

An Argumentation Algorithm

Yijun Yu

The Open University

When an informal argumentation is formalised, the following algorithm can relate them as rebuttals or mitigations through a series of proofs.

1. DESCRIPTION OF THE ALGORITHM

The algorithm takes as input a set of arguments in the argumentation syntax.

An argumentation is a process involving a number of arguments in a number of rounds.

Each round may introduce one to many arguments to either rebuts the one of the original arguments, or to mitigates the effects of the rebuttals in previous rounds.

Thus an argument is associated with a natural number, indicating which round it is added during the argumentation process.

An argument also states the truth for one proposition, the claim, which is supported by a set of ground facts that are considered to be true, and a set of facts that are, for the sake of terminating endless argumentation, believed to be true too.

Considering a non-monotonic knowledge base grows with the addition of new arguments incrementally.

It accumulates the facts that support the initial arguments, until certain facts/domain knowledge rules previously believed to be true that no longer believed to be true anymore in the new situation (also known as trust assumptions). This leads to the possibility of rebuttals where the introduction of a new argument and its associated facts/domain knowledge rules makes the knowledge base inconsistent.

Similarly, an inconsistent knowledge base may restore its consistency by introducing a set of facts through the arguments that mitigates. The output of the algorithm visualises the argumentation process as a directed acyclic structure (we do not allow cyclic rebuttals / mitigations) where the nodes are arguments as defined earlier and the arrows are either a rebuttal or a mitigation.

An arrow is considered as a rebuttal only if the accumulated knowledge base from the node j to the source node i in every path was consistent, and there is a path from the source node $j + 1$ to the target node that creates an inconsistent knowledge base.

An arrow is considered as a mitigation only if the accumulated knowledge base at step $j + 1$ was inconsistent, and at step $j + 2$ becomes consistent after introducing the facts to mitigate the effect of a rebuttal.

If throughout the argumentation process there was a single claim, then this structure would degenerated into a sequence. However, as every claimed proposition may be argued, the overall argumentation structure may not be so linear. Therefore we perform Algorithm 1 first to enumerate all possible sequences in the argumentation, then for each sequence we perform the Algorithm 2 to check for rebuttals and mitigations.

Input: $A^* = \{(n, a) \mid (n \in \mathcal{N})\}$: a set of incremental arguments annotated by the natural number n to indicate which round these arguments are; B as given in Algorithm 2.

Output: $R^* = \{(a_i, a_j)\}$ and $M^* = \{(a_i, a_j, a_k)\}$ respectively for the rebuttals and mitigations relationships among the arguments.

```

1 begin
2    $A := \langle \rangle$  // initially the sequence is empty
3   for each  $a_i \in A^*$  do
4      $A := A ; \langle a_i \rangle$  // concatenate the sequence
5     for each  $a_j \in A^* \mid n(a_j) = n(a_i) + 1$  do
6        $R, M := \text{CheckingArgumentRelationships}(A; \langle a_j \rangle, B)$ 
7        $R^*, M^* := R^* \cup R, M^* \cup M$ 
8     end
9   end
10 end

```

Algorithm 1: AllArgumentRelationships

ACKNOWLEDGMENTS

Received February 1986; November 1993; accepted January 1996

Input: 1) $A = \langle (C_n, \Delta G_n, \Delta W_n) \mid (n \in \mathcal{N}) \wedge (\Delta G_n, \Delta W_n \vdash C_n) \rangle$: a sequence of incremental arguments where C_n is the proposition claimed by the increment, grounds ΔG_n and warrants ΔW_n are respectively incremental sets of ground facts and domain knowledge rules;
 2) $B = \{(p, q) \mid (p \vdash \neg q)\}$: a set of mappings between two facts or domain knowledge rules p and q , where q is called a *trust assumption* of an argument in a round earlier than that of p from a sequence $\langle a_0, \dots, a_n \rangle$. Note that if both $(p, q) \in B$ and $(q, r) \in B$, then (q, r) is removed from B when the incremental argument that concerns p is considered.

Output: For the argument at the i -th round with a claim C_i where $0 \leq i \leq n$:
 1) $R = \{(a_j, a_{j+1}) \mid (\mathcal{KB}_j \vdash C_i) \wedge (\mathcal{KB}_{j+1} \not\vdash C_i)\}$: a set of rebuttal relationships, where \mathcal{KB}_i is the knowledge base consists of all the facts and the domain knowledge rules in A , except for those trust assumptions denied by B in the latest incremental arguments a_i for $0 < i \leq n$;
 2) $M = \{(a_j, a_{j+1}, a_{j+2}) \mid (\mathcal{KB}_j \vdash C_i) \wedge (\mathcal{KB}_{j+1} \not\vdash C_i) \wedge (\mathcal{KB}_{j+2} \vdash C_i)\}$: a set of mitigation relationships.

```

1 begin
2    $\mathcal{KB}_0 := \{\}$ 
3   for  $i = 0, |A|, +1$  do
4     for each  $q \in (\Delta G_i \cup \Delta W_i)$  do
5       for  $j = |A|, i + 1, -1$  do
6         if  $\exists p \in (\Delta G_j \cup \Delta W_j) \mid (p, q) \in B$  then
7            $\mathcal{KB}_i := \mathcal{KB}_i \wedge q$ 
8         end
9       end
10    end
11    for  $i_0 = 0, i, +1$  do
12       $V_{i, i_0} := \text{eval}(\mathcal{KB}_i \vdash C_k) // \text{decreasoner}$ 
13    end
14  end
15  for  $i = 0, |A| - 1, +1$  do
16    for  $j = i, |A|, +1$  do
17      if  $V_{j, i} \wedge \neg V_{j+1, i}$  then
18         $R := R \cup \{(A_j, A_{j+1})\}$ 
19        if  $V_{j+2, i}$  then
20           $M := M \cup \{(A_j, A_{j+1}, A_{j+2})\}$ 
21        end
22      end
23    end
24  end
25 end

```

Algorithm 2: CheckingArgumentRelationships