$\mathit{priorodds}(k) = p$ is x-implied by Δ', where x can be 'credulously', 'skeptically' or 'not', and Δ is the default theory specified by $I^g(P_4) \cup I^d(P_4)$.

Note that the default theory of P_4 has two extensions, one containing $\mathit{priorodds}(k) = \frac{1}{9999}$ and the other containing $\mathit{priorodds}(k) = \frac{1}{49999}$. So our informal problem statement has two solutions, viz.

s_1 = '$\mathit{priorodds}(k) = \frac{1}{9999}$ is credulously implied by Δ'; and
s_2 ='$\mathit{priorodds}(k) = \frac{1}{49999}$ is credulously implied by Δ'.

Now the first solution yields (approximately) $\Pr(k|d) = 0.995$ while the second results in (approximately) $\Pr(k|d) = 0.976$, so since the proof standard for k is 0.99, the outcome of the debate on the prior odds of k is crucial to the question whether k can be proven beyond reasonable doubt.

As explained above, subproblems are related by output-input transformers. They are formally defined as follows.

Definition 3. *An* O/I transformer *from problem solving methods $M_1, \ldots, M_m$ to a problem solving method M_n is a function from $\mathcal{L}^O_{M1} \times \ldots \times \mathcal{L}^O_{Mm}$ into $\mathcal{L}^I_{Mn}$.*

In other words, an O/I transformer takes a set of formulas stated in the output languages of a set of problem solving methods, and converts it into a formula in the input language of a single problem solving method. Note that O/I transformers need not be translation functions between languages: firstly, an O/I transformer can take output formulated in a *set* of languages as input, and secondly, O/I transformers may express problem solving knowledge that goes beyond semantic knowledge, as the following transformers of our running example show.

We first define the following O/I transformer schemes from *MBayes* and *MFOL* to *MPropDL*.

> T_1: If $\Pr(H|E) \geq s$ is implied by Π in *MBayes* and 'The proof standard for H is s' is implied by T in *MFOL* then $H \in F$ in *MPropDL*.
>
> T_2: If $(1 - s) \leq \Pr(H|E) < s$ is implied by Π in *MBayes* and 'The proof standard for H is s' is implied by T in *MFOL* then $\top{:}\top/H \in D$ in *MPropDL*.

In other words, if a proof standard for H has been met, it is put into the facts F of a default theory, while if it has been met for neither H or $\neg H$ then both are put into the defaults D of the default theory (strictly speaking the defaults $\top{:}\top/H$ and $\top{:}\top/\neg H$ are added to D). With the laws of probability this implies for any h that $\top : h/h \in D$ iff $\top : \neg h/\neg h \in D$, that no proposition and its negation are in F and that for any proposition to which a transformer can be applied, either the proposition or its negation is in F, or one of the corresponding defaults is in D. Note that T_2 is an example of an O/I transformer that is not a translation function between languages: it takes input from a *set* of problems; and it expresses substantive legal instead of just semantic knowledge.

Which O/I transformers are appropriate from P_4 to P_2? Here there are several options. It is clear that any prior odds on k that is skeptically implied by a default theory will be part of Π in P_2. However, what if it is only credulously implied, as in the above default theory of P_4? Then there are two options. One is to 'block the ambiguity' and to input neither odds statement into P_2. In that case it is easy to see that no posterior probability on k can be calculated by P_2 so that P_1 receives no input on k and m is not derivable in P_1 in any sense. However, another option is to 'propagate the ambiguity', that is, to input two alternative prior odds of k into *MBayes* and to see whether their difference makes a difference in P_1. Here I do not want to argue for one option as the best and I simply choose the latter option to illustrate how the abstract formalism deals with conflicting problems. We thus have the following transformer scheme:

T_3: If $priorodds(k) = p$ is skeptically or credulously implied by Δ in *MFolDL* then $priorodds(k) = p \in Pr$ in *MBayes*.

Problem treatments are assumed to be based on a so-called general problem specification, which is a set of pairs where each pair is a problem solving method plus a set of formulas in its input language.

