Subjective Logic and Arguing with Evidence

Nir Oren [a,*], Timothy J. Norman [a], Alun Preece [a]

[a]*Department of Computing Science, University of Aberdeen, AB24 3UE, Scotland*

Abstract

This paper introduces a Subjective Logic based argumentation framework primarily targeted at evidential reasoning. The framework explicitly caters for argument schemes, accrual of arguments, and burden of proof; these concepts appear in many types of argument, and are particularly useful in dialogues revolving around evidential reasoning. The concept of a sensor is also useful in this domain, representing a source of evidence, and is incorporated in our framework. We show how the framework copes with a number of problems that existing frameworks have difficulty dealing with, and how it can be situated within a simple dialogue game. Finally, we examine reasoning machinery to enable an agent to decide what argument to advance with the goal of maximising its utility at the end of a dialogue.

Key words: Argumentation, Dialogue Game, Heuristic, Evidence

1 Introduction

It has long been recognised that argumentation research can be divided into two main strands [16]. The first involves the analysis of argument, while the second borrows ideas from argumentation theory in an attempt to create powerful reasoning mechanisms. In this paper, we follow the latter strand, using argument to create a powerful framework for evidential and diagnostic reasoning. Informally, we are trying to address situations where different agents, each with their own goals and viewpoints, are attempting to reach a shared agreement about the state of a subset of their environment. We further assume that the environment is partially observable, and that any information about

---

* Corresponding Author.
   Email addresses: `noren@csd.abdn.ac.uk` (Nir Oren),
`tnorman@csd.abdn.ac.uk` (Timothy J. Norman), `apreece@csd.abdn.ac.uk`
(Alun Preece).

it is obtained through the use of fallible sensors. Finally, we assume that the agents are self interested, and that different agents may have opposing goals.

Without a trusted third party, a centralised solution to this problem is difficult. Our proposed approach involves the agents engaging in dialogue with each other, exchanging arguments, and probing sensors for additional information about the environment. By combining the information from sensors and arguments, a shared world view can be constructed. To tackle the problem, therefore the following is needed:

- A representation mechanism for the environment, agents' knowledge, arguments and any other components with which interaction is required.
- A technique for determining which conclusions are justified when opposing arguments interact.
- A specification detailing how agents should engage in dialogue with each other.
- A way for the agents to decide which arguments to advance and what sensors to probe.

Prakken [15] identified these as the logical, dialectic, procedural and heuristic layers of an argument framework. Our logical layer is built around Subjective Logic [6], allowing us to represent concepts such as likelihood and uncertainty in a concise and elegant manner. The way in which arguments are constructed in our framework and used at the dialectic level is intended to support a rich representation of arguments; we are able to represent concepts such as accrual of arguments, argument schemes and argument reinforcement in a natural manner. While the logical and dialectic layers are domain independent, acting as a general argument framework, the explicit introduction of sensors at the procedural level allows us to attack our problem.

A sensor refers to anything that can determine the state of a subset of the environment. Multiple sensors may exist for certain parts of the environment, and some of these sensors may be more accurate than others. Finally, sensors may not perform their services for free. Thus, sensors capture an abstract notion of a source of evidence within our framework.

At the procedural level, agents engaging in dialogue, taking turns to advance arguments and probe sensors in an attempt to achieve their goals. In this context, an agent's goal involves showing that a certain environment state holds. We assume that an agent associates a utility with various goal states. Our heuristic layer guides an agent and tells it what arguments to advance, and which sensors to probe during its turn in the dialogue game.

Using argumentation for evidential reasoning has a number of advantages over other approaches, including:

- *Understandability.* It is much easier to follow the reasoning behind a dialogue than to attempt to interpret a complicated formula.
- *Resource bounded reasoning.* It is possible to plug in different agents with different capabilities (and hence different computational costs) and still obtain some (possibly non-accurate) answers.
- *Anytime.* Related to the previous point, it is possible to terminate the dialogue at any time, with the provision that inaccurate, or incorrect answers may be obtained.
- *Ease of knowledge engineering.* At any point in the argument process, it is easy to introduce additional facts and see how they alter the dialogue.

In this section, we provided a brief overview of the problem we are trying to tackle, and outlined our proposed solution. Next, we discuss Subjective Logic, as it forms a core part of our formalism. Once this is done, we proceed to describe our formalism, following which an illustrative example is provided. We then examine the strengths and weaknesses of our approach in more detail, and compare it with existing techniques. We also examine possible areas for future work before concluding the paper.

2 Subjective Logic

Subjective Logic [6] provides a standard set of logical operators (such as negation, conjunction and disjunction), intended for use in domains containing uncertainty, and, more specifically, domains in which *opinions* regarding the truth or falsehood of a (set of) domain elements differ. Subjective logic also contains a number of other operators, designed to combine opinions in an intuitively correct manner. The semantics of our formalism, presented in Section 3, are based on Subjective Logic (hereafter abbreviated SL), and we therefore now provide a brief overview of the area. Most of this description is taken directly from Jøsang's original paper [6].

Since SL is based on Dempster-Shafer evidence theory, it operates on a *frame of discernment*, denoted by Θ. A frame of discernment contains the set of possible system states, only one of which represents the actual system state. These are referred to as atomic system states.

In many situations, it is difficult to determine what state one is in, and it thus makes sense to talk about non-atomic states, consisting of the union of a number of primitive states. If the system is in primitive atomic state x_i, it is also in all states x_j such that $x_i \subseteq x_j$. The powerset of Θ, denoted by 2^Θ, consists of all possible unions of primitive states.

An observer assigns a belief mass to various states based on its strength of

belief that the state (or one of its substates) is true:

Definition 1 *(Belief Mass Assignment) Given a frame of discernment Θ, one can associate a belief mass assignment[1] $m_\Theta(x)$ with each substate $x \in 2^\Theta$ such that*

(1) $m_\Theta(x) \geq 0$
(2) $m_\Theta(\emptyset) = 0$
(3) $\sum_{x \in 2^\Theta} m_\Theta(x) = 1$

For a substate x, $m_\Theta(x)$ is its *belief mass.*

Belief mass is an unwieldy concept to work with. When one speaks of belief regarding a certain state, one refers not only to the belief mass in the state, but also to the belief masses of the state's substates. Similarly, when one speaks about disbelief, that is, the total belief that a state is not true, one needs to take substates into account. Finally, subjective logic introduces the concept of uncertainty, that is, the amount of belief that one might be in a superstate or a partially overlapping state. We can define these concepts formally as:

Definition 2 *(Belief, Disbelief and Uncertainty) Given a frame of discernment Θ and a belief mass assignment m_Θ on Θ, we can define the belief function for a state x as*

$$b(x) = \sum_{y \subseteq x} m_\Theta(y) \ where \ x, y \in 2^\Theta$$

The disbelief function as

$$d(x) = \sum_{y \cap x = \emptyset} m_\Theta(y) \ where \ x, y \in 2^\Theta$$

And the uncertainty function as

$$u(x) = \sum_{\substack{y \cap x \neq \emptyset \\ y \not\subseteq x}} m_\Theta(y) \ where \ x, y \in 2^\Theta$$

These functions have a number of features that should be noted. First, they all range between zero and one. Second, they always sum to one, meaning that it is possible to deduce the value of one function given the other two. If the entire belief mass is assigned to Θ, then $u(x) = 1$ if $x \neq \Theta$. This situation is analogous to total uncertainty. Dogmatic beliefs occur when no belief mass is assigned to Θ.

---

[1] A belief mass assignment is often also referred to as a basic belief assignment, or bba, within belief theory. Since our framework is Subjective Logic based, we will refer to belief mass assignment within this paper.

