Justified argument revision in agent dialogue

Mark Snaith and Chris Reed

School of Computing, University of Dundee, Dundee, DD1 4HN, UK
{marksnaith,chris}@computing.dundee.ac.uk

Abstract. In certain dialogue protocols, when an agent concedes information that conflicts with an existing commitment, its opponent can force it into retracting that existing commitment, as well as other commitments from which the retracted commitment can be inferred. Belief revision studies how a minimal amount of information can be given up in the face of new, conflicting information, with prominent theories in this field being applied belief sets, expressed in propositional logic. While many dialogue games also find their roots in propositional logic, relatively recent advances have seen these games adapted for use with systems of argumentation, which offer more sophisticated models of reasoning and conflict beyond propositional inference and classical negation. In this paper, we define and describe the properties of two operators for the modification of a knowledge base in a system of structured argumentation. We then go on to present a method by which an agent can determine the minimal change when applying these operators in order to justify the addition or removal of propositions to bring about consistency in its commitment store.

1 Introduction

During the course of an argumentative dialogue, a software agent may be forced into conceding information that conflicts with existing commitments. In certain dialogue games, the agent's opponent can then force it into retracting those original commitments, and if they are a consequence of other commitments, then those other commitments should also be withdrawn, in what [13] terms a "stability adjustment". However, if a commitment is a consequence of several other commitments working in combination, there will be multiple possible ways in which the original commitment can be given up. This presents the agent with the problem of exactly which commitment(s) it should give up.

The field of belief revision aims to answer a related question, in terms of belief sets: when an agent is required to give up a belief and faces a choice as to exactly which belief, how it does make the choice? Once of the most influential theories in belief revision is the AGM theory, which provides a set of postulates describing valid *revisions*, *contractions* and *expansions* of belief sets [1]. These three processes are additionally guided by the concept of minimal change, with "minimal" being measured in terms of epistemic entrenchment — those beliefs with the lowest degree of entrenchment are more willingly given up [6, 7].

A significant drawback of applying the AGM theory in the context of multi-agent dialogues is that it only applies to a knowledge base consisting of sentences in some logical language. While it is true that classic dialogue games find their roots in propositional logic (to which the AGM theory can be applied), recent developments (for instance [3]) have seen dialogue games adapted for use with systems of argumentation, such as the abstract frameworks of [4], which offers a more sophisticated model of reasoning based on attacks between arguments. A recent extension to Dung's work has been to instantiate the abstract approach by incorporating previous work on structured argumentation to provide arguments with structure, through the application of strict and defeasible inference rules to a knowledge base [10]. This in turn has been applied in a dialogical context in [14].

The ASPIC^+ framework of [10] provides structure to arguments through a logical language which, although left unspecified, can be propositional, providing a natural link back both to classic dialogue games and the original AGM theory. It is, therefore, possible to determine an entrenchment ordering over the knowledge base in an argumentation system, however the question remains as to exactly how this should be done. ASPIC^+ contains partial pre-orders over the knowledge base and rule set, which in turn are used to create a preference ordering over arguments that is used in the process of argument evaluation. However, these preference orderings are both not exhaustive, and they capture different concepts to an entrenchment ordering. An entrenchment ordering is used in determining the importance of a belief with respect to other beliefs, while a preference ordering over arguments is used in the process of evaluation, in order to determine whether or not one argument defeats another.s

Despite the ability to use a propositional language in [10], the tools and techniques of the AGM theory remain insufficient in exploring minimal change when carrying out revisions, because additional features such as argument acceptability, preferences, contrariness and rules are not accounted for. We therefore require a new approach to measuring minimal change that takes into account not only the knowledge base, but also the models of reasoning employed in systems of structured argumentation.

Connections between argumentation and belief revision have recently found new momentum [5]. The work of [11, 12, 8] on Argument Theory Change (ATC) sees belief revision techniques employed to revise an argumentation system when a new argument is added, such that the argument becomes warranted. However, ATC does not consider the process of either removing or changing the acceptability of an argument, and how minimal change can be determined when doing so.

In this paper, we describe a method by which a software agent can reason about the ways in which it can modify the knowledge base in an argumentation system in order to bring about a change in its commitments in a dialogue. We do this by taking an initial look at the effects of contracting and expanding an argumentation system by, respectively, removing formulae from and adding formulae to its knowledge base. These effects will consider the impact on other arguments in the system in terms of construction and acceptability.

The paper proceeds as follows; in section 2 we provide a brief introduction to the system of [10] and to belief revision; in section 3 we briefly describe a dialogue framework; in section 4 we define, and describe the properties of, our change operators for argumentation systems; in section 5 we provide an example and in section 6 we conclude the paper and offer directions for potential future work.

2 Preliminaries

In this section, we provide a brief introduction to previous work upon which this paper will build.

2.1 Argumentation

The ASPIC$^+$ framework [10] further developed the work of [2] and instantiates the abstract approach to argumentation in [4]. The basic notion of the framework is an argumentation system:

Definition 1. *An argumentation system* $\mathcal{AS} = \langle \mathcal{L}, ^-, \mathcal{R}, \leq \rangle$ *where* $\mathcal{L}$ *is a logical language,* $^-$ *is a contrariness function from* $\mathcal{L}$ *to* $2^{\mathcal{L}}$*,* $\mathcal{R} = \mathcal{R}_s \cup \mathcal{R}_d$ *is a set of strict* $\mathcal{R}_s$ *and defeasible* $\mathcal{R}_d$ *inference rules such that* $\mathcal{R}_s \cap \mathcal{R}_d = \emptyset$ *and* $\leq$ *is a partial preorder on* $\mathcal{R}_d$*.*

Remark 1. The contrariness function is represented as $p \in \overline{q}$*, which means "p is a contrary of q". Where p and q are contraries of each other (i.e.* $p \in \overline{q}$ *and* $q \in \overline{p}$*) they are said to be contradictory, which is represented as* $p = -q$*.*

An argumentation system contains a knowledge base, $\langle \mathcal{K}, \leq' \rangle$ where $\mathcal{K} \subseteq \mathcal{L}$ and $\leq'$ is a partial preorder on $\mathcal{K} \backslash \mathcal{K}_n$. $\mathcal{K} = \mathcal{K}_n \cup \mathcal{K}_p \cup \mathcal{K}_a \cup \mathcal{K}_i$, where $\mathcal{K}_n$ is a set of (necessary) axioms, $\mathcal{K}_p$ is a set of ordinary premises, $\mathcal{K}_a$ is a set of assumptions and $\mathcal{K}_i$ is a set of issues.

From the knowledge base ($\mathcal{K}$) and rules ($\mathcal{R}$), arguments are constructed. For an argument $\mathcal{A}$, $Prem(\mathcal{A})$ is a function that returns all the premises in $\mathcal{A}$; $Conc(\mathcal{A})$ is a function that returns the conclusion of $\mathcal{A}$; $Sub(\mathcal{A})$ is a function that returns all the sub-arguments of $\mathcal{A}$; $DefRules(\mathcal{A})$ is a function that returns all defeasible rules in $\mathcal{A}$; and $TopRule(\mathcal{A})$ is a function that returns the last inference rule use in $\mathcal{A}$.

On the basis of these functions, $\mathcal{A}$ is:

1. φ if $\varphi \in \mathcal{K}$ with, $Prem(\mathcal{A}) = \{\varphi\}$, $Conc(\mathcal{A}) = \varphi$, $Sub(\mathcal{A}) = \{\varphi\}$, $DefRules(\mathcal{A}) = \emptyset$ and $TopRule(\mathcal{A}) =$ undefined.
2. $\mathcal{A}_1, \ldots, \mathcal{A}_n \rightarrow \psi$ if $\mathcal{A}_1, \ldots, \mathcal{A}_n$ are arguments such that there exists a strict rule $Conc(\mathcal{A}_1), \ldots Conc(\mathcal{A}_n) \rightarrow \psi$ in $\mathcal{R}_s$ with $Prem(\mathcal{A}) = Prem(\mathcal{A}_1) \cup \ldots \cup Prem(\mathcal{A}_2)$, $Conc(\mathcal{A}) = \psi$, $Sub(\mathcal{A}) = Sub(\mathcal{A}_1) \cup \ldots \cup Sub(\mathcal{A}_n) \cup \{\mathcal{A}\}$, $DefRules(\mathcal{A}) = DefRules(\mathcal{A}_1) \cup \ldots \cup DefRules(\mathcal{A}_n)$ and $TopRule(\mathcal{A}) = Conc(\mathcal{A}_1), \ldots, Conc(\mathcal{A}_n) \rightarrow \psi$.