Definition 4. *A* general problem specification *(GPS) is a set G of pairs (M, I^g) where*

1. *M is a problem solving method;*
2. *$I^g \subseteq \mathcal{L}_M^I$*
3. *If $(M, I^g) \in G$ and $(M', I^g) \in G$ then $M = M'$.*

Further constraints on GPS could be defined but their study must be left for future research.

In our running example we have in fact specified the following GPS:

$$\{(MPropDL, I^g(P_1)), (MBayes, I^g(P_2)), (MFOL, I^g(P_3)), (MFolDL, I^g(P_4))\}$$

It can now be defined how problem treatments can be combined.

Definition 5. *A* combination of problem treatments *based on a GPS G is a finite sequence $A = P_1, \ldots, P_m$ of problem treatments satisfying the following conditions for every P_i in A.*

1. *There exists a pair $(M, I^g) \in G$ such that $M(P_i) = M$ and $I^g(P_i) = I^g$; and*
2. *for each $\varphi_i \in I^d(P_i)$ there exist one or more problem treatments $P_j, \ldots, P_k$ $(j, k < i)$ in A with solutions $\varphi_j, \ldots, \varphi_k$ and a transformer T from $M(P_j) \times \ldots M(P_k)$ to $M(P_i)$ such that $T(\varphi_j, \ldots, \varphi_k) = \varphi_i$. (In such cases we say that P_i* depends *in A on $P_j, \ldots, P_k$.)*

For any combination of problem treatments $\mathcal{A}$ the set $\mathcal{A}^* = \{P \mid$ there exists an $A \in \mathcal{A}$ that contains $P\}$.

So the input of a problem treatment must be treated with the method specified by the GPS, and its 'derived' input must be provided via O/I transformers by, possibly combined, solutions of other problem treatments. Note that together Definitions 4 and 5 formalise that, for all problem treatments, I^g is given while I_d is obtained from other treatments.

On the basis of our example GPS the following combined problem treatments can be constructed for our solutions of interest.

$A_1 =$
$P_4 = \langle MFolDL, I^g(P_4), \emptyset, \{s_1, s_2\}\rangle$
$P_3 = \langle MFOL, I^g(P_3), \emptyset, \{s_3\}\rangle$
$P_{2a} = \langle MBayes, I^g(P_{2a}), \{s_1\}, \{s_4\}\rangle$ (T_3 applied to P_4)
$P_{1a} = \langle MPropDL, I^g(P_{1a}), \{s_5\}, \{s_6\}\rangle$ (T_1 applied to P_3 and P_{2a})
Where

$s_1 = \mathit{priorodds}(k) = \frac{1}{9999}$ is credulously implied by Δ
$s_2 = \mathit{priorodds}(k) = \frac{1}{49999}$ is credulously implied by Δ
$s_3 = \mathit{Standard}(k) = 0.99$ is implied by T
$s_4 = \Pr(k|E) = 0.995$ is implied by Π
$s_5 = k \in F$
$s_6 = m$ is skeptically implied by Δ

$A_2 =$
$P_4 = \langle \mathit{MFolDL}, I^g(P_4), \emptyset, \{s_1, s_2\}\rangle$
$P_3 = \langle \mathit{MFOL}, I^g(P_3), \emptyset, \{s_3\}\rangle$
$P_{2b} = \langle \mathit{MBayes}, I^g(P_{2b}), \{s_2\}, \{s_7\}\rangle$ (T_3 applied to P_4)
$P_{1b} = \langle \mathit{MPropDL}, I^g(P_{1b}), \{s_8\}, \{s_9, s_{10}\}\rangle$ (T_2 applied to P_3 and P_{2a})

Where
$s_7 = \Pr(k|E) = 0.976$ is implied by Π
$s_8 = k \in D$
$s_9 = m$ is credulously implied by Δ
$s_{10} = m$ is not skeptically implied by Δ

We now come to an important element of the formalism. To deal with situations in which different problem treatments provide mutually contradictory input for a certain problem, a binary relation of *conflict* between problem treatments is introduced. This relation captures situations in which the same 'informal' problem can be formalised in different ways. For instance, if an informal problem is to be modelled as an application of probability theory but there are two conflicting inputs on what are the correct probabilities for a certain set of probabilistic variables, then there should be two alternative probability distributions. To ensure this, the conflict relation is a necessary ingredient.

