Proceedings of the 23rd International Conference on Principles of Knowledge Representation and Reasoning
Main Track

# Elucidating Arguments Maps in Propositional Logic: Addressing Enthymemes and their Relationships

**Jonathan Ben-Naim**[1] , **Victor David**[2] , **Anthony Hunter**[3]
[1]Université Paul Sabatier, CNRS, IRIT, France
[2]Université Côte d'Azur, Inria, CNRS, I3S, France
[3]University College London, UK
jonathan.ben-naim@irit.fr, victor.david@inria.fr, anthony.hunter@ucl.ac.uk

**Abstract**

To better understand, and analyse, natural language arguments, it is desirable to represent them as logical arguments. However, most real-world arguments are enthymemes (i.e. some of the premises and/or claims are implicit), and therefore, there is a need to identify these implicit aspects. A ramification of this is that we may then need to edit some of the explicit premises and/or claim to remove redundant aspects and/or to allow the newly identified implicit formulae to work correctly with the explicit formulae. Furthermore, we may need to edit the claim so that it correctly attacks or supports other arguments as predicted by argument mining or as required by the user. To address these requirements, we propose a logic-based framework, based on classical propositional logic, for representing enthymemes, and manipulating them through a range of logical operations. We introduce meta-level rules to manipulate arguments (e.g. to add or delete premises, to edit claims, to split an argument into two arguments, and to merge two arguments into one). In order to direct the use of meta-level rules, we also introduce gain measures. When choosing a sequence of meta-level rules to apply, we can choose those that increase gain. This meta-level reasoning framework provides some clarity on the nature of enthymemes, and on how agents might elucidate them through a transparent and incremental process.

## 1 Introduction

There are a number of frameworks for modelling argumentation in logic. They incorporate a formal representation of individual arguments, where the premises imply the claim, and techniques for comparing conflicting arguments (Atkinson et al. 2017). However, real arguments (i.e. arguments presented by humans) usually do not have enough explicitly presented premises for the entailment of the claim. This is because there is some contextual, common, or commonsense, knowledge that can be assumed by a proponent of an argument and the recipient of it (Walton 2001).

To understand a natural language enthymeme, we can translate the explicit premises and claim into logical formulae, and then try to decode it, for instance, by identifying formulae to add to the premises that would then allow the premises to entail the claim.

**Example 1.** *Suppose, we hear the enthymeme "it is a square, the sum of its corners is 360 degrees". We can translate it into logic. Let* $\mathtt{s}$ *be "it is a square", and* $\mathtt{t}$ *be "the sum of its corners is 360 degrees". So* $\mathtt{s}$ *does not logically entail* $\mathtt{t}$*. Now let* $\mathtt{f}$ *be "it has four corners", and* $\mathtt{e}$ *be "each corner is 90 degrees". So for instance, with the set of formulae* $\{\mathtt{s}, \mathtt{s} \rightarrow \mathtt{f} \wedge \mathtt{e}, \mathtt{f} \wedge \mathtt{e} \rightarrow \mathtt{t}\}$ *we can infer the claim* $\mathtt{t}$*.*

Whilst human agents constantly need to understand enthymemes, whether in everyday or professional life, there is a lack of adequate methods for supporting this process. There are some proposals for extending argument mining to identify implicit knowledge (for a review, see (Sviridova, Cabrio, and Villata 2025)), though this is a challenging task. For logic-based representation of enthymemes, decoding can be modelled as abduction (Hunter 2007; Black and Hunter 2012; Hosseini, Modgil, and Rodrigues 2014), and there has been some consideration of how this can be undertaken within a dialogue (Black and Hunter 2008; de Saint-Cyr 2011b; de Saint-Cyr 2011a; Xydis et al. 2020; Panisson, McBurney, and Bordini 2022; Leiva, Gottifredi, and García 2023). However, these proposals only consider some kinds of enthymemes, and they lack logical tools to support a user in investigating, and reasoning with, enthymemes arising in natural language. Two recent proposals offer ways to judge candidates for decoding of enthymemes (Ben-Naim, David, and Hunter 2025; David and Hunter 2025), but they assume that somehow we can find these candidates. They do not provide a means for incrementally constructing candidates, and furthermore, they do not consider how this is part of a wider process of elucidating a set of arguments and counterarguments. We aim to address some of these shortcomings in this paper.