3. $\mathcal{A}_1, \ldots, \mathcal{A}_n \Rightarrow \psi$ if $\mathcal{A}_1, \ldots, \mathcal{A}_n$ are arguments such that there exists a defeasible rule $Conc(\mathcal{A}_1), \ldots Conc(\mathcal{A}_n) \Rightarrow \psi$ in $\mathcal{R}_d$ with $Prem(\mathcal{A}) = Prem(\mathcal{A}_1) \cup \ldots \cup Prem(\mathcal{A}_2)$, $Conc(\mathcal{A}) = \psi$, $Sub(\mathcal{A}) = Sub(\mathcal{A}_1) \cup \ldots \cup Sub(\mathcal{A}_n) \cup \{\mathcal{A}\}$, $DefRules(\mathcal{A}) = DefRules(\mathcal{A}_1) \cup \ldots \cup DefRules(\mathcal{A}_n)$ and $TopRule(\mathcal{A}) = Conc(\mathcal{A}_1), \ldots, Conc(\mathcal{A}_n) \Rightarrow \psi$.

From [10], we write $S \vdash \varphi$ if there exists a strict argument for φ with all premises taken from S and $S \mid\!\sim \varphi$ if there exists a defeasible argument for φ with all premises taken from S.

An argument can be attacked in three ways: on a (non-axiom) premise (undermine), on a defeasible inference rule (undercut) or on a conclusion (rebuttal).

Given an argumentation system $\mathcal{AS}$ and a knowledge base $\mathcal{KB}$, an argumentation theory is $\mathcal{AT} = \langle \mathcal{AS}, \mathcal{KB}, \preceq \rangle$ where $\preceq$ is an argument ordering on the set of all arguments that can be constructed from $\mathcal{KB}$ in $\mathcal{AS}$. An argumentation theory is *well-formed* iff: if ϕ is a contrary of ψ, then ψ is not in $\mathcal{K}_n$ and is not the consequent of a strict rule.

In this paper, will use propositional logic as the language $\mathcal{L}$ and employ the following notations with respect to an argumentation system $\mathcal{AS}$: $\mathcal{K}_X(\mathcal{AS})$, with $X \in \{n, p, a, i\}$, refers to the specified subset of $\mathcal{K}$; where X is unspecified (i.e. $\mathcal{K}(\mathcal{AS})$), it is the entire knowledge base that is being referenced. $\mathcal{R}_Y(\mathcal{AS})$, with $Y \in \{s, d\}$, refers to the specified subset of $\mathcal{R}$; where Y is unspecified (i.e. $\mathcal{R}(\mathcal{AS})$), the entire rule set is being referenced; $\mathcal{A}rgs(\mathcal{AS})$ is the set of all arguments; $\mathcal{AT}_{\mathcal{AS}}$ is the argumentation theory on the basis of $\mathcal{AS}$ and $\mathcal{AF}_{\mathcal{AS}}$ is the abstract argumentation framework induced from $\mathcal{AT}_{\mathcal{AS}}$. For argument acceptability, we shall use $E_{\mathcal{AS}}$ to refer to an extension E in the framework $\mathcal{AF}_{\mathcal{AS}}$, under some unspecified single-extension semantics, subsumed by complete semantics. Finally, Π is a set of infinite length, containing all possible argumentation systems.

For a rule $\phi_1, \ldots, \phi_n \Rightarrow \setminus \rightarrow \phi$, we will employ the following notations: $Cons(r) = \phi$, the consequent and $Ant(r) = \{\phi_1, \ldots, \phi_n\}$, the antecedents.

Additionally, while it is possible for the rule, contrariness and preferences in an argumentation system to be modified in order to yield a new argumentation system, for the purposes of this work we shall focus exclusively on modifying the knowledge base. This is because the current paper is presenting the first steps towards a theory of argument revision, with a focus on the arguments themselves. Rule, contrariness and preference revision will require further significant work, which is beyond the scope of what we are presenting here.

2.2 Belief revision

Belief revision is the study of how a knowledge base can accommodate new or conflicting information. The field has been shaped by Alchourrón et. al. [1] where they proposed the AGM postulates for what they consider to be the three types of change that can be made to a knowledge base: **expansion** (denoted $K + \Phi$), where a new sentence Φ is added to a belief system, K, together with its logical consequences; **revision** (denoted $K \dot{+} \Phi$), where a new sentence that is inconsistent with K is added, but consistency is maintained by removing some sentences from K; and **contraction** (denoted $K \dot{-} \Phi$), where a sentence Φ in K is retracted without adding any new sentences. To maintain consistent closure under logical consequences, some other sentences may need to be given up.

The main principle of the AGM theory is that of *minimal change* — when selecting what, if any, information to change in a belief set, the decision is made based on what will have the smallest impact on the remaining beliefs. This is achieved through the use of *epistemic entrenchment*, where an entrenchment ordering is placed over beliefs.

3 Dialogue framework

In this section, we present a dialogue framework built on the basis of [10].

A dialogue protocol based on the ASPIC$^+$ framework was previously specified in [14]. This was defined in terms of towers of meta-argumentation systems; a meta-argumentation system being an argumentation system that describes another argumentation system, and a tower being a set of these, where the system $\mathcal{AS}_n$ describes the system $\mathcal{AS}_{n-1}$. The purpose of this is to allow for discussion and reasoning about preferences between arguments, although the main principles (for instance, the communication language) can equally be applied to a single system. However, we consider this protocol to have a significant drawback, in that its *claim* locution is defined over ASPIC$^+$ *arguments* and not formulae of the logical language. Thus if an agent possesses an argument for some conclusion ϕ, in order to claim that conclusion it must, on the basis of the definition of an argument (from [10], as described in section 2), claim its entire argument for it. This has two disadvantages; the first is that it can result in relatively trivial dialogues, where one agent claims an argument and his opponent claims another argument that defeats it — there is no scope for a continued claim-challenge-defend process, such as those found in other protocols [13]. Secondly, it limits the scope for producing different argument structures, that aren't merely a subset of the combination of the participants' argumentation systems.

This section proceeds as follows: first, we explain the concept of private and shared argumentation systems, and their role in a dialogue, before going on to provide our modified version of the communication language of [14]. Commitment and structural rules are then specified, governing, respectively, how commitments are modified, and how a dialogue progresses.

3.1 Private and shared Argumentation Systems

Each participant in the dialogue possesses a private argumentation system with a corresponding well-formed argumentation theory from which their beliefs will be derived. For a participant $\alpha \in \mathcal{P}$ (where $\mathcal{P}$ is the set of all participants), we represent this as $\mathcal{AS}_\alpha$. Beliefs are derived on the basis of some extension $E_{\mathcal{AS}}$, such that $B_\alpha = \{Conc(\mathcal{A}) \mid \mathcal{A} \in Args(\mathcal{AS}_\alpha)\}$

During a dialogue, the participants will construct a shared argumentation system (and resultant well-formed theory) based on the moves made. We label this shared argumentation system as $\mathcal{AS}_D$; similarly, the resultant argumentation theory is labelled as $\mathcal{AT}_D$. A superscript is used to refer to $\mathcal{AS}_D$ or $\mathcal{AT}_D$ at a certain time in the dialogue; for instance, $\mathcal{AS}_D$ at time $t = i$ is $\mathcal{AS}_D^i$. Ensuring the theory is well-formed places restrictions on the possible dialogue moves, which we shall leave implicit on the basis of the definition of well-formedness provided in section 2.