Another concept introduced in subjective logic is relative atomicity. The relative atomicity of two states x and y is the ratio of the number of states shared between them and the number of states in y. Relative atomicity is needed to compute the expected probability of an outcome, and captures the idea of prior probabilities. However, by taking into account the provision that we cannot translate directly from opinions to probabilities, we ignore atomicity in the interests of simplifying our framework.

A focused frame of discernment for a state x is a frame of discernment containing only the states x and $\bar{x}$, the complement of x. Jøsang provides a transformation from a frame of discernment to a focused frame of discernment. However, for a state x, the only value that changes between the two frames is its atomicity.

An opinion consists of the belief, disbelief, uncertainty (and relative atomicity) as computed over a binary frame of discernment:

Definition 3 (Opinion) *Given a binary frame of discernment Θ containing x and its complement $\bar{x}$, and assuming a belief mass assignment m_Θ with belief, disbelief, uncertainty and relative atomicity functions on x in Θ of $b(x), d(x), u(x)$ and $a(x)$, we define an opinion over x, written ω_x as*

$$w_x \equiv \langle b(x), d(x), u(x), a(x) \rangle$$

Since we ignore atomicity, we write an opinion for a state x as the triple $\langle b(x), d(x), u(x) \rangle$. When the context is clear, we may refer to (for example) the belief component of an opinion ω_x as $b(\omega_x)$. For compactness, we may occasionally write b_x, d_x, u_x instead of $b(x), d(x)$ and $u(x)$.

Jøsang has defined a large number of operators that are used to combine opinions, some of which are familiar such as conjunction and disjunction, and some less so such as abduction. We look at three operators, namely negation, discounting, and consensus.

The propositional negation operator calculates the opinion that a proposition does not hold, and is defined as follows:

Definition 4 (Propositional Negation) *For a $\omega_x = \langle b_x, d_x, u_x \rangle$, the propositional negation is computed as $\omega_{\neg x} = \langle d_x, b_x, u_x \rangle$.*

Given an agent α, we represent its opinion on a proposition x as ω_x^α. Discounting is a model of hearsay. That is, given that an agent α holds an opinion ω_β^α about agent β's reliability, and given that β has an opinion ω_x^β about proposition x, $\omega_x^{\alpha\beta}$ gives the opinion α has about x.

Definition 5 (Discounting) *Given two opinions $\omega_\beta^\alpha = \langle b_\beta^\alpha, d_\beta^\alpha, u_\beta^\alpha \rangle$, and*

$\omega_x^\beta = \langle b_x^\beta, d_x^\beta, u_x^\beta \rangle$, *the discounted opinion is* $\omega_x^{\alpha\beta} = \langle b_\beta^\alpha b_x^\beta, b_\beta^\alpha d_x^\beta, d_\beta^\alpha + u_\beta^\alpha + b_\beta^\alpha u_x^\beta \rangle$

The independent consensus operator represents the opinion an imaginary agent would have about x if it had to assign equal weighting to the opinions ω_x^α and ω_x^β. Later work [7] suggests how one can handle situations where $\kappa = 0$, but, by assuming that sensors reflect reality, we can ensure that κ is never 0 in our framework.

Definition 6 (*Independent Consensus*) *Given two independent opinions* ω_x^α *and* ω_x^β *about the same proposition* x, *the independent consensus opinion is defined as* $\omega_x^{\alpha,\beta} = \langle (b_x^\alpha u_x^\beta + b_x^\beta u_x^\alpha)/\kappa, (d_x^\alpha u_x^\beta + d_x^\beta u_x^\alpha)/\kappa, u_x^\alpha u_x^\beta/\kappa \rangle$ *Where* $\kappa = u_x^\alpha + u_x^\beta - u_x^\alpha u_x^\beta$ *such that* $\kappa \neq 0$.

To simplify notation, we may represent the operators as follows:

$$\omega_{\neg x} \equiv \neg \omega_x$$
$$\omega_x^{\alpha\beta} \equiv \omega_\beta^\alpha \otimes \omega_x^\beta$$
$$\omega_x^{\alpha,\beta} \equiv \omega_x^\alpha \oplus \omega_x^\beta$$

With this grounding, we are now in a position to describe our framework.

3   The Framework

Following Prakken's model[15], we build our framework in layers, starting at the logical layer, where we describe how an argument is constructed. In the dialectic layer, we look at how arguments interact, and then show how agents may engage in dialogue in the procedural layer. Finally, in the heuristic layer, we show how agents can decide which lines of argument should be advanced in a dialogue.

One concept that cuts across a number of layers is that of an *argument scheme* [20]. Argument schemes are common, stereotypical patterns of reasoning which are, typically, non-deductive and nonmonotonic. In our framework, arguments are instantiated instances of argument schemes. Thus, our universe of discourse is a tuple $U = (PF, AS)$ where PF contains (a finite number of) possible facts about our universe, and AS is the set of argument schemes. We also assume that we have two distinct sets of symbols Σ and Φ.

Facts are represented as grounded predicates, and have an associated opinion. A set of predicates with an identical name acts as a frame of discernment. Since only a single state within a frame of discernment can be true, we cannot simply match individual predicates to states. Instead, we define the atomic

states by computing the powerset of the individual predicates, and associate a non-atomic state with the predicate that encapsulates all atomic states in which the predicate holds. The left hand side of Figure 1 provides an example of this.

Definition 7 *(**Predicate**) Given a universe of discourse $U = (PF, AS)$, a predicate is a tuple $(Name, Parameters) \in PF$ where $Name \in \Sigma$. Parameters is itself a tuple of finite arity whose members are elements of PF. Given two predicates $(N_1, Parameters_1)$, and $(N_2, Parameters_2)$, $|Parameters_1| = |Parameters_2|$ if $N_1 = N_2$.*

We refer to the frame of discernment 2^P, where P is the set $\{Parameters | (Name, Parameters) \in PF \text{ and } Name = N\}$ as Θ_N. This frame of discernment has additional states $p_i \in P$ containing all states $s \in 2^P$ such that $p_i \subseteq s$.

The atomic states of a frame of discernment Θ_N are referred to as $atomic(\Theta_N)$. We differentiate between atomic and non-atomic states by writing the latter in bold.

A predicate is thus embedded within a frame of discernment, which in itself is another predicate. We assume the existence of an anonymous, top level frame of discernment in which predicates reside. While it is possible to remove this nesting, and store all predicates within a single universal frame of discernment, embedding them in this way helps reduce the exponential explosion in the number of atomic states.

We place one restriction on the nesting of frames of discernment, and that is that the graph of nestings must be acyclic, that is, a predicate may not nest other predicates such that it eventually nests itself.

As an example, consider the symbols $a, b, fred, holds, geologist, expert$ and $geology$. We may have the following predicates:

$(holds, \{a\}), (holds, \{b\}), (holds, \{geologist\})$
$(geologist, \{fred\}), (expert, \{(geologist, geology)\})$

For convenience, we rewrite a predicate of the form (A, B) as $A(B)$. Thus, some of our possible facts include $holds(a), expert(geologist, geology)$, as well as $holds(geologist)$ and $holds(geologist(fred))$. Note that $atomic(\Theta_{holds})$ is $\{a, b, geologist\}$.