Translating natural language arguments into logic offers a clear and unambiguous representation of the explicit information in the arguments. This is the starting point for what we present in this paper. We do not investigate translating language into logic in this paper but the advent of large language models offers promising methods we could use (e.g. (Lu et al. 2022; Olausson et al. 2023; Pan et al. 2023; Lalwani et al. 2024; Yang et al. 2024; Feng and Hunter 2025; Liu 2025; Sadeddine and Suchanek 2025)).

As context for this paper, we assume argument mining is applied to some text to identify the arguments and the support and attack relationships between them. Then the text describing each argument is translated into classical logic which results in an *argument map* (a directed graph where a logical argument is assigned to each node). As the arguments are enthymemes, it may be necessary to edit them in order to *elucidate* (i.e. make clearer) the argument map — for instance, in one or more of the following ways:

- to *change premises* because some premises are missing (and so require additional premises), and/or there are unnecessary premises (and so require removal);
- to *change claims* by strengthening or weakening so they have the correct relationship (attack, support, or neither) with other arguments as identified by argument mining;
- to *restructure arguments* by dividing some arguments because they are complicated (perhaps because they have too many premises) and/or merging some arguments (perhaps because each has too few premises, and the graph could benefit by combining the arguments into fewer but more substantial arguments).

We illustrate the above ways of elucidation in the following examples where we assume an argument is a set of formulae (premises) and a formula (claim). An argument is valid if its premises entail its claim.

**Example 2.** *To illustrate changing premises, let* $\mathtt{r}$ *be "weather report predicts rain", and let* $\mathtt{u}$ *be "you should take an umbrella". The pair* $\langle\{\mathtt{r}\},\mathtt{u}\rangle$ *is an enthymeme since the premises are insufficient for entailing the claim. We can change the premises by adding a formula so that* $\langle\{\mathtt{r},\mathtt{r}\rightarrow\mathtt{u}\},\mathtt{u}\rangle$ *is now a valid argument.*

*Now let* $\mathtt{c}$ *be "weather report predicts cold weather", and consider the argument* $\langle\{\mathtt{r},\mathtt{r}\rightarrow\mathtt{u},\mathtt{c}\},\mathtt{u}\rangle$*. Since this has an irrelevant premise* $\mathtt{c}$*, we can delete this premise to give the valid argument* $\langle\{\mathtt{r},\mathtt{r}\rightarrow\mathtt{u}\},\mathtt{u}\rangle$*.*

**Example 3.** *To illustrate changing a claim, consider the atoms from Example 2, and let* $\mathtt{w}$ *be "the weather report predicts strong wind". Below, the left argument has a claim that does not logically contradict the premises of the right argument, even though argument mining suggests it is a counterargument. By editing premises and claim of the counterargument to give* $\langle\{\mathtt{w},\mathtt{r},\mathtt{w}\rightarrow\neg\mathtt{u}\},\neg(\mathtt{r}\rightarrow\mathtt{u})\rangle$*, we have a counterargument that logically attacks the right argument by negating the defeasible implication* $\mathtt{r}\rightarrow\mathtt{u}$*.*

$$\boxed{\langle\{\mathtt{w}\},\mathtt{w}\rangle} \xrightarrow{-} \boxed{\langle\{\mathtt{r},\mathtt{r}\rightarrow\mathtt{u}\},\mathtt{u}\rangle}$$

**Example 4.** *To illustrate restructuring, let* $\mathtt{l}$ *be "he needs his laptop",* $\mathtt{p}$ *be "he needs to make a presentation at the meeting",* $\mathtt{i}$ *be "he needs to impress his boss at the meeting", and* $\mathtt{a}$ *be "he is applying for promotion", which can be used for the argument* $\langle\{\mathtt{l},\mathtt{l}\rightarrow\mathtt{p},\mathtt{p}\rightarrow\mathtt{i},\mathtt{i}\rightarrow\mathtt{a}\},\mathtt{a}\rangle$*. Here, it might be desirable to split this argument into simpler arguments (each making a more focused point) as follows.*

$$\boxed{\langle\{\mathtt{l},\mathtt{p}\rightarrow\mathtt{p}\},\mathtt{p}\rangle} \xrightarrow{+} \boxed{\langle\{\mathtt{p},\mathtt{p}\rightarrow\mathtt{i}\},\mathtt{i}\rangle} \xrightarrow{+} \boxed{\langle\{\mathtt{i},\mathtt{i}\rightarrow\mathtt{a}\},\mathtt{a}\rangle}$$