The purpose of the shared argumentation system is to capture the overall argument structure generated from the dialogue, and thus it is deductively monotonic, in that any formula that can be inferred in $\mathcal{AS}_D^i$ can also be inferred in $\mathcal{AS}_D^{i+1}$. This allows participants to reference formulae (and arguments) that have previously been retracted. It is possible for a formula to be removed from the knowledge base in $\mathcal{AS}_D$, but only on the condition that it is immediately replaced with a rule and antecedents that allow an argument for that formula to still be constructed.

The construction of the knowledge base and set of rules in $\mathcal{AS}_D$ is closely connected to commitment. This will be elaborated on in section 3.3 below.

3.2 Communication language

In a dialogue, the participants use a communication language to express locutions.

Definition 2. *A communication language* $\mathcal{L}_C$ *for an argumentation system* $\mathcal{AS} = \langle \mathcal{L}, ^-, \mathcal{R}, \leq \rangle$ *is:*

- $\forall \phi \in \mathcal{L}$, *claim*($\phi$), *why*($\phi$), *retract*($\phi$), *ground*($\phi$), *concede*($\phi$), *resolve*($\phi$) $\in \mathcal{L}_C$
- $\forall \Phi \in 2^{\mathcal{L}}$, $\forall r \in \mathcal{R}$, $justify(\langle \Phi, r \rangle)$, *accept*($\langle \Phi, r \rangle$) $\in \mathcal{L}_C$

The two key differences between our language and that of [14] are the format of the *claim* locution, and the inclusion of $justify$, *ground* and *accept*.

Our *claim* is defined over formulae of $\mathcal{L}$, which allows a participant to claim components of an argument, as opposed to an entire argument itself. The $justify$ locution allows a participant to provide support for a claim, again without having to present their entire argument for it, with the *accept* locution allowing another participant to accept a justification. In situations where the claim has no support (i.e. it is an atomic argument), the participant will use the locution *ground*.

Given the communication language (definition 2) and a set of participants $\mathcal{P}$, we can now define a dialogue move.

Definition 3. *A dialogue move is a tuple,* $m = \langle id, pl, loc, t \rangle$ *where:*

- $id \in \mathbb{N}$ *the ID of the move*
- $pl \in \mathcal{P}$, *the participant*
- $loc \in \mathcal{L}_C$, *the locution, including its content*

– $t \in \mathbb{N}$, *the target of the move*

For notational convenience, we provide each element of a dialogue move with a corresponding function that, for a move m_i, returns the value of that element in the move: $id(m_i)$, $pl(m_i)$, $loc(m_i)$ and $t(m_i)$.

A dialogue consists of a topic, a set of participants and a set of moves.

Definition 4. *A dialogue is a tuple $D = \langle \tau, \mathcal{P}, \mathcal{M} \rangle$ where:*

– $\tau \in \mathcal{L}$, *the topic of the dialogue*
– $\mathcal{P}$ *is the set of participants in the dialogue*
– $\mathcal{M} = \{m_1, \ldots, m_n\}$ *is a set of moves, where $loc(m_1) = claim(\tau)$ and* $\mid \mathcal{M} \mid = \mid \bigcup_{i=1}^{n} loc(m_i) \mid$

3.3 Commitment rules

We now describe how a participant incurs commitment in our dialogue framework.

A commitment store is a representation of everything to which a participant has become committed during a dialogue. In our framework, commitment stores are represented as sets of formulae of the language $\mathcal{L}$ used by the participants in their private argumentation systems, with commitment being incurred through claims, concessions, justifications and acceptance. Commitments are removed through retractions.

The following rules describe the commitment store C^i_α of a participant $\alpha \in \mathcal{P}$ at time $t = i$ in the dialogue (with for all participants $C^0 = \emptyset$), following a locution at move m_i. The commitment store of a participant who does not make a move at $t = i$ is the identical to their commitment store at $t = i-1$; that is, $\forall p' \in \mathcal{P} \backslash \{pl(m_i)\}, \mathcal{C}^i_{p'} = \mathcal{C}^{i-1}_{p'}$. Also, unless otherwise stated, the shared argumentation system remains unchanged (i.e. $\mathcal{AS}^i_D = \mathcal{AS}^{i+1}_D$). Where no explicit commitment rule is provided for a locution, the commitment stores of all participants remain unchanged.

Our first commitment rule states that if a participant claims a formula, the participant is committed to that formula. The formula is also placed in the knowledge base of $\mathcal{AS}_D$ as an ordinary premise. This is because the formula has been stated (i.e. it is not assumed), but is not axiomatic.

C1 if $loc(m_i) = claim_\alpha(\phi)$ then:
– $\mathcal{C}^i_\alpha = \mathcal{C}^{i-1}_\alpha \cup \{\phi\}$
– $\mathcal{K}_p(\mathcal{AS}^i_D) = \mathcal{K}_p(\mathcal{AS}^{i-1}_D) \cup \{\phi\}$

Conversely, if a participant retracts a formula, they are no longer committed to it.

C2 if $loc(m_i) = retract_\alpha(\phi)$, then $\mathcal{C}^i_\alpha = \mathcal{C}^{i-1}_\alpha \backslash \{\phi\}$

If a participant concedes a formula to their opponent, they become committed to that formula.

C3 if $loc(m_i) = concede_\alpha(\phi)$, then $\mathcal{C}^i_\alpha = \mathcal{C}^{i-1}_\alpha \cup \{\phi\}$

The fourth commitment rule describes the commitments incurred when a participant justifies a previously claimed formula; the participant becomes committed to the consequent and antecedents of the rule in the justification. Additionally, all formulae in Φ are added as ordinary premises in $\mathcal{K}(\mathcal{AS}_D)$, while all unstated antecedents of the rule (i.e. those antecedents that do not appear in Φ) are added as assumptions (but could subsequently be made explicit through claims or justifications) . The consequent of the rule is removed from $\mathcal{K}_p(\mathcal{AS}_D)$, because it is no longer a premise — it is the conclusion of a non-atomic argument.

C4 if $loc(m_i) = justify_\alpha(\langle\Phi, r\rangle)$, then:
- $\mathcal{C}^i_\alpha = \mathcal{C}^{i-1}_\alpha \cup Ant(r) \cup \{Cons(r)\}$
- $\mathcal{K}_p(\mathcal{AS}^i_D) = (\mathcal{K}(\mathcal{AS}^{i-1}_D) \backslash \{Cons(r)\}) \cup \Phi$
- $\mathcal{K}_a(\mathcal{AS}^i_D) = \mathcal{K}_a(\mathcal{AS}^{i-1}_D) \cup (Ant(r) \backslash \Phi)$
- $\mathcal{R}(\mathcal{AS}^i_D) = \mathcal{R}(\mathcal{AS}^{i-1}_D) \cup \{r\}$

The acceptance of a justification has a similar effect on the accepter's commitments as the justification does on the justifier's.

C5 if $loc(m_i) = accept_\alpha(\langle\Phi, r\rangle)$, then $\mathcal{C}^i_\alpha = \mathcal{C}^{i-1}_\alpha \cup Ant(r) \cup Cons(r)$

3.4 Structural rules

The structural rules in a dialogue dictate how a dialogue progresses, by restricting what locutions can be made after other locutions, and what their content is. For simplicity, our framework will assume only two participants, $\mathcal{P} = \{\alpha, \beta\}$. Turntaking is fully-specified and implicit in the structure of each rule.

Our first structural rule describes what must follow a *claim* locution. Following a claim by one participants, the second participants must either question it, claim a contrary or contradictory formula, or concede the claimed formula.

R1 if $loc(m_i) = claim_\alpha(\phi)$, then $loc(m_{i+1})$ must be either:
- $why_\beta(\phi)$
- $claim_\beta(\psi)$ only if $\psi \in \overline{\phi}$
- $concede_\beta(\phi)$

Our second structural rule dictates that when a claim is questioned, it must either be justified or retracted.

R2 if $loc(m_i) = why_\beta(\phi)$, then $loc(m_{i+1})$ must be either:
- $justify_\alpha(\langle\Phi, r\rangle)$ only if $Cons(r) = \phi$ an $\Phi \subseteq Ant(r)$
- $ground_\alpha(\phi)$ only if $\phi \in \mathcal{K}(\mathcal{AS}_\alpha)$
- $retract_\alpha(\phi)$

The third structural rule shows a reply to a $justify$ locution — either a claim of a contrary or undercutting formula, or an acceptance of the justification.