Let us examine a subset of the *holds* frame of discernment, containing the predicates *geologist* (which in turn contains the predicate *fred*) and a. Figure 1
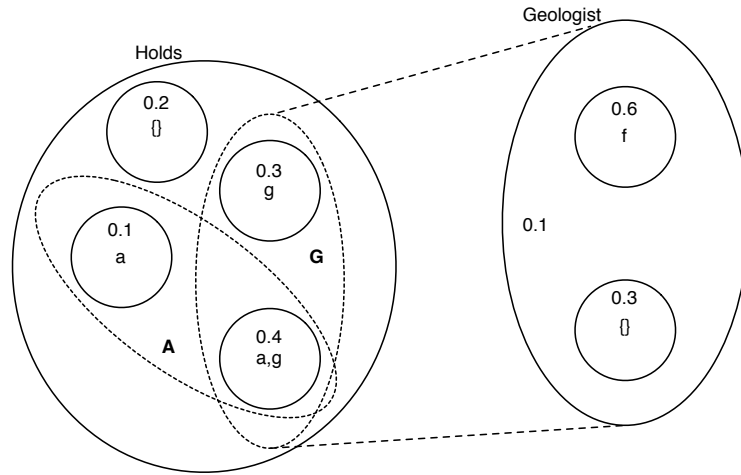
Fig. 1. The frames of discernment and associated belief mass assignments for the predicates $(holds, \{a, geologist\}), (geologist, \{fred\})$. g stands for $geologist$, and f for $fred$. Capitalised letters show non-atomic states.
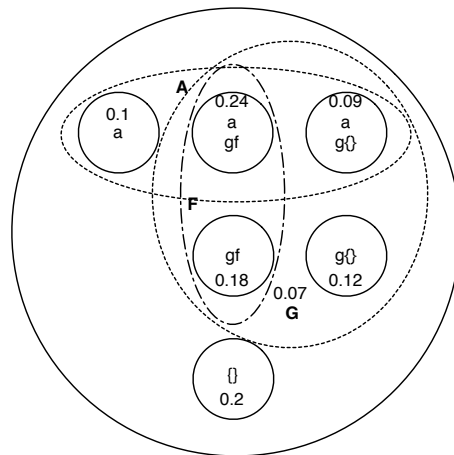

Fig. 2. The merged frame of discernment for the predicates $(holds, \{a, geologist\}), (geologist, \{fred\})$. gf stands for the predicate $geologist(fred)$, and g is the primitive state obtained from the empty set of the original (unmerged) $geologist$ frame of discernment. The dotted ellipses, together with the associated capitalised letters show non-atomic states.

shows these frames of discernment, together with a belief mass assignment to the various states.

One question that we must answer is how nested belief masses (and from them opinions) are computed. For example, what would the belief mass of $holds(geologist(fred))$ be? To answer this question, we need to merge our frames of discernment. Figure 2 shows the merged frame of discernment.

To create a merged frame of discernment from the frame of discernment of two predicates $\Theta_{pr_1}, \Theta_{pr_2}$ where $pr_1 = (n_1, \{p_{11}, \ldots, p_{1m}\})$, $pr_2 = p_{1i}$ for some $1 < i < m$, and $pr_2 = (n_2, \{p_{21}, \ldots, p_{2n}\})$, we first define pa, the reduced set

of parameters as $\{p_{11}, \ldots, p_{1m}\} \backslash p_{2n}$.

Then the atomic states in the merged frame of discernment are

$$\Theta_{pa} \cup ((\Theta_{pa} \backslash \{\}) \times \Theta_{pr_2}) \cup \Theta_{pr_2}$$

Note that we must differentiate between the empty sets found in both predicates when in the merged state, and will thus refer to the former as $\{\}_1$ and the latter as $\{\}_2$.

The new frame of discernment contains $n + m$ non-atomic states. As before, these non-atomic states encapsulate all atomic states containing a parameter of the same name. The only exception to this is the state p_{1i}, which encompasses any atomic state with elements in p_{2j} for $j = 1 \ldots n$.

The belief mass assignments (BMAs) for all atomic states from pr_1 remain the same. BMAs for atomic states containing elements from pr_2 are computed by multiplying the BMA from the relevant state in pr_1 with the state from pr_2. Thus for example, the BMA for state a, f in Figure 2 is $0.4 \times 0.6 = 0.24$. As can be seen from the figure, compound state **G** was also assigned a BMA. This was because the frame of discernment in the original predicate had an associated BMA. The BMA for the p_{1i} compound state is computed as the BMA assigned to the original frame of discernment multiplied by the BMA assigned to all its substates in pr_1.

Converting BMAs to opinions, we see that an opinion about a proposition remains the same in both the merged and unmerged models. This is an important result for the rest of our framework, as arguments are based on opinions rather than BMAs. It should be noted that $\omega(geologist(fred)) \neq \omega(holds(geologist(fred)))$, even though the belief mass assignments remain the same. However, $\omega(a)$ remains the same in both cases.

Argument schemes, while mainly used in the higher levels of our framework, form the second part of our universe of discourse. They are instantiated to form concrete arguments. We assume that our framework contains only a finite number of argument schemes:

Definition 8 (Argument Scheme) *Given a universe of discourse $U = (PF, AS)$, an argument scheme $as \in AS$ is a tuple*

$$(Name, Premises, Conclusions, F, A)$$

Name uniquely identifies the argument scheme, Premises and Conclusions are tuples of the form $(l_i, (s_1, \ldots, s_n))$ such that $\exists (l_i, Parameters) \in PF$ where $n = |Parameters|$. $s_i \in \Phi$ are symbols. All tuples within Premises and Conclusions must differ from each other. $A : 2^{Premises} \rightarrow \{true, false\}$ is the applicability

function. F is a user-defined function mapping opinions over all premises to opinions over all conclusions.

The admissibility function A is used to compute whether an argument (defined below) can be used given a specific set of premises. If it can, the F function is used to assign a set of opinions to the conclusions of the argument based on the opinions assigned to its premises. Both the A and F functions can make use of the following:

- The standard arithmetic operators $(+, -,$ etc).
- Comparison operators $(<, \geq,$ etc).
- Boolean operators $(not, or,$ etc).
- The functions $b(\omega)$, $d(\omega)$, $u(\omega)$ on an opinion ω.
- Subjective logic boolean (Definition 4), independent consensus and discounting (Definitions 5 and 6), and other [7] operators, operating on opinions.
- References to predicates mentioned in *Premises* and *Conclusions*.
- An if/then conditional.
- An "unpacking" operator to refer to the parameters of a predicate.

As an example, Modus Ponens can be represented with the argument scheme $(ModusPonens, \{holds(A), implies(A, B)\}, \{holds(B))\}, F, true)$. Here, F is:

$$\omega(holds(B)) = \begin{cases} \langle 0, 0, 1 \rangle & b(holds(A)) < 0.5 \text{ or} \\ & b(implies(A, B)) < 0.5 \\ \omega(holds(A)) & b(holds(A) < b(implies(A, B)) \\ \omega(implies(A, B) & \text{otherwise} \end{cases}$$

The first condition is not strictly necessary, as the applicability function can be crafted to prevent it from ever being evaluated. The second and third conditions choose an opinion based on the strength of the premises. Clearly, other F functions are also possible.

We make use of first order unification to transform an argument scheme into a concrete argument. Symbols found in the argument scheme's premises and conclusions are replaced with symbols found in the predicate's frame of discernment. As in most forms of unification, identical symbols are transformed into identical variables. Arguments are thus instantiated argument schemes.

Definition 9 *(Instantiated Argument)* *An instantiated argument for an argument scheme $(Name, Premises, Conclusions, F, A)$ is a tuple of the form $A = (Name, M)$ where M is a set of symbol pairs (s, l) such that $s \in \Sigma$ and $l \in \Phi$. Given that the argument scheme's Premises and Conclusions are of the*

form $(l_i, (s_1, \ldots, s_n))$, we have the restriction that $s \in \{s_1, \ldots, s_n\}$. Finally, given two pairs $(s_1, l_1), (s_2, l_2) \in M$, such that $s_1, s_2 \in S$ and $l_1, l_2 \in L$, $l_1 = l_2$ iff $s_1 = s_2$.