Definition 6. *A problem treatment P* conflicts *with a problem treatment P' if $I^g(P) \cup I^d(P) \cup I^g(P') \cup I^d(P') \vdash \bot$. A set S of problem treatments is called* conflict-free *if no member of S conflicts with a member of S.*

Note that the conflict relation is not only symmetric but also irreflexive since a problem treatment's input is assumed consistent.

In our example P_{2a} and P_{2b} conflict with each other since $I^d(P_{2a})$ contains '$\mathit{priorodds}(k) = \frac{1}{9999} \in \Pi$' while $I^d(P_{2b})$ contains '$\mathit{priorodds}(k) = \frac{1}{49999} \in \Pi$'. (Recall that $I^g(P_{2a})$ and $I^g(P_{2b})$ are assumed to contain axioms that make these two statements contradictory.) Likewise P_{1a} and P_{1b} are conflicting since $I^d(P_{1a})$ contains '$k \in F$' while $I^d(P_{1b})$ contains '$k \in D$'. The two combined problem treatments and these conflict relations are displayed in Figure 3 below (O/I transformers are displayed as solid lines and conflict relations as dashed lines.)

How can 'overall' solutions to general problem specifications be defined? Here we can exploit the fact that at a certain level of abstraction the structure of the present formalism is isomorphic to that of [12]. Firstly, in the present framework combined problem treatments are sequences of elementary problem treatments connected by applications of O/I transformers, so they can be represented as directed acyclic graphs. Likewise, in Pollock's framework arguments are sequences

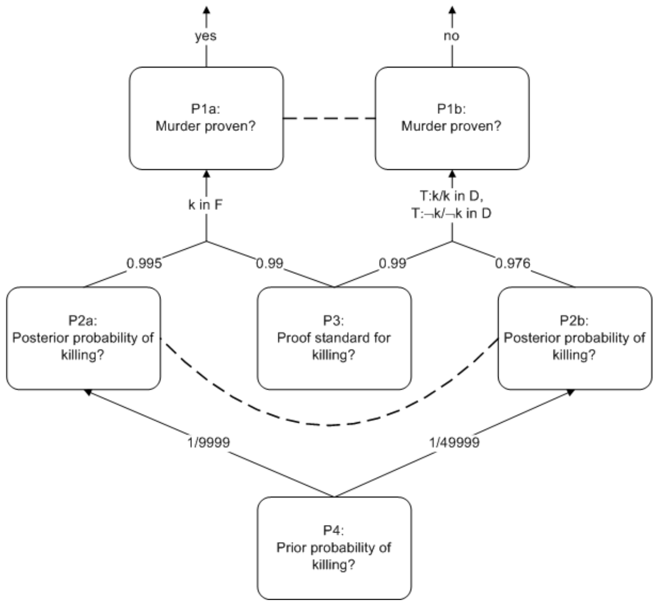

Fig. 1. The of the murder case

of elementary 'lines of argument' (statements plus a set of assumptions and a strength) connected by applications of inference rules, so Pollock's arguments can also be represented as directed acyclic graphs. Secondly, in the present framework conflict relations can hold between the elements of a combined problem treatment; likewise, in Pollock's framework 'defeat' relations can hold between lines of arguments. Now Pollock defines so-called defeat status assignments as labellings of argument lines, and he does so in terms of just a set of arguments represented as DAGs plus the defeat relation between lines of these arguments: he does not use the internal structure of argument lines nor the nature of the defeat relations for these purposes. Therefore, because of the just-explained isomorphism between the two frameworks, the same labellings can be defined for the present framework.

Note that despite this isomorphism, there are at a more concrete level two main differences between the present formalism and Pollock's. Firstly, while a problem treatment contains two sets of formulas related by an application of a problem solving method (which can be quite a complex process), a line of argument is just a single proposition plus a set of assumptions and a strength: within a line of argument no process takes place at all. Secondly, while problem treatments are connected via O/I transformers, lines of argument are connected by inference rules, which are quite different in nature from O/I transformers.