R3 if $loc(m_i) = justify_\beta(\langle \Phi, r \rangle)$, then $loc(m_{i+1})$ must be either:
- $claim_\alpha(\psi)$ only if $\psi \in \overline{Cons(r)}$ or $\exists \phi \in Ant(r)$ s.t. $\psi \in \overline{\phi}$
- $claim_\alpha(\psi)$ only if $\psi \in \overline{\mathcal{A}}$, with $r \in DefRules(\mathcal{A})$
- $accept_\beta(\langle \Phi, r \rangle)$

The fourth rule describes the response to a *ground* location; this is similar to the response to a claim, except *why* cannot be used.

R4 if $loc(m_i) = ground_\alpha(\phi)$, then $loc(m_{i+1})$ must be either:
- $claim_\beta(\psi)$ only if $\psi \in \overline{\phi}$
- $concede_\beta(\phi)$

The fifth rule describes the response to a participant being forced to resolve its commitment store.

R5 if $loc(m_i) = resolve_\alpha(\phi)$ then $loc(m_{i+1})$ must be $retract_\beta(\phi)$ if $\phi \in \mathcal{C}^i_\beta$

Following a retraction, a participant may immediately either retract another formula, or make a $justify$ locution; the latter allows the participant to present an undercutter to the argument whose conclusion they have retracted. Otherwise, the participant may make any move, provided the dialogue has not terminated.

R6 if $loc(m_i) = retract_\beta(\phi)$ then $loc(m_{i+1})$ must be either:
- $retract_\beta(\psi)$ only if $\psi \in \mathcal{C}^i_\beta$ and ψ is a premise in an argument with conclusion ϕ
- $justify_\beta(\langle \Phi, r \rangle)$ only if $Cons(r) = \overline{\mathcal{A}}$ with $Conc(\mathcal{A}) = \phi$ and $\Phi \subseteq Cons(r)$
- any other move by β, except in the conditions of **R8**

When a participant concedes or accepts, its opponent can either force it to resolve an inconsistency in its commitment store, or make any other move provided the dialogue has not terminated.

R7 if $loc(m_i) = concede_\alpha(\phi)$ or $loc(m_i) = accept_\alpha(\langle \Phi, r \rangle)$ (with $Cons(r) = \phi$) then $loc(m_{i+1})$ must be either:
- $resolve_\beta(\phi)$ only if C^i_α is inconsistent w.r.t. ϕ
- any other move by β, except in the conditions of **R8**

The final rule describes the termination conditions — either a concession or acceptance of the topic, τ, by the opponent (β), or a retraction by the proponent (α).

R8 a dialogue terminates at move m_i if either:
- $loc(m_i) = concede_\beta(\tau)$
- $loc(m_i) = accept_\beta(\langle \Phi, r \rangle)$ only if $Conc(r) = \tau$
- $loc(m_i) = retract_\alpha(\tau)$.

4 Revising an argumentation system

If for a participant $p \in \mathcal{P}$, at time $t = i$ in a dialogue, C_p^i is inconsistent, p can be forced by its opponent to update its commitments in order to resolve the inconsistency. A commitment store in a dialogue based on ASPIC$^+$ is inconsistent if it contains two formulae that are either contradictory, or one is the contrary of the other.

Definition 5. *The commitment store C_p^i of participant $p \in \mathcal{P}$ at time $t = i$ in a dialogue is inconsistent if $\exists \phi, \psi \in C_p^i$ s.t. $\phi \in \overline{\psi}$ and/or $\psi \in \overline{\phi}$ in $\mathcal{AS}_p$.*

Resolving an inconsistency will take the form of a participant removing a formula or formulae from its commitment store through a retraction move. Simply removing the source of the inconsistency may not, however, be sufficient, because if that source is the conclusion of an argument that has been rebutted, the commitment store will still contain the premises used to construct that argument — that is, for some non-atomic argument $\mathcal{A}$ with $Conc(\mathcal{A}) = \phi$, $C_p \backslash \{\phi\} \mid\!\sim \phi$. Thus, the participant will also need to either retract a commitment that is a premise of $\mathcal{A}$, or incur a commitment that brings about an exception to a defeasible rule in $\mathcal{A}$. Given a choice of retractions or additions, it will be desirable for the participant to minimise the effect on other commitments and future claims or justifications it can advance in the dialogue

Assessing the impact of retracting and/or incurring commitments can be done through the participant examining its private argumentation system. While the participant may not be explicitly revising its beliefs, techniques used in a revision process can identify the effects of rendering a belief incommunicable in a dialogue. In order to describe these effects, we define two functions — **formula removal** and **formula expansion**. While expansion performs a similar role to its belief revision counterpart, removal is not defined in the original AGM theory. We shall provide a redefinition of contraction later in this section.

The *formula removal function* governs the removal of a formula from the knowledge base of an argumentation system. The knowledge base of the resultant argumentation system is the same as that of the input system, less the removed formula.

Definition 6. ***Formula removal function***
$\mathcal{AS} - \phi$: $\mathcal{K}(\mathcal{AS} - \phi) = \mathcal{K}(\mathcal{AS}) \backslash \{\phi\}$

The *formula expansion function* governs the addition of a formula to the knowledge base of an argumentation system. When a formula is added, it is added to the set of assumptions, because arbitrarily adding information to the knowledge base, without justification, only assumes that it is true.

Definition 7. ***Formula expansion function***
$\mathcal{AS} + \phi$: $\mathcal{K}_a(\mathcal{AS} + \phi) = \mathcal{K}_a(\mathcal{AS}) \cup \{\phi\}$

Remark 2. *Note also that because $\mathcal{K}_a(\mathcal{AS}) \subseteq \mathcal{K}(\mathcal{AS})$, $\mathcal{K}(\mathcal{AS} + \phi) = \mathcal{K}(\mathcal{AS}) \cup \{\phi\}$*

We can demonstrate several properties of these operators. Firstly, a removal will always yield a well-formed argumentation system if the input system is well-formed.

Proposition 1. *If $\mathcal{AT}_{\mathcal{AS}}$ is well-formed, $\mathcal{AT}_{\mathcal{AS}-\phi}$, is well-formed*

Proof. *Since $\mathcal{R}(\mathcal{AS}) = \mathcal{R}(\mathcal{AS} - \phi)$, we consider only the knowledge base in $\mathcal{AS} - \phi$. If $\mathcal{AT}_{\mathcal{AS}}$ is well-formed, then $\mathcal{K}(\mathcal{AS})$ satisfies [10, Definition 6.8]. Since $\mathcal{K}(\mathcal{AS} - \phi) \subseteq \mathcal{K}(\mathcal{AS})$, $\mathcal{K}(\mathcal{AS} - \phi)$ also satisfies the definition.* □

We can also prove that the formula expansion function is deductively monotonic (that is, no arguments, acceptable or otherwise, are lost in an expansion).

Proposition 2. *The formula expansion function, $+$, is deductively monotonic.*

Proof. *$\forall \mathcal{A} \in Args(\mathcal{AS})$, $Prem(\mathcal{A}) \subseteq \mathcal{K}(\mathcal{AS})$. For some $\phi \in \mathcal{L}$, $\mathcal{K}(\mathcal{AS}) \subseteq \mathcal{K}(\mathcal{AS} + \phi)$. Thus, $\forall \mathcal{A} \in Args(\mathcal{AS})$, $Prem(\mathcal{A}) \subseteq \mathcal{K}(\mathcal{AS} + \phi)$.*

There is no change to an argumentation system if the input formula to an expansion is already an assumption in the knowledge base, or if the input formula to a removal is not in any subset of the knowledge base:

Proposition 3. *If for $\mathcal{AS} + \phi$, $\phi \in \mathcal{K}_a(\mathcal{AS})$, $\mathcal{AS} + \phi = \mathcal{AS}$*

Proof. *$\mathcal{K}_a(\mathcal{AS}+\phi) = \mathcal{K}_a(\mathcal{AS}) \cup \{\phi\}$. Since $\phi \in \mathcal{K}_a(\mathcal{AS})$, $K_a(\mathcal{AS}+\phi) = \mathcal{K}_a(\mathcal{AS})$.*

Proposition 4. *If for $\mathcal{AS} - \phi$, $\phi \notin \mathcal{K}(\mathcal{AS})$, $\mathcal{AS} - \phi = \mathcal{AS}$.*

Proof. *$\mathcal{K}(\mathcal{AS} - \phi) = \mathcal{K}(\mathcal{AS}) \backslash \{\phi\}$. Since $\phi \notin \mathcal{K}(\mathcal{AS})$, $K(\mathcal{AS} - \phi) = \mathcal{K}(\mathcal{AS})$.*

It is also the case that a removal can only be "undone" [6] (through expansion) if the input formula was an assumption in the original knowledge base.