We name the set of all possible instantiated arguments $Args$.

Thus, for example, given the argument scheme for Modus Ponens defined above, and assuming a knowledge base containing the predicates $holds(a)$ and $implies(a, b)$ associated with sufficiently high levels of belief, we may generate the instantiated argument $(ModusPonens, \{(A, a), (B, b))\}$. Using instantiated arguments in this form is unwieldy as references to premises and conclusions from the argument scheme must constantly be made. We can write an instantiated argument in an abbreviated form. For example, given the previously described argument scheme for Modus Ponens, and the previously described instantiated argument, we will write

$$(ModusPonens, \{holds(a), implies(a, b)\}, \{holds(b)\}, A, F)$$

If the A and F functions are not used in the context in which we refer to the argument, we may leave them out.

Until now, we have described what individual arguments look like. However, arguments do not exist in isolation. Instead, they interact with each other, reinforcing or weakening opinions about predicates in the process. Unlike most other argumentation frameworks, we do not explicitly model rebutting and undercutting attacks to show how arguments interact. Instead, we use the concept of accrual of arguments to allow for both argument strengthening and weakening. To represent interactions between arguments, we must be able to answer the following question: what happens when two different arguments have opinions about a (partially shared) set of predicates in their conclusions?

The independent consensus operator gives us a default technique for applying accrual. Thus, given a set of arguments for and against a certain conclusion, and given no extra information, we apply the consensus operator based on the opinions garnered from the arguments to arrive at a final opinion for the conclusion. While the consensus operator works well in most cases, it fails in a number of situations. Specifically, accruals do not always accrue in the manner captured by this operator. We now explore some situations where this occurs.

Prakken [14] lays out three principles that the scheme of accrual of arguments adheres to. These are:

(1) Accruals are sometimes weaker than their elements.
(2) An accrual makes its elements inapplicable.
(3) Flawed reasons or arguments may not accrue.

The weakening of an accrual, according to Prakken, is due to the possibility that the accruing reasons are not independent. As an example, he suggests the case where two reasons exist not to go running, namely that it is hot, and that it is raining. However, for some runners, this specific combination of conditions may be less unpleasant than either condition alone, or may even be a reason to go running. Prakken claims that the interactions between reasons means that, in general, it is impossible to calculate the strength of an accrual from is accruing elements. We believe that while this claim is true in specific cases, in the general case the consensus operator combines accrual elements in an intuitively correct manner. Our framework is, however, designed to cater for special cases where the consensus operator should not be used. In these cases, a custom function is used instead of the consensus operator to perform the accrual.

While relatively obvious, the second principle is critical. Any framework supporting accrual of arguments must not count evidence twice. However, if the accrual is defeated, its component parts should be reinstated.

The final point states that if a component within an accrual is defeated, it should not be counted when performing the accrual; otherwise this might mean that the accrual as a whole becomes invalid.

While some researchers have suggested that accrual of arguments is an argument scheme and can be treated as such (arguably, for example [13]), Prakken's view, in our understanding, is that the best way to handle accrual of arguments is by following a two stage process. First, determine what arguments may enter into an accrual, and second compute the effects of the accrual. We agree that accrual of arguments cannot be treated as "just another" argument scheme due to its role and nature. We believe, however, that in certain situations (usually obeying principle 1), accrual of evidence can be treated as an argument scheme. The way in which our framework aligns these two views is one of its most unique aspects.

We will provide an informal outline of how we approach accrual of arguments before giving a formal description of the process. Informally, given multiple arguments for a conclusion, we apply the standard consensus rule. However, if an argument is advanced which subsumes (some of the) arguments which take part in the consensus, the subsumed argument's conclusions are ignored, and the subsuming rule is used instead. If any of those arguments are attacked and defeated, then our accrual rule is itself defeated, allowing all its undefeated (and previously subsumed) members to act again. If some of the newly activated sub-members were, in turn, part of accruals, those accruals would enter into force again.

We claim that an argument subsumes another if the subsumed argument's

premises are a subset of the subsuming argument's premises, and at least one conclusion is shared. However, any conclusions that are not shared are still in force. Thus, for example, given the three arguments "if it is raining, we do not run", "if it is hot, we do not run", and "if it is hot and raining, we do run", it is clear that the third argument would subsume the other two. It is also clear that if the first argument is changed to read "if it is raining, we do not run and do not hang washing out to dry", we would run, but still not hang washing out to dry. Formally,

Definition 10 *(Argument Subsumption)* *Given two instantiated arguments (written in abbreviated form),*
$(Arg1, \{p_{11}, \ldots, p_{l1}\}, \{c_{11}, \ldots, c_{m1}\})$, and $(Arg2, \{p_{12}, \ldots, p_{n2}\}, \{c_{12}, \ldots, c_{o2}\})$,
we say that $Arg2$ subsumes $Arg1$ for a set of conclusions C iff the following two conditions hold:

(1) for all p_{i1}, $i = 1 \ldots l$ there is a $j \in \{1 \ldots l\}$ such that $p_{i1} = p_{j2}$.
(2) for some $1 < i < m$ and $1 < j < o$, $c_{i1} = c_{j2}$ and $c_{i1}, c_{j2} \in C$

If $Arg2$ subsumes $Arg1$ for a set of conclusions C, we may write $Arg2 \gg_C Arg1$.

$Arg2$ maximally subsumes $Arg1$ for a set of conclusions C if $Arg2$ subsumes $Arg1$ for a set of conclusions C and there is no set of conclusions D for which $Arg2$ subsumes $Arg1$ such that $|D| > |C|$.

When given multiple arguments for a conclusion, we apply only the argument that maximally subsumes all other arguments for that conclusion. If any arguments remain that can be applied, they are combined using the Subjective Logic independent consensus operator.

We are now in a position to provide an algorithm for evaluating how sets of (instantiated) arguments interact. Our algorithm is shown in Figure 3; it is inspired by the way reasoning is performed in probabilistic networks, and, in fact, is best explained by thinking of our sets of arguments and predicates as a graph. Both predicates and arguments can be thought of as nodes, with a directed edge between the two if the predicate appears in the premises or conclusions of an argument. The edge enters the argument in the case of the predicate being a premise, and exits the argument otherwise.

One weakness of our approach is the assumption that our argument graph is acyclic. This makes it difficult to represent certain classes of arguments such as "a holds iff b holds". Another family of cycles that can arise in the graph involves self reinforcing, or self defeating chains of argument. However, due to the nature of our algorithm, this class of argument does not pose as big a problem. Some possible solutions to this issue are discussed in Section 5.

Given: a set of instantiated arguments A
      a set of possible facts and associated opinions PF, ω_{PF}
Variables:
      visitedFacts VF
      visitedArguments VA

$VF = PF$

repeat until $A = VA$
  $\forall a = (p, c, applicable, F) \in A \backslash VA$
    if $\forall p_i \in p, p_i \in VF$
      if $admissible(a)$
        $VA = VA \cup A$
      else
        $A = A \backslash a$
  $\forall r \in PF \backslash VF$
    if $\nexists a \in A \backslash VA$ such that $r \in c(a)$
      $\forall a \in VA$ such that $r \in c(a)$ and $\nexists a_2 \in VA$ such that $a_2(c(a)) \gg a(c(a))$
        $\omega_{PF}(r) = \bigoplus(F(a))$
      $VF = VF \cup r$
    else if $\nexists a \in A$ such that $r \in c(a)$
      $\omega_{PF}(r) = \langle 0, 0, 1 \rangle$
      $VF = VF \cup r$

Fig. 3. An algorithm to compute conclusions given a set of argument schemes, instantiated arguments, and optionally, some opinions. $c(a)$ represents the conclusions of instantiated argument a, $F(a)$ the application of a's F function, and $a_2(c(a))$ and $a(c(a))$ represent the arguments whose conclusions are $c(a)$. Note that the abbreviated form of an instantiated argument is used in the algorithm.