Let us now formalise these observations, adapting the notation and terminology of [12] to the present formalism.

Definition 7. *[labellings.] Let $\mathcal{A}$ be a combination of problem treatments. A* labelling *of $\mathcal{A}$ is a labelling of the elements of $\mathcal{A}^*$ defined as follows. For every $P \in \mathcal{A}^*$ belonging to the combination of problem treatments A:*

1. *P is labelled 'in' if:*
   (a) *all problem treatments on which P depends in A are labelled 'in'; and*
   (b) *all problems in $\mathcal{A}^*$ that conflict with P are labelled 'out';*
2. *P is labelled 'out' if:*
   (a) *some problem treatment on which P depends in A is labelled 'out'; or*
   (b) *P conflicts with a problem in $\mathcal{A}^*$ that is labelled 'in'.*

A labelling L of $\mathcal{A}$ is maximal if for all P and all labellings L' of $\mathcal{A}$ it holds that if P is in (out) in L' then P is in (out) in L. A labelling of $\mathcal{A}$ is complete if all nodes in $\mathcal{A}^*$ are labelled.

Intuitively, that a problem treatment is 'in' means that it can be regarded as a possible (but maybe not the only) way to solve a problem given the other problem treatments to which it is related by transformer and conflict relations.

In our running example the set $\{A_1, A_2\}$ has two complete labellings, viz L_1, in which P_{2b} and P_{1b} are out and the other problems are in, and L_2, in which P_{2a} and P_{1a} are out and the other problems are in.

It is now possible to define overall solutions to a GPS, analogously to the skeptical and credulous consequence notions of [12].

Definition 8. *Relative to a combination of problem treatments $\mathcal{A}$ a problem treatment $P \in \mathcal{A}^*$ is* defensible *if some maximal labelling of $\mathcal{A}$ labels P 'in', and P is* justified *if all maximal labellings of $\mathcal{A}$ label P 'in'. A formula φ is a* defensible solution *if φ is a solution of a defensible problem treatment, and φ is a* justified solution *if all labellings of $\mathcal{A}$ make some problem treatment 'in' that has φ as a solution.*

In our running example P_3 and P_4 are justified problem treatments while the other treatments are defensible. In consequence, there are two defensible solutions on the issue of murder. The first is that murder is skeptically derivable (based on L_1) while the second is that murder is not skeptically but only credulously derivable (based on L_2). So in the end there is no justification for convicting the suspect for murder.

The isomorphism with the formalism of [12] makes that the following properties can be proven.

Proposition 1. *Let $\mathcal{A}$ be an arbitrary combination of problem treatments.*

1. *All labellings of $\mathcal{A}$ are conflict-free.*
2. *$\mathcal{A}$ has at least one maximal labelling.*
3. *All maximal labellings of $\mathcal{A}$ are complete.*
4. *If $\mathcal{A}^*$ is conflict-free, then $\mathcal{A}$ has a unique labelling which is complete and which makes all problems of $\mathcal{A}^*$ 'in'.*

Proof. First, since Pollock's defeat status assignments are by Theorem 6.15 of [15] equivalent to an argumentation framework in the sense of [10] with preferred semantics, the results of [10] on preferred semantics also hold for labellings of GPS. Then (1) and (2) follow from Theorem 1 of [10]. Furthermore, (3) follows from Theorem 33 of [10] and symmetry and irreflexivity of the conflict relation between problems, so that all cycles through $\mathcal{C}$ relations are of even length. From this, property (4) also follows.

To conclude this section, it should be noted that Definitions 7 and 8 are not the only possible ones. Since [15] have shown that Pollock's system is an instance of Dung's abstract argumentation frameworks, any other semantics of [10] could also be used. The above definitions have been chosen since they allow for 'floating solutions' to a problem statement, analogous to floating conclusions in nonmonotonic logic. In the present context this seems useful: if a problem has alternative conflicting modellings but they solve some subproblem in the same way then the above definitions say that that solution is justified.