Proposition 5. *$\mathcal{AS} - \phi + \phi = \mathcal{AS}$ iff $\phi \in \mathcal{K}_a(\mathcal{AS})$*

Proof.

- *If $\phi \notin \mathcal{K}_a(\mathcal{AS})$, $\mathcal{AS} - \phi = \mathcal{AS}$. However, $\phi \in \mathcal{K}_a(\mathcal{AS} + \phi)$, hence $\mathcal{AS} + \phi \neq \mathcal{AS}$.*
- *If $\phi \in \mathcal{K}_a(\mathcal{AS})$, $\mathcal{K}(\mathcal{AS} - \phi) = \mathcal{K}_a(\mathcal{AS}) \backslash \{\phi\}$. $\mathcal{K}_a((\mathcal{AS} - \phi) + \phi) = \mathcal{K}_a(\mathcal{AS} - \phi) \cup \{\phi\} = (\mathcal{K}_a(\mathcal{AS}) \backslash \{\phi\}) \cup \{\phi\} = \mathcal{K}_a(\mathcal{AS})$*

Similarly, an expansion can only be "undone" (through removal) if the input formula was not already in any subset of the original knowledge base.

Proposition 6. *$\mathcal{AS} + \phi - \phi = \mathcal{AS}$ iff $\phi \notin \mathcal{K}(\mathcal{AS})$*

Proof.
- *If $\phi \in \mathcal{K}_a(\mathcal{AS})$, $\mathcal{AS} + \phi = \mathcal{AS}$. However, $\mathcal{AS} - \phi \neq \mathcal{AS}$.*
- *If $\phi \notin \mathcal{K}(\mathcal{AS})$, $\mathcal{K}(\mathcal{AS} + \phi) = \mathcal{K}(\mathcal{AS}) \cup \{\phi\}$. $\mathcal{K}((\mathcal{AS} + \phi) - \phi) = \mathcal{K}(\mathcal{AS} + \phi) \backslash \{\phi\} = (\mathcal{K}(\mathcal{AS}) \cup \{\phi\}) \backslash \{\phi\} = \mathcal{K}(\mathcal{AS})$*

A removal or expansion is required to "undo" an expansion or removal because, in contrast to [9], we are also considering the loss and gain of arguments themselves and not just their acceptability.

In section 4.1, we describe some further properties of these operators, in terms of measuring minimal change.

Both the formula removal and formula expansion functions yield new argumentation systems. However, it's possible that these resultant systems present the need for further modifications (for instance, if adding a formula causes the argumentation theory derived from the system to no longer be well-formed) and, as with the initial retraction, there may be multiple possible modifications, which again raises the question of which to choose. To model these possible modifications, we use a structure called a *change graph*.

In order to define a change graph, we first define a *change path*:

Definition 8. *A change path* $CP(\mathcal{AS}, \mathcal{AS}^-) = \langle \Upsilon, \Omega \rangle$ *from* $\mathcal{AS}$ *to* $\mathcal{AS}^-$ *is a rooted tree where:*

1. $\Upsilon \subseteq \Pi$
2. $\Omega \subseteq \Upsilon \times \Upsilon$ *where* $\forall \omega \in \Omega$ *such that* $\omega = (\mathcal{AS}', \mathcal{AS}'')$, $\mid \mathcal{K}(\mathcal{AS}') \mid - \mid \mathcal{K}(\mathcal{AS}'') \mid = \pm 1$ *(atomic change).*
3. *there exists exactly one* $\omega \in \Omega$ *whose predecessor is the root* ($\mathcal{AS}$) *and exactly one* $\omega' \in \Omega$ *whose successor is a leaf* ($\mathcal{AS}^-$)*.*
4. $CP(\mathcal{AS}, \mathcal{AS}^-) = \langle \Upsilon, \Omega \rangle$ *is edge minimal in that* $\nexists CP(\mathcal{AS}, \mathcal{AS}^-) = \langle \Upsilon', \Omega' \rangle$ *s.t.* $\bigcup_{\omega' \in \Omega'} mod(\omega') \subset \bigcup_{\omega \in \Omega} mod(\omega)$*, where* $mod(\omega)$ *is the input formula to the function (removal or expansion) performed on the edge.*

Remark 3. $CP_A(\mathcal{AS}, \mathcal{AS}^-) = \{\langle \Upsilon_i, \Omega_i \rangle \mid \langle \Upsilon_i, \Omega_i \rangle$ *is a change path from* $\mathcal{AS}$ *to* $\mathcal{AS}^-\}$*, the set of all change paths from* $\mathcal{AS}$ *to* $\mathcal{AS}^-$

Given a set of change paths, we can now define a change graph:

Definition 9. *A change graph* $CG(\mathcal{AS}, \mathcal{AS}^-)$ *for a change from* $\mathcal{AS}$ *to* $\mathcal{AS}^-$ *is a tuple* $\langle \Upsilon_G, \Omega_G \rangle$ *where given* $CP_A(\mathcal{AS}, \mathcal{AS}^-) = \{\langle \Upsilon_1, \Omega_1 \rangle, \ldots, \langle \Upsilon_n, \Omega_n \rangle\}$, $\Upsilon_G = \bigcup_{i=1}^{n} \Upsilon_i$ *and* $\Omega_G = \bigcup_{i=1}^{n} \Omega_i$

If $\phi \notin \{Conc(\mathcal{A}) \mid \mathcal{A} \in E_{\mathcal{AS}}\}$ and $\mathcal{AT}_{\mathcal{AS}^-}$ is well-formed, $\mathcal{AS}^-$ is a *contraction* of $\mathcal{AS}$ by ϕ; we denote this using the **argument contraction operator**: $\mathcal{AS} \dot{-} \phi$. A change path from $\mathcal{AS}$ to $\mathcal{AS} \dot{-} \phi$ defines a procedure for revising $\mathcal{AS}$ into $\mathcal{AS} \dot{-} \phi$; a change graph captures all edge-minimal procedures for the revision. Note that $\mathcal{AS} \dot{-} \phi$ (for any $\phi \in \mathcal{L}$) is not unique; it describes *any* argumentation system derived from $\mathcal{AS}$, which contains no acceptable argument for ϕ and whose argumentation theory is well-formed.

A change graph shows the ways in which $\mathcal{AS}$ *can* be revised into $\mathcal{AS} \dot{-} \phi$; it does not, however, answer the question of which way should be chosen, with respect to minimal change, which is what we shall now go on to explore.

4.1 Measuring minimal change

Modifying an argumentation system in order to eliminate all acceptable arguments for a certain conclusion takes one of three forms. Either the arguments for the conclusion can be completely removed, through removing premises; they can be defeated by making existing defeaters acceptable, or introducing completely new defeaters; or a combination of both.

Both removing arguments and modifying acceptability can be achieved by making changes to the knowledge base — to remove an argument would involve removing elements from the knowledge base, while adding an argument would involve adding elements. Making acceptable an existing (unacceptable) defeater could be achieved through either of these. In the same way that modifying a belief set (through addition or removal) can have an impact on other beliefs, doing the same to a knowledge base in an argumentation system can have an impact on other arguments, aside from the one being removed or added. This impact, however, is not solely structural when using the ASPIC$^+$ framework; being built on Dung's abstract theory, the acceptability of arguments in the framework is evaluated using various sceptical and credulous semantics. In broad terms, an argument is acceptable if it is not defeated by other acceptable arguments, and not acceptable if it is. Arguments can defend other arguments by defeating defeaters (for instance, an argument A defends an argument C if A defeats B, which in turn defeats C).