To operate, our algorithm requires an argument graph, as well as a starting set of opinions. We assume that these opinions are not under dispute, and the associated nodes must, therefore, have no edges leading into them. Our algorithm then propagates these opinions forward through the graph, until all applicable arguments in the graph have been taken into account. A few issues need to be taken into consideration to ensure the proper functioning of our algorithm:

- Only justified arguments should be used.
- Defaults must be taken into account.
- It must be possible to differentiate between visited and unvisited nodes.
- All, or only some of an argument's conclusions may participate in an accrual.

A predicate node is assigned an opinion if all the argument nodes leading into it have associated opinions (taking into account accrual of arguments). An

argument node is evaluated if all predicate nodes leading into it are assigned an opinion, *unless the predicate node has no arguments leading into it* (for example due to argument defeat) in which case they are assigned a default opinion of $\langle 0, 0, 1 \rangle$. Our algorithm terminates in $O(n)$ time, where n is the number of edges.

If an argument is not admissible, it is removed from evaluation. It should be noted that once an argument is removed, it cannot be reinstated. However, arguments are only removed when there is no chance that they will be admissible, so the framework yields the same results as our intuition. As we will discuss in Section 5, there is only a weak relation between our semantics and Dung's argumentation semantics. We also defer discussion of other representational issues relating to the underlying frameworkto Section 5.

At this point, we have a way of determining which conclusions hold given a set of arguments. Next, we describe a procedure for how the set of arguments is generated. This is done in two parts. We assume that our argument framework is used within the context of a dialogue. The utterances made in the course of the dialogue result in the set of arguments. Thus, we begin by formalising the dialogue process, after which we provide a decision rule which dialogue participants can use to determine which arguments they should advance at any point in the dialogue. Once this is done, we can show how agents, arguments and argument schemes interact to form our complete framework.

To specify the dialogue, we need to further constrain and describe the environment in which it takes place. We assume that dialogue occurs between two or more agents, each of which has a private knowledge base, opinions about the environment, and goals. The dialogue environment contains a public commitment store into which the agent's arguments are inserted, as well as the set of valid argument schemes. Since we are interested in arguing about evidence in partially observable domains, we make the assumption that the environment holds a set of sensors. These sensors may be probed to obtain opinions about the value of various relations. In practise, sensors may be agents, static parts of the environment, or some other entity capable of providing an opinion about the environment. We assume that multiple sensors can give opinions about the same relations, and that some sensors are more reliable than others.

Definition 11 *(Environment)* *Given a universe of discourse* (PF, AS), *The environment Env is a tuple* $(Agents, CS, S, PC)$ *where Agents is the set of agents operating in the environment,* $CS \subseteq PF$ *is the commitment store (a public knowledge base of arguments), and* S *is the set of sensors present in the environment.* $PC : 2^S \to \mathbb{R}$ *is the sensor probing cost function.*

Definition 12 *(Agents)* *Given environment* $Env = (Agents, CS, S, PC)$, *an agent* $\alpha \in Agents$ *is a tuple* $(Name, KB, G, C)$ *consisting of the agent's*

name, Name, a private knowledge base $KB \subseteq PF$ containing opinions about the environment, a goal function $G : \Theta \to \mathbb{R}$ mapping combinations of opinions on predicates (obtained by looking at the frame of discernment) to utility values, and a variable $C \in \mathbb{R}$ to keep track of the agent's utility cost.

Definition 13 (Sensors) *A sensor s is a structure (Ω_s, Ω_p). Ω_s is a set containing predicate, opinion pairs representing the reliability of a sensor with respect to the predicate. Ω_p is another predicate, opinion pair which stores the sensor's opinion regarding the state of the predicate.*

Agents take turns to advance a line of argument (consisting of one or more instantiated arguments), and probe sensors to obtain more information about the environment. Such an action is called an utterance. In each turn, the contents of an agent's utterance is added to the commitment store; any sensors probed are marked as such (a sensor may not be probed more than once for the value of a specific relation), and costs are updated. Once made, there is no way to withdraw the contents of an utterance from the commitment store.

Definition 14 (Utterances) *The utterance function*

$$utterance : Environment \times Name \to 2^{Args} \times Probes$$

takes in an environment and an agent (via its name), and returns the utterance made by the agent. The first part of the utterance lists the arguments advanced by the agent, while the second lists the probes the agents would like to undertake where $Probes \in 2^S$.

Definition 15 (Turns) *The turn function*

$$turn : Environment \times Name \to Environment$$

takes in an environment and an agent label, and returns a new environment containing the effects of an agent's utterance.

In our framework, the turn function is defined as $turn = (NewAgents, CS \cup Ar, NewSensors, PC)$ where $Ar, NewAgents$ and $NewSensors$ are computed from the results of the utterance function. Assuming that the agent making the utterance is agent α, if $utterance(Env, Name) = (Ar, Probes)$ then $NewAgents = (Agents \setminus \{\alpha\}) \cup (Name, KB, G, C + PC(Probes))$ and, $\forall s, l \in Probes$, where l is a predicate that sensor s is able to probe, $NewSensors = (Sensors \setminus \{s\}) \cup (\Omega_s, \Omega_p \cup \omega_p(l)))$.

It should be noted that the *utterance* depends on agent strategy; one possible *utterance* function was described in [11], and will be described briefly later. Before doing so, we must define a protocol which agents may use to argue with each other. This protocol, often referred to as a dialogue game [9], contains only one locution (in which agents advance an argument and probe sensors), and

allows agents to alternate in making utterances. More complicated dialogue games are also possible, but are not examined here as they are auxiliary to the focus of this paper.

We may assume that our agents are named $Agent_0, Agent_1, \ldots, Agents_{n-1}$ where n is the number of agents participating in the dialogue. We can define the dialogue game in terms of the turn function by setting $turn_0 = turn((Agents, CS_0, S, admissible, PC), Agent_0)$, and then having $turn_{i+1} = turn(turn_i, Agent_{i \bmod n})$. The game ends if $turn_i \ldots turn_{i-n+1} = turn_{i-n}$.

When the dialogue starts, CS_0 contains publicly known arguments. It is usually empty. It should be noted that an agent may make a null utterance $\{,\}$ during its turn to (eventually) bring the game to an end. In fact, given a finite number of arguments and sensors, it should be clear that the dialogue is guaranteed to terminate, as, eventually, no utterances will be possible that will modify the public knowledge base CS.

At any time, we may compute an agent's utility by combining its utility gain (for achieving its goals) with its current costs. At any stage of the dialogue, given the environment's CS, and the set of all opinions probed by the sensors $\{\Omega_p | s = (\Omega_s, \Omega_p) \in S\}$, as well as the set of legal argument schemes, we can run the reasoning algorithm to compute the set of "proven" relations; that is, relations for which an opinion exceeds a predetermined admissibility bound. Similarly, we can determine which relations have their negation proven, and which relations are simply unproven.

Given an environment CS, the set of all opinions probed by the sensors $\{\Omega_p | s = (\Omega_s, \Omega_p) \in S)\}$ and an admissibility function on opinions $Admissible(\omega) \rightarrow [true, false, unknown]$, we can run the reasoning algorithm over all possible facts to create a set of true, false and unproven predicates. If we name these sets f_{true}, f_{false} and $f_{unknown}$, then the agent's net utility gain is $G(f_{true}, f_{false}, f_{unknown}) - C$, where C is the agent's utility cost.