*Further benefits of this restructuring include the first new argument represents only one possible way to achieve the final goal (i.e. applying for a promotion). In the original argument, if we assume, one or two attacks, such that his laptop does not work or that he forgot it, the entire argument collapses (in a deductive setting). However, alternative options could exist, such as borrowing another laptop.*

In the following, we review deductive argumentation (Besnard and Hunter 2001), and we show how this can be used for modelling enthymemes. Then we introduce our novel framework based a set of *meta-level rules* that manipulate argument maps, and a *gain function* that can be used to ensure that applications of meta-rules improves the elucidation of the argument map (e.g. increases the number of valid arguments and/or the number of logical arguments that correctly attack or support the other arguments). We illustrate how the framework can be used, and provide some results, with proofs in (Ben-Naim, David, and Hunter 2026).

## 2 Deductive Argumentation

We adapt a version of deductive argumentation proposed for handling enthymemes (Hunter 2007; Hunter 2022).

Let $\mathcal{L}$ be the set of propositional formulae composed as usual from atoms $\mathcal{A}$ and the logical connectives $\wedge$, $\vee$, $\neg$, and $\rightarrow$. We use $\alpha,\beta,\gamma,\delta,\phi,\psi,\ldots$ for arbitrary formulae and $\Delta,\Phi,\Psi,\ldots$ for arbitrary sets of formulae. A knowledgebase $K\subseteq\mathcal{L}$ is a set of formulae. Let $\vdash$ denote the classical consequence relation, and write $\Delta\vdash\bot$ to denote that $\Delta$ is inconsistent. Let $\mathsf{Atoms}(\Delta)$ give the atoms appearing in the formulae in $\Delta$.

An **argument** is a pair $\langle\Phi,\alpha\rangle$ where $\Phi\subseteq\mathcal{L}$ and $\alpha\in\mathcal{L}$. This is a very general definition. It does not assume that $\Phi$ is consistent, or that $\Phi$ entails $\alpha$. Let $\mathcal{A}=\{\langle\Phi,\alpha\rangle\mid\Phi\subseteq\mathcal{L}\text{ and }\alpha\in\mathcal{L}\}$ be the set of all arguments. For $\langle\Phi,\alpha\rangle\in\mathcal{A}$, let $\mathsf{Support}(\langle\Phi,\alpha\rangle)$ be $\Phi$, and let $\mathsf{Claim}(\langle\Phi,\alpha\rangle)$ be $\alpha$.

Some kinds of arguments include: If $\Phi\vdash\alpha$, then $\langle\Phi,\alpha\rangle$ is **valid**; If $\Phi\not\vdash\bot$, then $\langle\Phi,\alpha\rangle$ is **consistent**; If $\Phi\vdash\alpha$, and there is no $\Phi'\subset\Phi$ such that $\Phi'\vdash\alpha$, then $\langle\Phi,\alpha\rangle$ is **minimal**; And if $\Phi\vdash\alpha$, and $\Phi\not\vdash\bot$, then $\langle\Phi,\alpha\rangle$ is **expansive** (i.e. it is valid and consistent, but it may have unnecessary premises). A **strict** argument is a valid, consistent, and minimal, argument. In addition, we require a further kind of argument that has the potential to be transformed into a valid argument: If $\Phi\not\vdash\alpha$, and $\Phi\not\vdash\neg\alpha$, then $\langle\Phi,\alpha\rangle$ is a **precursor** (i.e. it is a precursor for a valid argument). Therefore, if $\langle\Phi,\alpha\rangle$ is a precursor, then there exists some $\Psi\subset\mathcal{L}$ such that $\Phi\cup\Psi\vdash\alpha$ and $\Phi\cup\Psi\not\vdash\bot$, and hence $\langle\Phi\cup\Psi,\alpha\rangle$ is expansive.