Thus we must consider at least four effects when modifying a knowledge base — **argument loss**: those previously acceptable arguments that can no longer be constructed in the system; **acceptability loss**: those arguments that remain in the system, but have become unacceptable; **argument gain**: new arguments in the system; and **acceptability gain**: those arguments that remain in the system and have gained acceptability.

These effects can be captured by defining functions that take as their input a formula, and output a set of arguments that are affected. In the following definitions, Υ_G is the set of all argumentation systems in a change graph G, with $\mathcal{A}rgs(\Upsilon_G)$ being the set of all arguments in all argumentation systems in G. We will leave the type of change (expansion or removal) unspecified, however, in subsequent sections, we will where necessary make explicit the type of change with a superscript $+$ (for expansion) or $-$ (for removal).

In a dialogical context, the effects should not be measured by considering only other commitments, but also in terms of beliefs that are still private. This is because retracting a commitment will not just impact on existing commitments, but also future potential commitments; for instance, if an agent retracts a formula ϕ, any arguments in which ϕ is a premise are now no longer available to it. Thus when the following functions are applied to dialogue, they measure the effects of a revision with respect to the impact on an agent's private argumentation system and, hence, its beliefs as opposed to only its commitments. They do, however, still capture the effects on commitments, because its commitment store can contain beliefs.

In the following definitions, $E_{\mathcal{AS}}$ refers to an extension E under some unspecified single-extension semantics, subsumed by complete semantics.

The first function, the *argument drop function* identifies those acceptable arguments that are completely lost when removing or adding a formula.

Definition 10. *The* argument drop function Δ_A*:*

$\Delta_A \colon \mathcal{L} \times \Upsilon_G \to 2^{\mathcal{A}rgs(\Upsilon_G)}$,
$\Delta_A(\phi, \mathcal{AS}) = \{\mathcal{A} \mid \mathcal{A} \in E_{\mathcal{AS}},\ \mathcal{A} \notin \mathcal{A}rgs(\mathcal{AS} \pm \phi)\}$

When expanding an argumentation system, the argument drop function always yields an empty set.

Proposition 7. $\forall \phi \in \mathcal{L},\ \Delta_A^+(\phi, \mathcal{AS}) = \emptyset$

Proof. From proposition 2, + is deductively monotonic. Thus no arguments can be dropped.

The second function, the *acceptability drop function*, identifies those arguments that lose acceptability when removing or adding a formula.

Definition 11. *The* acceptability drop function Δ_S*:*

$\Delta_S \colon \mathcal{L} \times \Upsilon_G \to 2^{\mathcal{A}rgs(\Upsilon_G)}$,
$\Delta_S(\phi, \mathcal{AS}) = \{\mathcal{A} \mid \mathcal{A} \in E_{\mathcal{AS}},\ \mathcal{A} \notin E(\mathcal{AS} \pm \phi,\ \mathcal{A} \in \mathcal{A}rgs(\mathcal{AS} \pm \phi)\}$

The third function, the *argument gain function*, identifies those arguments that the system gains when removing or adding a formula.

Definition 12. *The* argument gain function Γ_A*:*

$\Gamma_A \colon \mathcal{L} \times \Upsilon_G \to 2^{\mathcal{A}rgs(\Upsilon_G)}$,
$\Gamma_A(\phi, \mathcal{AS}) = \{\mathcal{A} \mid \mathcal{A} \notin \mathcal{A}rgs(\mathcal{AS}),\ \mathcal{A} \in \mathcal{A}rgs(\mathcal{AS} \pm \phi)\}$

The argument gain function always yields an empty set in a removal.

Proposition 8. $\forall \phi \in \mathcal{L},\ \Gamma_A^-(\phi, \mathcal{AS}) = \emptyset$.

Proof. Consider the opposite: $\mathcal{A} \notin \mathcal{A}rgs(\mathcal{AS}), \mathcal{A} \in \mathcal{A}rgs(\mathcal{AS}) - \phi$*. Thus,* $Prem(\mathcal{A}) \not\subseteq \mathcal{K}(\mathcal{AS})$ *and* $Prem(\mathcal{A}) \subseteq \mathcal{K}(\mathcal{AS} - \phi)$*. However,* $\mathcal{KAS} - \phi) = \mathcal{K}(\mathcal{AS}) \backslash \{\phi\}$*, hence* $\mathcal{K}(\mathcal{A} - \phi) \subseteq \mathcal{K}(\mathcal{AS})$*. Contradiction!*

The final function, the *acceptability gain function*, identifies those arguments that were unacceptable in the input argumentation system and have remained in the output system, but have lost acceptability.

Definition 13. *The* acceptability gain function Γ_S*:*

$\Gamma_S \colon \mathcal{L} \times \Upsilon_G \to 2^{\mathcal{A}rgs(\Upsilon_G)}$,
$\Gamma_S(\phi, \mathcal{AS}) = \{\mathcal{A} \mid \mathcal{A} \notin E_{\mathcal{AS}},\ \mathcal{A} \in E_{\mathcal{AS} \pm \phi}\}$

The drop and gain functions for each type of change (i.e. argument and acceptability) are linked to each other, in that no arguments (resp. acceptability) that are dropped are also gained, and vice versa.

Proposition 9. *For* $X \in \{A, S\}$, $\Delta_X(\phi, \mathcal{AS}) \cap \Gamma_X(\phi, \mathcal{AS}) = \emptyset$

Proof. Consider the opposite in terms of argument drop and gain: $\mathcal{A} \in \Delta_A(\phi, \mathcal{AS})$, $\mathcal{A} \in \Gamma_A(\phi, \mathcal{AS})$. *From definitions 10 and 12,* $\mathcal{A} \notin Args(\mathcal{AS} \pm \phi)$ *and* $\mathcal{A} \in Args(\mathcal{AS} \pm \phi)$ *respectively. Contradiction!*
Consider the opposite in terms of acceptability drop and gain: $\mathcal{A} \in \Delta_S(\phi, \mathcal{AS})$, $\mathcal{A} \in \Gamma_S(\phi, \mathcal{AS})$. *From definitions 11 and 13,* $\mathcal{A} \notin E_{\mathcal{AS}\pm\phi}$, $\mathcal{A} \in E_{\mathcal{AS}\pm\phi}$. *Contradiction!*

Propositions 8 and 9 work together in ensuring that any argumentation system can be revised with respect to any argument in that system, because we will always be able to remove propositions (and hence arguments) with no gains.

Having now specified four functions for *measuring* minimal change, we now show how to apply them to change graphs in order to *determine* minimal change.

4.2 Determining minimal change

To use a change graph to determine minimal change, we first assign costs to the graph's edges, based on outputs of the functions defined in section 4.1. We do this through the *edge cost function*, which takes as input an edge in a change graph and outputs a vector with the drop and gain functions as its components.