At the end of the dialogue, we assume that agents agree that literals in the f_{true} and f_{false} sets hold in the environment.

One simple decision procedure for an agent (described in detail in [11]) involves it performing one step look-ahead to decide which utterance to make. The agent computes what probes it can make by looking at what sensors have not yet been probed, and what arguments it can advance (by looking at its knowledge base, the commitment store, and the set of argument schemes). It then calculates the utility gain for each combination of probes and advanced arguments, advancing the ones that maximise its utility.

4 Example

In this section, we describe a dialogue in a hypothetical bridge building scenario. Two agents, α and β must use these argument schemes to have a discussion about the amount of concrete and steel needed to build a bridge. Agent α's goal involves attempting to minimise the amount of steel needed — α is responsible for the supply of steel. This may be achieved by showing that the environment is in such a state where little steel, and lots of concrete is needed. β is responsible for providing the bridge's concrete, and would like to minimise the amount of concrete used. In the interests of clarity, our description is semi-formal.

Assume we have the following general argument schemes (here, $ArgExpertOp$ is the scheme for an argument from expert opinion [20]:

Name	Premises	Conclusions	A	F
$ModusPonens$	$\{A, implies(A, B)\}$	$\{B\}$	A_1	F_1
$ArgExpertOp$	$\{expert(E, D), claims(E, A),$	$\{A\}$	A_2	F_2
	$inDomain(A, D)\}$			
D_1	$\{sand(L), support(X, L)\}$	$\{concrete(X)\}$	A_3	F_3
D_2	$\{rock(L), support(X, L)\}$	$\{steel(X)\}$	A_4	F_4
D_3	$\{mud(L), support(X, L)\}$	$\{concrete(X)\}$	A_5	F_5
D_4	$\{rock(L), sand(L),$	$\{steel(X),$	A_6	F_6
	$support(X, L)\}$	$concrete(X)\}$		

where

$A_1 : true$ if $b(A)$ and $b(implies(A, B))$ are both ≥ 0.5 else $false$

$$F_1 : \omega(B) = \begin{cases} \langle 0, 0, 1 \rangle & \text{if } b(holds(A)) < 0.5 \text{ or} \\ & b(implies(A, B)) < 0.5 \\ \omega(holds(A)) & \text{if } b(holds(A) < b(implies(A, B)) \\ \omega(implies(A, B)) & \text{otherwise} \end{cases}$$

$A_2 : true$ if $d(expert(E, D)) < 0.5$ & $d(inDomain(A, D)) < 0.5$

$F_2 : \omega(A) = claims(E, A)$

For $D_1 \ldots D_4$, the admissibility function requires that the belief in rock, sand

or mud be greater than 0.5, and the F function sets the conclusions to the same strength as the premises, except for D_4. Here, $steel(X)$ is set to the average value of $rock(L)$ and $sand(L)$, while $concrete(X)$ is set to $\neg steel(X)$.

The top level frame of discernment is created using the predicates

$$sand, rock, mud, implies, expert, claims, inDomain, fastWater$$

The $sand, rock, mud$ and $fastWater$ frames of discernment contain l (representing a location). The $implies$ frame of discernment contains the tuple $(fastWater, mud)$, while $expert$ contains $geologist$. $inDomain$ contains the tuple $(geology, sand)$, and $claims$ contains

$$(geologist, sand), (geologist, mud), (geologist, rock)$$

Let α have an opinion of $\langle 0.9, 0, 0.1 \rangle$ regarding $sand(l)$ stored in it's knowledge base. Assume that it also believes $fastWater(l)$ and $claims(geologist, sand(l))$.

Finally, let the environment contain sensors s_1, s_2, s_3, s_4, with the first two being able to monitor the status of $sand(l)$, s_3 able to observe all states in the $claims$ frames of discernment, and s_4 being able to detect $fastWater(l)$. Let s_1 also be able to discern the status of $rock(l)$. We associate opinions $\langle 0.7, 0.2, 0.1 \rangle, \langle 0.8, 0.2, 0.1 \rangle, \langle 0.9, 0, 0.1 \rangle$ and $\langle 0.8, 0.1, 0.1 \rangle$ regarding the sensors' respective reliabilities.

Agent α begins the conversation by making the utterance

$$((D1, \{sand(l), support(bridge, l)\}, \{concrete(bridge)\}), \{(s_1, sand(l))\})$$

That is, it probes whether sand exists in the location the bridge is to be built, and instantiates an argument based on D_1 claiming that since it is sandy, a large amount of concrete is required for the bridge[2]. Assume that s_1 returns a value of $\langle 0.7, 0, 0.3 \rangle$. This means that $sand(l)$ is now associated with an opinion of $\langle 0.56, 0.07, 0.37 \rangle$ in the CS.

Agent β responds by probing another sensor

$$(, \{(s_2, sand(l))\})$$

to show that the bank is not in fact sandy. This sensor returns an opinion $\langle 0.3, 0.6, 0.1 \rangle$, meaning that the $\omega(sand(l))$ is now $\langle 0.45, 0.36, 0.09 \rangle$. Agent α's argument is now no longer admissible.

---

[2] The probe may return values, contrary to the agent's belief, that cause the argument to be inapplicable.

No more sensors exist that can be used to determine whether $sand(l)$ holds or not. Agent α thus advances the argument

$$((ArgExpertOp, \{expert(geologist, geology),$$

$$claims(geologist, sand(l)), indomain(geology, sand(l))\}, \{sand(l)\}),$$

$$\{(s_3, claims(geologist, sand(l)))\})$$

The probe here represents asking the witness for their testimony. Assume that the witness returns an opinion of $\langle 0.8, 0.1, 0.1 \rangle$. $sand(l)$ would now once again be a justified conclusion. At this point, it should be noted that the admissibility requirements for the argument from expert opinion mean that the burden of proof is assigned to the agent challenging the argument. More complex behaviour, such as requiring the agent introducing the argument to justify their assumptions, is easily introduced by changing the form of A and F.

Agent β now turns to an argument using accruals. It points out that

$$((D_4, \{rock(l), support(bridge, l)\}, \{concrete(bridge), steel(bridge)\}),$$

$$\{(s_1, rock(l))\})$$

In other words, the presence of rock together with sand means that steel rather than concrete is required. Since this argument scheme's F function is not dogmatic, the conclusion for concrete is weakened, but not eliminated.

Finally, α responds with the argument

$$(\{(ModusPonens, \{fastWater(l), implies(fastwater(l), mud(l))\},$$
$$\{mud(l)\}), ((D_3, \{mud(l), support(bridge, l)\},$$
$$\{concrete(bridge)\}))\}, \{(s_4, fastWater(l))\})$$

That is, by showing that there is fast water at location l, it supports the conclusion that mud exists at l. The existence of mud means that argument scheme D_3 can be used, which accrues (via the default consensus operator) with the current opinion regarding the need for concrete.