**Example 5.** *Let* $\Delta=\{\mathtt{a},\neg\mathtt{a}\vee\mathtt{b},\mathtt{c},\neg\mathtt{b},\mathtt{b},\neg\mathtt{c},\neg\mathtt{b}\vee\mathtt{c}\}$*. Some arguments from* $\Delta$ *that are valid include* $\{\mathtt{A}_1,\mathtt{A}_2,\mathtt{A}_3,\mathtt{A}_4,\mathtt{A}_5,\mathtt{A}_8\}$ *of which* $\{\mathtt{A}_1,\mathtt{A}_3,\mathtt{A}_4,\mathtt{A}_8\}$ *are expansive,* $\{\mathtt{A}_2,\mathtt{A}_4,\mathtt{A}_8\}$ *are minimal, and* $\{\mathtt{A}_4,\mathtt{A}_8\}$ *are strict arguments. Also, some arguments that are not valid include* $\{\mathtt{A}_6,\mathtt{A}_7\}$ *of which* $\mathtt{A}_6$ *is a precursor.*

$$\begin{array}{ll}
\mathtt{A}_1=\langle\{\mathtt{a},\neg\mathtt{a}\vee\mathtt{b},\mathtt{c},\mathtt{b}\},\mathtt{b}\rangle & \mathtt{A}_2=\langle\{\mathtt{c},\neg\mathtt{c}\},\mathtt{b}\rangle\\
\mathtt{A}_3=\langle\{\mathtt{a},\neg\mathtt{a}\vee\mathtt{b},\mathtt{c}\},\mathtt{b}\rangle & \mathtt{A}_4=\langle\{\mathtt{a},\neg\mathtt{a}\vee\mathtt{b}\},\mathtt{b}\rangle\\
\mathtt{A}_5=\langle\{\mathtt{a},\neg\mathtt{a}\vee\mathtt{b},\mathtt{c},\neg\mathtt{c}\},\mathtt{b}\rangle & \mathtt{A}_6=\langle\{\neg\mathtt{a}\vee\mathtt{b}\},\mathtt{b}\rangle\\
\mathtt{A}_7=\langle\{\neg\mathtt{a}\vee\mathtt{b},\neg\mathtt{b}\vee\mathtt{c},\neg\mathtt{c}\},\mathtt{b}\rangle & \mathtt{A}_8=\langle\{\},\mathtt{a}\vee\neg\mathtt{a}\rangle
\end{array}$$

An **enthymeme** is a precursor that can be generated from a strict argument (i.e. a precursor $\langle\Phi,\alpha\rangle$ is an enthymeme for argument $\langle\Psi,\beta\rangle$ iff $\Phi\cup\Psi\not\vdash\bot$ and $\{\alpha,\beta\}\not\vdash\bot$). So if a proponent has a strict argument that it wishes to present to a recipient, the **intended argument**, then the proponent may send an enthymeme instead of the intended argument to the recipient. The notion of an enthymeme, and the connection

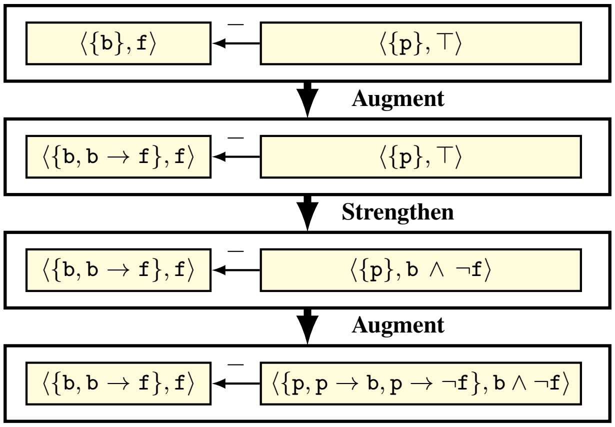


Figure 1: Example of a metaproof. Each box contains an argument map. Let the left and right nodes of each argument map be $\mathtt{n}_1$ and $\mathtt{n}_2$. The argument is given for each node in each argument map where b is bird, f is fly, p is penguin. The expected relation is given above the arc in each argument map. By convention, only attack and support relationships are given in the figures for metaproofs. The meta-level rule used is on the right of each big arrow.

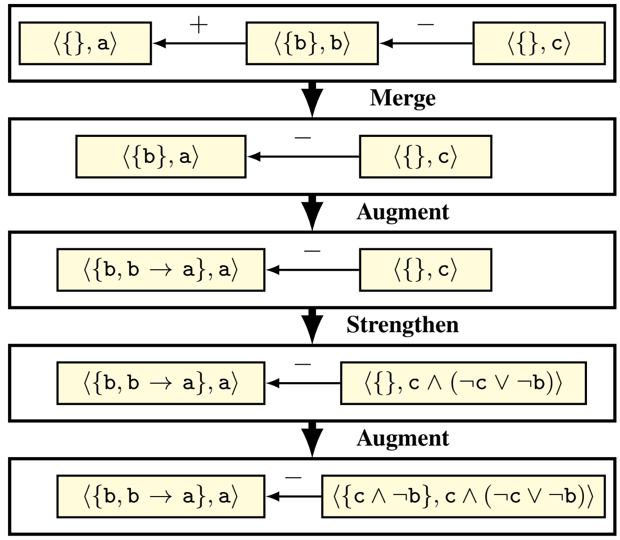


Figure 2: Example of a metaproof. Each box contains an argument map. The meta-level rule used is on the right of each big arrow.