Definition 14. *The edge cost function* V *for a change graph* $CG(\mathcal{AS}, \mathcal{AS}^-) = \langle \Upsilon_G, \Omega_G \rangle$ *where* $Args(\Upsilon_G) = \bigcup_{\mathcal{AS} \in \Upsilon_G} Args(\mathcal{AS})$ *:*

$V \colon \Omega \to (2^{Args(\Upsilon_G)})^4$

$$V((\mathcal{AS}', \mathcal{AS}' \pm \phi)) = \begin{pmatrix} \Delta_S(\phi, \mathcal{AS}') \\ \Delta_A(\phi, \mathcal{AS}') \\ \Gamma_S(\phi, \mathcal{AS}') \\ \Gamma_A(\phi, \mathcal{AS}') \end{pmatrix}$$

Keeping the measures separate is important for computing the overall cost of a change path. Given a change path $CP(\mathcal{AS}, \mathcal{AS}^-)$ in a change graph, we can compute an overall change vector for that path based on the edge cost functions. We do this by defining a new operator $\uplus$:

$$\biguplus_{i=1}^{n} \begin{pmatrix} \Delta_S(\phi_i, \mathcal{AS}_i) \\ \Delta_A(\phi_i, \mathcal{AS}_i) \\ \Gamma_S(\phi_i, \mathcal{AS}_i) \\ \Gamma_A(\phi_i, \mathcal{AS}_i) \end{pmatrix} = \begin{pmatrix} \bigcup_{i=1}^{n} (\Delta_S(\phi_i, \mathcal{AS}_i) \setminus \bigcup_{j=i}^{n} \Gamma_S(\phi_j, \mathcal{AS}_j)) \\ \bigcup_{i=1}^{n} (\Delta_A(\phi_i, \mathcal{AS}_i) \setminus \bigcup_{j=i}^{n} \Gamma_A(\phi_j, \mathcal{AS}_j)) \\ \bigcup_{i=1}^{n} (\Gamma_S(\phi_i, \mathcal{AS}_i) \setminus \bigcup_{j=i}^{n} \Delta_S(\phi_j, \mathcal{AS}_j)) \\ \bigcup_{i=1}^{n} (\Gamma_A(\phi_i, \mathcal{AS}_i) \setminus \bigcup_{j=i}^{n} \Delta_A(\phi_j, \mathcal{AS}_j)) \end{pmatrix}$$

The first (resp. third) component captures the overall acceptability drops (resp. gains) in the path, while excluding those arguments that are re-instated (resp. dropped), and remain reinstated (resp. dropped). To do this, we consider the output of the i^{th} acceptability drop (resp. gain) function and remove from it any arguments that are subsequently reinstated (resp, dropped), which is the union of all acceptability gain (resp. drop) functions from the i^{th} onwards.

Similarly, the second (resp. fourth) component captures the overall argument drops (resp. gains) in the path, while exclusing those arguments that are regained (resp. dropped), and remain in the path (resp. dropped). This is done in the same way as for the first and third components, except it is with respect to arguments themselves and not acceptability.

Thus, the overall path cost for a path $CP(\mathcal{AS}, \mathcal{AS}^-) = \langle \Upsilon, \Omega \rangle$ with $\Omega = \{\omega_1, \ldots, \omega_n\}$ is $\biguplus_{i=1}^{n} V(\omega_i)$, where V is the edge cost function (Definition 14).

Once a cost has been established for a path, a numeric value is obtained by considering the cardinality of the union of the components in the cost vector. These values are then compared to other path cost values to to decide which path or paths represent the minimal change.

5 Example

In this section, we provide an example that illustrates the internal reasoning process that assists an agent in selecting what commitment(s) to retract, and possibly incur, to resolve an inconsistency in its commitment store.

Two agents $\mathcal{P} = \{\alpha, \beta\}$ are engaged in a dialogue, with the following shared rules, rule preferences and contrariness:

$$\mathcal{R}_d = \left\{ \begin{array}{ll} r1 : a \Rightarrow b, & r2 : a \Rightarrow c, \\ r3 : d \Rightarrow e, & r4 : f \Rightarrow g, \\ r5 : i \Rightarrow j, & r6 : j \Rightarrow k, \\ r7 : h \Rightarrow \neg r1 & \end{array} \right\}$$

$r3 < r2$ $\qquad$ $c \in \overline{e}$, $e \in \overline{f}$, $h \in \overline{i}$, $l \in \overline{b}$, $m \in \overline{n}$ and $b \in \overline{m}$

For the purposes of this example, it will be α that carries out an argument revision process. We will therefore provide a full knowledge base for α and only a minimal knowledge base for β:

$\mathcal{K}(\mathcal{AS}_\alpha) = \{a, d, f, i, n\}$ $\qquad$ $\mathcal{K}(\mathcal{AS}_\beta) = \{l, m\}$

α can construct eleven arguments:

$\mathcal{A}_1 : a$ $\qquad$ $\mathcal{A}_2 : d$
$\mathcal{A}_3 : f$ $\qquad$ $\mathcal{A}_4 : i$
$\mathcal{A}_5 : n$ $\qquad$ $\mathcal{A}_6 : \mathcal{A}_1 \Rightarrow b$
$\mathcal{A}_7 : \mathcal{A}_1 \Rightarrow c$ $\qquad$ $\mathcal{A}_8 : \mathcal{A}_2 \Rightarrow e$
$\mathcal{A}_9 : \mathcal{A}_3 \Rightarrow g$ $\qquad$ $\mathcal{A}_{10} : \mathcal{A}_4 \Rightarrow j$
$\mathcal{A}_{11} : \mathcal{A}_{10} \Rightarrow k$

β can construct two arguments, $\mathcal{B}_1 : l$ and $\mathcal{B}_2 : m$.

The grounded extensions of each agent's private argumentation systems, applying the weakest link principle [10], are:

- $GE_{\mathcal{AS}_\alpha} = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \mathcal{A}_5, \mathcal{A}_6, A_7, \mathcal{A}_9, \mathcal{A}_{10}, \mathcal{A}_{11}\}$
- $GE_{\mathcal{AS}_\beta} = \{\mathcal{B}_1, \mathcal{B}_2\}$

From the grounded extensions, we derive the agents' beliefs:

$B_\alpha = \{a, b, c, d, f, g, i, j, k, n\}$ $\qquad$ $B_\beta = \{l, m\}$

Consider the dialogue fragment below. For the sake of brevity, we shall only include a *why* and *justify* related to the move $claim(b)$, because b will be the formula with which we illustrate the argument revision techniques.

id (t)	participant	Locution	Target	$C_{participant}$
1	α	$claim(n)$	$\emptyset$	$\{n\}$
2	β	$claim(m)$	$\{1\}$	$\{m\}$
3	α	$claim(b)$	$\{2\}$	$\{b\}$
4	β	$why(b)$	$\{3\}$	$\{m\}$
5	α	$justify(\langle\{a\}, a \Rightarrow b\rangle)$	$\{4\}$	$\{a, b\}$
6	β	$claim(l)$	$\{5\}$	$\{l, m\}$
7	α	$concede(l)$	$\{6\}$	$\{a, b, l\}$
8	β	$resolve(b)$	$\{7\}$	$\{l, m\}$

In the dialogue, α has claimed b and then, when challenged by β, has justified it with a, placing both a and b into α's commitment store. β then claims l, which is a contrary of b and so defeats the argument. α, having no defence against l, concedes it. β then forces α to resolve his commitment store with respect to b. To perform this resolution, α must not only remove b, but also either a or claim h, which introduces an exception to rule $a \Rightarrow b$. To choose which, α performs an argument revision process in their private argumentation system.

Consider the outputs of the four functions for a and h, with respect to $\mathcal{AS}_\alpha$.

ϕ	$\Delta_A(\phi, \mathcal{AS}_\alpha)$	$\Delta_S(\phi, \mathcal{AS}_\alpha)$	$\Gamma_A(\phi, \mathcal{AS}_\alpha)$	$\Gamma_S(\phi, \mathcal{AS}_\alpha)$
$-a$	$\{\mathcal{A}_1, \mathcal{A}_6, \mathcal{A}_7\}$	$\{\mathcal{A}_3, \mathcal{A}_9\}$	$\{\}$	$\{\mathcal{A}_8\}$
$+h$	$\{\}$	$\{\mathcal{A}_6\}$	$\{\mathcal{A}_{12}\}$	$\{\}$

With $A_{12} : h$.