At this point in the conversation, the agents have no further arguments to advance, and the dialogue terminates. predicates $concrete(l)$ and $steel(l)$ have opinions associated with them. Depending on the form of the *admissibility* function, they, or their negation, may be judged as proven or unproven. If, for
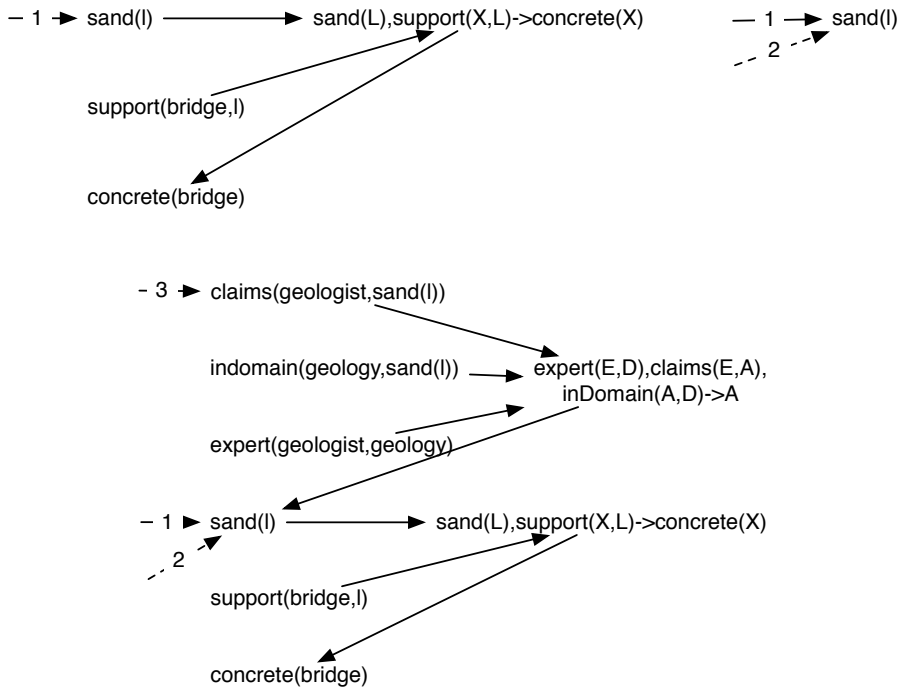
- 1 ▶ sand(l) ⟶ sand(L),support(X,L)->concrete(X)   — 1 ▶ sand(l)
                                                            - - 2

support(bridge,l)

concrete(bridge)

- 3 ▶ claims(geologist,sand(l))

indomain(geology,sand(l)) ⟶ expert(E,D),claims(E,A),
                              inDomain(A,D)->A

expert(geologist,geology)

- 1 ▶ sand(l) ⟶ sand(L),support(X,L)->concrete(X)
  - - 2

support(bridge,l)

concrete(bridge)

Fig. 4. The argument graph obtained after the first three utterances are made. The numbers show during which turn a sensor probe was made. Solid arrows indicate support for an argument or predicate, while dashed lines represent an attack.

example, $concrete(l)$ is judged to be admissible, both agents would agree that more concrete should be used at the site.

The first three utterances of the dialogue are shown in Figure 4. It is assumed that any unprobed predicates were in the commitment store at the start of the dialogue, together with their associated opinions. As can be seen, after the second utterance, the low opinion associated with $sand(l)$ means that the argument advanced at the start of the dialogue is no longer deemed applicable. Thus, the opinion associated with $concrete(bridge)$ would revert to its default value. The argument graph for the entire dialogue is shown in Figure 5.

5 Discussion

In this section, we examine some of the novel features of our framework in detail, as well as looking at related research and possible future work.

Our framework was designed to allow for complex argument to take place, particularly in the domain of evidential reasoning. Uncertainty is a key feature of such domains, hence our decision to base our framework on Subjective Logic. Catering for uncertainty in argumentation frameworks is by no means new. Pollock [13] made probability a central feature of his OSCAR architecture.
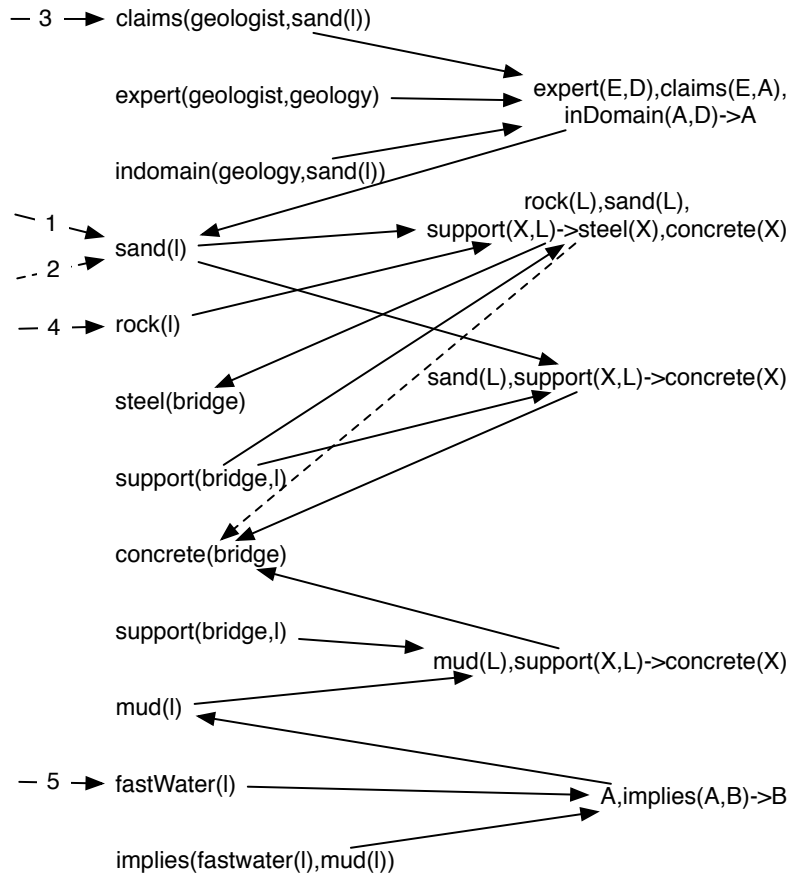
Fig. 5. The complete argument graph for the dialogue. The numbers show during which turn a sensor probe was made. Solid arrows indicate support for an argument or predicate, while dashed lines represent an attack.

We disagree with his extensive use of the "weakest link" principle, however, believing that, while it may hold in general, it is not always applicable (as mentioned in [14]. His use of probability, rather than uncertainty is another point at which our approaches diverge. Other notable work includes that of Vreeswijk [18], and Haenni [5]. The latter suggests an approach called "Probabilistic argumentation systems". These systems are designed to perform inference under uncertainty within conflicting knowledge bases. While lacking a dialogical aspect, he shows a relation between his work and Dempster-Schafer theory.

Our use of Subjective Logic as the basis of the framework provides us with a large amount of representational richness. Not only are we able to represent probability (via belief), but we are also able to speak about ignorance (via uncertainty). Differentiating between these two concepts lets us represent defaults in a natural, and elegant way. A default can be represented by specifying, within the A function, that a conclusion may hold as long as the disbelief for a premise remains below a certain threshold. By requiring that belief remain above some threshold, normal premises can also be represented.

22

A simple example of this was provided in the previous section, where everyone, by default, is assumed to be an expert. Burden of proof [19] is very closely related to defaults, and we model it in the same way.

Argument schemes have been extensively discussed in the literature (see for example [3,20]). A small, but growing number of argumentation frameworks provide explicit support for argument schemes (e.g. [17]). We believe that supporting argument schemes in our framework not only enhances argument understanding, but that such support also provides clear practical advantages, including the separation of domain and argument knowledge, re-usability, and a possible reduction in computational complexity when deciding what arguments to advance. The separation between arguments and agent knowledge created by argument schemes raises the intriguing possibility of the modification and dynamic creation of argument schemes during a dialogue.

We have separated out the F and A functions within our representation of argument schemes as we believe that in some situations, the decision regarding whether an argument scheme is applicable, and how strongly it supports its conclusions, are independent of each other. Separating out the two functions is also practically useful; by explicitly excluding an argument based on the result of its A function, we can avoid extra calculations in our algorithm. Agents can also use the A function in the heuristic to avoid considering the application of invalid argument schemes.