with its intended argument, is very general. Often, there is a tighter relationship between an enthymeme $\langle \Phi, \alpha \rangle$ and intended argument $\langle \Psi, \beta \rangle$ — with for instance, $\Phi \cap \Psi \neq \emptyset$, or $\Phi \subseteq \Psi$ and $\{\beta\} \vdash \alpha$, holding.

**Example 6.** *Consider the atoms in Example 2. For an intended argument* $\langle \{\mathtt{r}, \mathtt{r} \rightarrow \mathtt{u}\}, \mathtt{u} \rangle$*, the enthymeme sent by the proponent to the recipient may be* $\langle \{\mathtt{r}\}, \mathtt{u} \rangle$*.*

Argument $A$ **attacks** argument $B$ if $\mathsf{C}(A) \vdash \neg \bigwedge \mathsf{S}(B)$. This is only one option but it is the most general definition considered in (Besnard and Hunter 2001). For other definitions, see (Cayrol 1995; Besnard and Hunter 2001)

**Example 7.** *Argument* $\langle \{\mathtt{a} \vee \mathtt{b}, \mathtt{c}\}, (\mathtt{a} \vee \mathtt{b}) \wedge \mathtt{c} \rangle$ *attacks argument* $\langle \{\neg\mathtt{a}, \neg\mathtt{b}\}, \neg\mathtt{a} \wedge \neg\mathtt{b} \rangle$*,*

An argument $A$ **supports** argument $B$ if there is an $\phi \in \mathsf{S}(B)$ such that $\mathsf{C}(A) \vdash \phi$.

**Example 8.** *Argument* $\langle \{\mathtt{a} \vee \mathtt{b} \vee \mathtt{c}, \neg\mathtt{c}\}, \mathtt{a} \vee \mathtt{b} \rangle$ *supports argument* $\langle \{\mathtt{a} \vee \mathtt{b} \vee \mathtt{d}, \neg\mathtt{d}, \neg\mathtt{b}\}, \mathtt{a} \rangle$*.*

In the next section, we explain how we can consider a set of arguments and counterarguments in a graphical structure.

## 3 Argument Maps

We assume that we start with an argument map (as defined below) that has been obtained by argument mining of an article (or perhaps a dialogue). In the definition below, $R$ is the expected relation between the pairs of nodes. For instance, if argument mining classifies the relationship from a text argument associated with node $x_1$ to a text argument associated with $x_2$ as attack (respectively support), then the expected assignment $R(x, y)$ is $-$ (respectively $+$), otherwise it is $0$ (i.e. neutral).

**Definition 1.** *An* **argument map** *is a tuple* $(N, R, I)$ *such that* $N$ *is a set of* **nodes***,* $R : N \times N \rightarrow \{+, 0, -\}$ *is an* **expected relation** *between the pairs of nodes, and* $I : N \rightarrow \mathcal{A}$ *is an injective* **instantiation** *function that assigns an argument to each node.*

We give examples of argument maps in Figures 1 and 2. For instance, in Figure 1, the top argument map is instantiated with enthymemes and the bottom is instantiated with strict arguments where each strict argument is obtained from an enthymeme in the first argument map.

The **support** for an argument map $M = (N, R, I)$, denoted $\mathsf{Support}(M)$, is the union of the premises of its arguments (i.e. $\mathsf{Support}(M) = \cup_{x \in N} \mathsf{S}(I(x))$). Let $\mathcal{M} = \{M \mid \mathsf{Support}(M) \subseteq \mathcal{L}\}$ be the set of all argument maps.

Given an argument map $M = (N, R, I)$, we can obtain the **logical relation** between the arguments as follow where $\mathsf{LR}^M : N \times N \rightarrow \{+, 0, -\}$. For each $x, y \in N$, if $I(x)$ attacks $I(y)$, then $\mathsf{LR}^M(x, y) = -$, else if $I(x)$ supports $I(y)$, then $\mathsf{LR}^M(x, y) = +$, else $\mathsf{LR}^M(x, y) = 0$. So if an argument both attacks and supports another argument, it is the attack that we identify. By letting attacks take precedence over support, we have a function.