Adding h, however, generates argumentation system $\mathcal{AS} + h$ whose argumentation theory ($\mathcal{AT}_{\mathcal{AS}+h}$) is not well-formed, because $h \in \mathcal{K}_a(\mathcal{AS} + h)$ and $h \in \bar{i}$, but $i \in \mathcal{K}_p(\mathcal{AS} + h)$. α must, therefore, perform another step in the process, which will be to assess the removal of i from $\mathcal{AS} + h$:

ϕ	$\Delta_A^-(\phi, \mathcal{AS}_\alpha)$	$\Delta_S^-(\phi, \mathcal{AS}_\alpha)$	$\Gamma_A^-(\phi, \mathcal{AS}_\alpha)$	$\Gamma_S^-(\phi, \mathcal{AS}_\alpha)$
$-i$	$\{\mathcal{A}_4, \mathcal{A}_{10}, \mathcal{A}_{11}\}$	$\{\}$	$\{\}$	$\{\}$

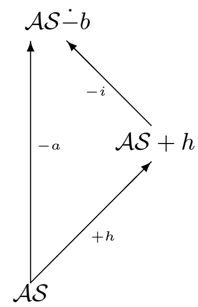

Fig. 1. Change graph for $\mathcal{AS}\dot{-}b$.

The change graph that results from these processes can be seen in Fig. 1, with the path costs being as follows:

$$V((\mathcal{AS}, \mathcal{AS}-a)) = \begin{pmatrix} \{\mathcal{A}_1, \mathcal{A}_6, \mathcal{A}_7\} \\ \{\mathcal{A}_3, \mathcal{A}_9\} \\ \{\} \\ \{\mathcal{A}_8\} \end{pmatrix}$$

$$V((\mathcal{AS}, \mathcal{AS}+h) \uplus V(\mathcal{AS}+h, \mathcal{AS}+h-i) =$$

$$\begin{pmatrix} \{\} \\ \{\mathcal{A}_6\} \\ \{\mathcal{A}_{12}\} \\ \{\} \end{pmatrix} \uplus \begin{pmatrix} \{\mathcal{A}_4, \mathcal{A}_{10}, \mathcal{A}_{11}\} \\ \{\} \\ \{\} \\ \{\} \end{pmatrix} = \begin{pmatrix} \{\mathcal{A}_4, \mathcal{A}_{10}, \mathcal{A}_{11}\} \\ \{\mathcal{A}_6\} \\ \{\mathcal{A}_{12}\} \\ \{\} \end{pmatrix}$$

Hence, the numeric costs are:

- $-a$: $| \{\mathcal{A}_1, \mathcal{A}_6, \mathcal{A}_7\} \cup \{\mathcal{A}_3, \mathcal{A}_9\} \cup \{\mathcal{A}_8\} | = 6$
- $+h-i$: $| \{\mathcal{A}_4, \mathcal{A}_{10}, \mathcal{A}_{11}\} \cup \{\mathcal{A}_6\} \cup \{\mathcal{A}_{12}\} | = 5$

Therefore, adding h to the knowledge base, then removing i represents the minimal change, and so the dialogue would proceed as follows:

id (t)	participant	Locution	Target	$C_{participant}$
...	...	...	...	...
9	α	$retract(b)$	$\{8\}$	$\{a, c, l, n\}$
10	α	$justify(\langle\emptyset, h \Rightarrow \neg r1\rangle)$	$\{8\}$	$\{a, c, h, l, n\}$

Note the empty set in the $justify$ locution. This is because α is only *assuming* h to be true, thus it remains unstated. Note also that i is not retracted because it has never been claimed; however, the argument revision process has identified that if α assumes h to be true, it cannot (rationally) claim i.

This example has illustrated two important principles: firstly, minimal change is not always represented by the shortest path (w.r.t. number of vertices visited) in a change graph; the second is that considering argument acceptability plays an important part in determining minimal change. Removing a would have resulted in four acceptability changes (three drops and one gain), while adding h and removing i results in only one. Neglecting these from the determination of the numeric costs would have resulted in $-a$: 1; $+h - i$: 4, hence resulting in $-a$ being minimal.

6 Conclusions and future work

In this paper, we have presented a method through which a software agent can reason about retraction of its commitments in a dialogue.

We defined, and proved certain properties of two new operators that describe changes to the argumentation systems of [10]: removal and expansion. Using these operators, an agent can reason about possible ways of ensuring that the conclusion of a rebutted argument can no longer be inferred in its commitments. This is done through either removing propositions from a knowledge base, and thus preventing the argument from being constructed, adding propositions to a knowledge base in order to activate an exception to a defeasible rule in an argument, or a combination of both.

A structure to model the possible choices, a change graph, was also defined, where the nodes represented argumentation systems and the edges represented an atomic removal or addition of a proposition. A cost function assigned to each edge a measure of the change that took place, in terms of the loss and gain of both argument acceptability, and arguments themselves. A further new operator was then defined to combine these edge costs in order to obtain an overall cost for each path from the original argumentation system, to the goal system. This operator considered the net effect of drops and gains, so as to exclude any arguments that were (in terms of both structure and acceptability) dropped (resp. gained) then subsequently regained (resp. re-dropped).

The work presented here forms part of a larger study into the connection between belief revision and argumentation, and its role in dialogue. In future work, we aim to further refine our model for measuring minimal change, by opening preferences, rules and contrariness to the revision process. We also aim to investigate the role of different acceptability semantics. In terms of preferences, we currently assume that all arguments identified by the drop and gain functions are of equal weight. However, ASPIC$^+$ incorporates a preference ordering over arguments, which intuitively should influence an agent's choice when deciding what propositions to add or remove in a revision process. Acceptability semantics are divided into two broad groups: sceptical and credulous. An argument that is sceptically acceptable has gone through a more rigorous process in order to determine its acceptability, and thus could be considered more important to an agent than an argument that is "only" credulously accepted.

This paper has demonstrated a core model through which principles of belief revision can be applied to a system of structured argumentation, using fundamental features of the system in the determination of minimal change.

Acknowledgements

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) of the UK government under grant number EP/G060347/1.

References

1. C.E. Alchourrón, P. Gärdenfors, and D. Makinson. On the logic of theory change: Partial meet contraction and revision functions. *The Journal of Symbolic Logic*, 50:2:510–530, 1985.
2. L. Amgoud, L. Bodenstaff, M. Caminada, P. McBurney, S. Parsons, H. Prakken, J. van Veenen, and G.A.W. Vreeswijk. Final review and report on formal argumentation system. Deliverable D2.6, ASPIC IST-FP6-002307, 2006.
3. J. Devereux and C. Reed. Strategic argumentation in rigorous persuasion dialogue. In *Proceedings of the Sixth International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2009)*, pages 94–113, 2009.
4. P. M. Dung. On the acceptability of arguments and its fundemental role in non-monotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.
5. M. A. Falappa, G. Kern-Isberner, and G. R. Simari. Belief revision and argumentation theory. In I. Rahwan and G.R. Simari, editors, *Argumentation in Artificial Intelligence*, chapter 17, pages 314–360. Springer, 2009.
6. P. Gärdenfors. *Knowledge in Flux*. MIT Press, 1988.
7. P. Gärdenfors. Belief revision: An introduction. In Peter Gärdenfors, editor, *Belief Revision*, pages 1–27. Cambridge University Press, 1992.
8. M.O. Moguillansky, N.D. Rotstein, M.A. Falappa, A.J. Garcia, and G.R. Simari. Argument theory change through defeater activation. In *Proceedings of the 3rd International Conference on Computational Models of Argument (COMMA 2010)*, pages 359–366, 2010.
9. C. da Costa Pereira, A.G.B. Tettamanzi, and S. Villata. Graded reinstatement in belief revision. In *IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pages 58–61, 2011.
10. H. Prakken. An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1:2:93–124, 2010.
11. N.D. Rotstein, M.O. Moguillansky, M.A. Falappa, A.J. Garcia, and G.R. Simari. Argument theory change: Revision upon warrant. In *Proceedings of the 2nd International Conference on Computational Models of Argument (COMMA 2008)*, pages 336–347, 2008.
12. N.D. Rotstein, M.O. Moguillansky, A.J. Garcia, and G.R. Simari. A dynamic argumentation framework. In *Proceedings of the third international conference on Computational Models of Argument (COMMA 2010)*, pages 427–438, 2010.
13. D.N. Walton and E.C.W. Krabbe. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. State University of New York Press, New York, 1995.
14. T.L. van der Weide and F. Dignum. Reasoning about and discussing preferences between arguments. In *Proceedings of the Eighth International workshop on Argumentation in Multi-Agent Systems (ArgMas 2011)*, 2011.