Jøsang has proposed a large number of additional operators for use in Subjective Logic which have not been mentioned in this paper (see for example [8]). Many of these operators appear to encapsulate common forms of reasoning about evidence, and can thus (with appropriate restrictions on premises and conclusions) form the basis of an argument scheme's F function. We believe that using our framework as a basis for investigating additional possible Subjective Logic operators may be fruitful.

Another area in which we plan to extend the framework involves unification and quantification. At the moment, we perform universal quantification over all the elements of a frame of discernment. That is, if the *expert* frame of discernment contains $(geologist, mud)$, we could deduce

$$expert(geologist(fred), mud(l)), expert(geologist(mary), mud(m))$$

with no way of specifying that in fact, *fred* is only an expert in mud at location l, not location m.

The interplay between sensors and arguments is an area in which little formal work has been done [12]. While our model is very simple, it elegantly captures the fact that sensor data is inherently unreliable in many situations. Enriching our model of sensors is one area in which we plan to do future work.

Our model was designed with support for accrual of arguments in mind. The way in which we deal with accrual of arguments, while powerful, is still limited, and overcoming these limitations is a priority. We are unable to handle accruals in which the accrued and accruing arguments share no conclusions. While it is often possible to add an explicit negated conclusion so as to negate the accrued argument's conclusions, this is incorrect in some situations. Instead, we recommend extending the framework by introducing an explicit accrual relationship between argument schemes. Representing such situations requires inputting additional domain knowledge into the framework, the domain-specific nature of such accruals means that no other way to handle them exists.

We are currently investigating what effects a mapping between our model and a Dung-like abstract model [4] will have. By allowing for linked arguments [10] and support between arguments [1], the translation of the graph that results from an application of the model to one embedded in an abstract model allows us to cater for some types of loops. However, this approach does not allow us to deal with argument strength in a satisfactory manner, meaning that techniques for representing self-reinforcing arguments must still be investigated.

The dialogue game we have proposed is very simple, and only guarantees dialogue termination (given a finite number of argumentation schemes, finite knowledge bases, and a finite number of sensors sensors). Dispute focus, that is, ensuring that agents advance arguments relevant to the conversation, is provided by the utility based argument heuristic. Also related to the dialogue game, as well as the structure of the agents and environment, is the problem of advancing an argument without proof. Depending on the structure of an argument scheme's F function, a sensor must always be probed before an argument may be advanced. Thus, the initial burden of proof always falls on the agent making an utterance. While some may claim that such an approach makes sense (after all, even commonsense knowledge requires some sort of shared experience, which could be viewed as a sensor), others may view this as a weakness of the system. There are a number of ways to avoid this issue. The most obvious is to set up initial predicates in the commitment store with the appropriate belief values. Another way would involve the addition of a single zero utility cost sensor that an agent may probe to set a predicate's initial value. It is also possible to craft the F function to allow for certain claims with no proof. None of these approaches are fully satisfying, as they allow for only one side of a claim to be made without the need for proof. Allowing agents to act as sensors might be a better way of overcoming this problem, and the addition of explicit "claim" and "challenge" moves in the dialogue is another way to attack this issue. The decision procedure we have described is based on some of our earlier work [11]. Other researchers have advanced other possible approaches to argument selection [2], and it would be interesting to integrate these techniques into our work.

6 Conclusions

Argumentation is a well recognised, powerful reasoning technique. With a few notable exceptions however, argument frameworks have had difficulty operating in domains where uncertainty was present. Furthermore, most argument frameworks have examined only a single aspect of the problem of argument, be it the underlying logical representation, the interaction between arguments, or an illustration of a protocol for argument.

In this paper, we presented a framework for argumentation in domains containing uncertainty. The concept of argument schemes is built into the framework, allowing for a rich set of primitives to be utilised in the argumentation process. We have also attempted to cater for other important concepts in argument such as accrual of arguments, defaults, and burden of proof. While the lowest levels of the framework are general enough to be applied to almost area in which argument is used, the higher levels are aimed at evidential reasoning. To this end, we introduced the concept of sensors, abstracting the notion of obtaining information from the environment. Finally, we introduced a dialogue game and a very simple decision procedure allowing agents using the framework to decide which arguments to advance.

Acknowledgements

References

[1] L. Amgoud, C. Cayrol, and M.-C. Lagasquie-Schiex. On the bipolarity in argumentation frameworks. In *Proceedings of the 10th International Workshop on Non-monotonic Reasoning*, pages 1–9, Whistler, Canada, 2004.

[2] L. Amgoud and H. Prade. Reaching agreement through argumentation: a possiblistic approach. In *Proc. of the 9th Int. Conf. on the Principles of Knowledge Representation and Reasoning*, pages 175–182, 2004.

[3] F. Bex, H. Prakken, C. Reed, and D. Walton. Towards a formal account of reasoning about evidence: Argumentation schemes and generalisations. *Artificial Intelligence and Law*, 11(2-3):125–165, 2003.

[4] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.

[5] R. Haenni. Towards a unifying theory of logical and probabilistic reasoning. In *Proc. of the 4th Int. Symp. on Imprecise Probabilties and Their Applications*, pages 193–202, Pittsburgh, Pennsylvania, USA, 2005.

[6] A. Jøsang. A logic for uncertain probabilities. *Int. Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9:279–311, 2001.

[7] A. Jøsang and D. McAnally. Multiplication and comultiplication of beliefs. *Int. Journal of Approximate Reasoning*, 38(1):19–51, 2004.

[8] A. Jøsang, S. Pope, and M. Daniel. Conditional deduction under uncertainty. In *Proc. of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 824–835, Barcelona, Spain, 2005.

[9] P. McBurney and S. Parsons. Dialogue games in multi-agent systems. *Informal Logic*, 22(3):257–274, 2002.

[10] S. H. Nielsen and S. Parsons. A generalization of Dung's abstract framework for argumentation: Arguing with sets of attacking arguments. In *Proc. of ArgMAS 2006*, pages 7–19, Hakodate, Japan, 2006.

[11] N. Oren, T. J. Norman, and A. Preece. Arguing with confidential information. In *Proc. of the 18th European Conf. on Artificial Intelligence*, pages 280–284, Riva del Garda, Italy, August 2006.

[12] N. Oren, T. J. Norman, and A. Preece. Argumentation based contract monitoring in uncertain domains. In *Proc. of the 20th Int. Joint Conf. on Artificial Intelligence*, pages 1434–1439, Hyderabad, India, 2007.

[13] J. L. Pollock. *Cognitive Carpentry*. Bradford/MIT Press, 1995.

[14] H. Prakken. A study of accrual of arguments, with applications to evidential reasoning. In *Proc. of the 10th Int. Conf. on Artificial Intelligence and Law*, pages 85–94, 2005.

[15] H. Prakken and G. Sartor. *Computational Logic: Logic Programming and Beyond. Essays In Honour of Robert A. Kowalski, Part II*, volume 2048 of *LNCS*, pages 342–380. Springer-Verlag, 2002.

[16] C. A. Reed and T. J. Norman, editors. *Argumentation Machines: New frontiers in argumentation and computation*. Kluwer, 2003.

[17] B. Verheij. Dialectical argumentation with argumentation schemes: An approach to legal logic. *Artificial intelligence and Law*, 11:167–195, 2003.

[18] G. A. W. Vreeswijk. Argumentation in Bayesian belief networks. In *Proc. of ArgMAS 2004*, number 3366 in LNAI, pages 111–129. Springer, 2004.

[19] D. N. Walton. Burden of proof. *Argumentation*, 2:233–254, 1988.

[20] D. N. Walton. *Argumentation Schemes for Presumptive Reasoning*. Erlbaum, 1996.