**Example 9.** *In Figure 1, let the top argument map be* $M_1$*, and let the bottom argument map be* $M_2$*. Therefore,* $\mathsf{LR}^{M_1}(\mathtt{n}_1, \mathtt{n}_1) = 0$, $\mathsf{LR}^{M_1}(\mathtt{n}_1, \mathtt{n}_2) = 0$, $\mathsf{LR}^{M_1}(\mathtt{n}_2, \mathtt{n}_1) = 0$, $\mathsf{LR}^{M_1}(\mathtt{n}_2, \mathtt{n}_2) = 0$, $\mathsf{LR}^{M_2}(\mathtt{n}_1, \mathtt{n}_1) = 0$, $\mathsf{LR}^{M_2}(\mathtt{n}_1, \mathtt{n}_2) = 0$, $\mathsf{LR}^{M_2}(\mathtt{n}_2, \mathtt{n}_1) = -$*, and* $\mathsf{LR}^{M_2}(\mathtt{n}_2, \mathtt{n}_2) = 0$*.*

To recap, for an argument map $(N, R, I)$, $R$ is the relation identified by argument mining between the natural language arguments, and so it is the relation we may want to hold for the logical arguments. In other words, we may choose to seek an instantiation $I$ such that $\mathsf{LR}^M = R$.

## 4 Meta-level Rules

We informally describe three types of meta-level rule for manipulating arguments and then, in Table 1, we formalize

| Operation | Preconditions | | Postconditions | |
|---|---|---|---|---|
| | Graphical | Logical | Logical | Graphical |
| Divide | $n_1 \in N$<br>$n_2, n_3 \notin N$ | $I(n_1) = \langle \Phi \cup \Psi, \alpha \rangle$ | $I(n_2) := \langle \Psi, \beta \rangle$<br>$I(n_3) := \langle \Phi, \alpha \rangle$ | $N := (N \setminus \{n_1\}) \cup \{n_2, n_3\}$<br>$R := \mathsf{Divide}(R, n_1, n_2, n_3)$ |
| Merge | $n_1, n_2 \in N$<br>$n_3 \notin N$<br>$R(n_1, n_2) = +$ | $I(n_1) = \langle \Phi, \alpha \rangle$<br>$I(n_2) = \langle \Psi, \beta \rangle$ | $I(n_3) := \langle \Phi \cup \Psi, \alpha \rangle$ | $N := (N \setminus \{n_1, n_2\}) \cup \{n_3\}$<br>$R := \mathsf{Merge}(R, n_1, n_2, n_3)$ |
| Augment | $n \in N$ | $I(n) = \langle \Phi, \alpha \rangle, \Psi \subseteq K$ | $I(n) := \langle \Phi \cup \Psi, \alpha \rangle$ | |
| Shrink | $n \in N$ | $I(n) = \langle \Phi, \alpha \rangle, \Psi \subseteq \Phi$ | $I(n) := \langle \Psi, \alpha \rangle$ | |
| Strengthen | $n \in N$ | $I(n) = \langle \Phi, \alpha \rangle, \{\beta\} \vdash \alpha$ | $I(n) := \langle \Phi, \beta \rangle$ | |
| Weaken | $n \in N$ | $I(n) = \langle \Phi, \alpha \rangle, \{\alpha\} \vdash \beta$ | $I(n) := \langle \Phi, \beta \rangle$ | |

Table 1: Definitions for meta-level rules where $(N, R, I)$ is an argument map and $K$ is a decoding knowledgebase. The first two rules concern arguments, the second two rules concern premises, and the third two rules concern claims. For each rule, if the preconditions are satisfied (i.e. there are the arguments of the specified form, and the specified consequence, and subset relationships hold), then the argument(s) in the post-condition are obtained. Note, for the augment rule, any formula added to the premises comes from a decoding knowledgebase $K$. The Divide and Merge functions are defined below in Definition 2 and Definition 3 respectively.

them. These are the *argument restructure* rules that merge or divide arguments, the *premise amendment* rules that add or delete formulae from the premises, and *claim editing* rules that make the claim weaker or stronger.