4 Discussion and conclusion

This paper has presented an abstract formalism for combining different modes of reasoning. The model is abstract enough to include a wide variety of reasoning methods, ranging from purely symbolic to purely numeric ones. The aims to capture dependencies between problems and to manage alternative solutions of problems have been realised by exploiting a formal relation between our problem specifications and [12]'s system. Thus established theory on argumentation systems can be applied to a new phenomenon. A practical benefit of the present approach is the possibility to reason in one formalism about modelling decisions in another formalism, rather than to leave this to an unspecified knowledge engineering phase. Although several examples in this paper involved the combination of a nonmonotonic logic with probability theory, the formalism is by no means restricted to such combinations but can combine any set of reasoning methods. Moreover, in such combinations the individual methods can be left as they are, which is another practical benefit of the present approach.

Although the formalism uses techniques from the formal study of argumentation in AI, it does not require that problems are formalised in terms of inference rules, arguments and counterarguments as in [12–14]. Instead it allows the use of any reasoning method, which is achieved by abstracting from their internal structure. A key idea of this paper has been that O/I transformers do not operate on object-level descriptions of problems, as inference rules do, but on metalevel descriptions: thus existing formalisations (or even implementations) of problem solving methods can simply be 'plugged into' the present formalism, without the need to translate them into a new format. These ideas cannot be formalised with purely abstract support and conflict relations as in [11].

Finally, much work remains to be done. Firstly, it would be interesting to investigate whether conflicts between problem treatments can be resolved in other problem treatments (cf. [16] for an extension of [10]'s framework with attacks on attacks.) Also, a systematic study of the nature of O/I transformers between different modes of reasoning must be carried out, profiting whenever possible from existing work on integrating KR & R systems.

References

1. Lempert, R.: The new evidence scholarship: Analyzing the process of proof. Boston University Law Review **66** (1986) 439–477
2. Poole, D.: Logical argumentation, abduction and Bayesian decision theory: a Bayesian approach to logical arguments and its application to legal evidential reasoning. Cardozo Law Review **22** (2001) 1733–1745
3. Thagard, P.: Causal inference in legal decision making: Explanatory coherence vs. Bayesian networks. Applied Artificial Intelligence **18** (2004) 231–249
4. Bex, F., Prakken, H., Reed, C., Walton, D.: Towards a formal account of reasoning about evidence: argumentation schemes and generalisations. Artificial Intelligence and Law **12** (2003) 125–165
5. Ashley, K.: Modeling Legal Argument: Reasoning with Cases and Hypotheticals. MIT Press, Cambridge, MA (1990)
6. Prakken, H., Sartor, G.: Modelling reasoning with precedents in a formal dialogue game. Artificial Intelligence and Law **6** (1998) 231–287
7. Bench-Capon, T.: Neural networks and open texture. In: Proceedings of the Fourth International Conference on Artificial Intelligence and Law, New York, ACM Press (1993) 292–297
8. Prakken, H., Sartor, G.: A dialectical model of assessing conflicting arguments in legal reasoning. Artificial Intelligence and Law **4** (1996) 331–368
9. Gabbay, D.: Fibring Logics. Number 38 in Oxford Logic Guides. Oxford University Press, Oxford (1999)
10. Dung, P.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n–person games. Artificial Intelligence **77** (1995) 321–357
11. Cayrol, C., Lagasquie-Schiex, M.C.: Graduality in argumentation. Journal Artificial Intelligence Research **23** (2005) 245–297
12. Pollock, J.: Cognitive Carpentry. A Blueprint for How to Build a Person. MIT Press, Cambridge, MA (1995)
13. Vreeswijk, G.: Abstract argumentation systems. Artificial Intelligence **90** (1997) 225–279
14. Gordon, T., Prakken, H., Walton, D.: The Carneades model of argument and burden of proof. Artificial Intelligence **171** (2007) 875–896
15. Jakobovits, H., Vermeir, D.: Robust semantics for argumentation frameworks. Journal of Logic and Computation **9** (1999) 215–261
16. Modgil, S.: An abstract theory of argumentation that accommodates defeasible reasoning about preferences. In: Proceedings of the 9th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty. Number 4724 in Springer Lecture Notes in CS (2007) 648–659