**Argument restructure rules** These can divide an argument into two arguments, or merge two arguments into a single argument. The divide rule is useful when an argument appears too complicated, whereas when a pair of arguments appear too busy, they can be merged into one, as illustrated in Example 10. We introduce (in Section 6) the *degree of detail* which judges an argument as overly complex if it exceeds a specified number of formulae in its premises. Further ways of judging arguments include evaluating the complexity of the formulae used in the premises and/or claim (e.g. the number of occurrences of logical connectives, the degree of nesting of logical connectives, and the number of different propositional letters occurring).

**Example 10.** *An argument* $\mathtt{A}_1 = \langle \{\mathtt{a}, \mathtt{a} \rightarrow \mathtt{c}, \mathtt{c} \rightarrow \mathtt{b}\}, \mathtt{b} \rangle$ *can be divided into* $\mathtt{A}_2 = \langle \{\mathtt{a}, \mathtt{a} \rightarrow \mathtt{c}\}, \mathtt{c} \rangle$ *and* $\mathtt{A}_3 = \langle \{\mathtt{c} \rightarrow \mathtt{b}\}, \mathtt{b} \rangle$. *Similarly,* $\mathtt{A}_2$ *and* $\mathtt{A}_3$ *can be merged into* $\mathtt{A}_1$.

**Premise amendment rules** An amendment rule either adds formulae from a knowledgebase $K$ (that we refer to as the **decoding knowledgebase**) to the premises (augment), or subtracts formulae from the premises (shrink) as in Example 11. If appropriate formulae are added and/or deleted, it can result in an enthymeme being made into a strict argument as in $\mathtt{A}_3$ of Example 11. Whether this is possible depends on what is in the decoding knowledgebase $K$.

**Example 11.** *For argument* $\mathtt{A}_1 = \langle \{\mathtt{c}, \mathtt{d}, \mathtt{f}\}, \mathtt{a} \rangle$*, an augmentation is* $\mathtt{A}_2 = \langle \{\mathtt{c}, \neg\mathtt{c} \vee \neg\mathtt{d} \vee \mathtt{a}, \mathtt{d}, \mathtt{f}\}, \mathtt{a} \rangle$. *For* $\mathtt{A}_2$, $\mathtt{A}_3 = \langle \{\mathtt{c}, \neg\mathtt{c} \vee \neg\mathtt{d} \vee \mathtt{a}, \mathtt{d}\}, \mathtt{a} \rangle$ *is a shrinkage.*

**Claim editing rules** The claim editing rules are intended to change the claim of the argument so that it is implied by the premises or that it attacks or supports another argument as required by the expected relation.

**Example 12.** *Consider the enthymeme* $\langle \{\mathtt{b}\}, \mathtt{e} \rangle$ *where* b *is "bird", and* e *is "lays eggs". Also let* $K = \{\mathtt{b} \wedge \mathtt{f}, \mathtt{b} \wedge \mathtt{f} \rightarrow \mathtt{c}\}$ *where* f *is "female" and* c *is "capable of laying eggs". Suppose we augment the premises to give* $\langle \{\mathtt{b}, \mathtt{b} \wedge \mathtt{f} \rightarrow \mathtt{c}\}, \mathtt{e} \rangle$*, To make this valid, we can apply the weakening and strengthening rules to change the claim, plus apply the augment and shrink rules to change the premises, to give* $\langle \{\mathtt{b} \wedge \mathtt{f}, \mathtt{b} \wedge \mathtt{f} \rightarrow \mathtt{c}\}, \mathtt{c} \rangle$.

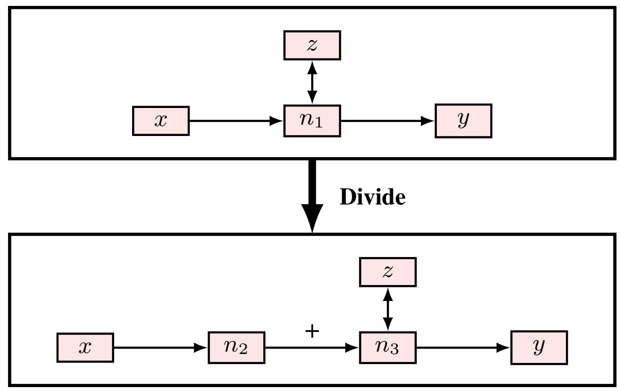


Figure 3: Schematic diagram of the divide operation (Definition 2) on a node $n_1$ and the nodes that it is connected to. As a result, $n_2$ and $n_3$ are introduced, with the connections of $n_1$ adjusted for $n_2$ and $n_3$. Everything else in the map remains the same.

We can characterize elucidation of enthymemes in terms of the six meta-level rules in Table 1. As we will discuss in the next section, these rules are very liberal. In a later section, we will introduce gain measures to help control the use of the meta-rules so that they guide the derivation of desirable elucidations of argument maps.

In general, elucidating an argument map, requires a sequence of steps (i.e. a sequence of meta-rules) to incrementally change the argument map (as in Figures 1 and 2).

The following definition returns the expected relation after the divide meta-level proof rule is applied. The function removes the arcs involving $n_1$ (the node being divided) and adds corresponding arcs for $n_2$ and $n_3$ (the nodes resulting from the division of $n_1$). A schematic representation of what arcs are added and deleted is given in Figure 3.

**Definition 2.** *The function* $\mathsf{Divide}(R, n_1, n_2, n_3)$ *returns the following set where* $\# \in \{+, 0, -\}$*:*

$$(R \setminus \mathsf{Sub}(R, n_1)) \cup \mathsf{Add}(R, n_1, n_2, n_3)$$

*where* $\mathsf{Sub}(R, n_1)$ *is* $\{(x, y, \#) \in R \mid x = n_1 \text{ or } y = n_1\}$, *and* $\mathsf{Add}(R, n_1, n_2, n_3)$ *is the union of the four sets below.*

1. $\{(n_2, n_3, +)\}$
2. $\{(x, n_2, \#) \mid (x, n_1, \#) \in R \text{ and } (n_1, x, \#) \notin R\}$
3. $\{(n_3, y, \#) \mid (n_1, y, \#) \in R \text{ and } (y, n_1, \#) \notin R\}$
4. $\{(z, n_3, \#), (n_3, z, \#) \mid (z, n_1, \#), (n_1, z, \#) \in R\}$

We explain each of the above four sets as follows: (1) this is the assumption that the arc from one of the new nodes to the other is a support relation; (2) a unidirectional attack (undercut) or support on $n_1$ is taken by $n_2$; (3) a unidirectional attack (undercut) or support by $n_1$ is taken by $n_3$; and (4 ) a bidirectional attack (rebuttal) or support involving $n_1$ is taken by $n_3$.

The following definition returns the expected relation after the merge meta-level proof rule is applied. The function removes the arcs involving $n_1$ and $n_2$ (the nodes being merged) and adds corresponding arcs for $n_3$ (the node resulting from the merger of $n_1$ and $n_2$). A schematic representation of this is given in Figure 4.

**Definition 3.** *The function* $\mathsf{Merge}(R, n_1, n_2, n_3)$ *returns the following set where* $\# \in \{+, 0, -\}$*:*

$$(R \setminus \mathsf{Sub}(R, n_1, n_2)) \cup \mathsf{Add}(R, n_1, n_2, n_3)$$

*where* $\mathsf{Sub}(R, n_1, n_2)$ *is*

$$\{(x, y, \#) \in R \mid x \in \{n_1, n_2\} \text{ or } y \in \{n_1, n_2\}\}$$

*and* $\mathsf{Add}(R, n_1, n_2, n_3)$ *is the union of the two sets below.*

1. $\{(x, n_3, \#) \mid (x, y, \#) \in R \text{ and } y \in \{n_1, n_2\}\}$
2. $\{(n_3, y, \#) \mid (x, y, \#) \in R \text{ and } x \in \{n_1, n_2\}\}$

In the above definition, the subtracted arcs are those involving the removed nodes $n_1$ and $n_2$, and the added arcs are those with node $n_3$ replacing either $n_1$ and $n_2$.

Note, in Definitions 2 and 3, when we divide or merge, we are specifying what we want from the expected relation $R$ in the restructured argument map, and so they do not depend on the current logical relationships. For instance, for divide, we introduce two new nodes $n_2$ and $n_3$ with $R(n_2, n_3) = +$. So then in subsequent proof steps, we will try to edit the premises and/or claim of the new supporting argument so that eventually $I(n_2)$ does indeed support $I(n_3)$.

## 5 Metaproofs

Now we take an argument map (e.g. from argument mining of an article or a dialogue), and then iteratively apply meta-level rules from Table 1, to form a metaproof (as defined below) to obtain a revised argument map.

**Definition 4.** *A* **metaproof** *is a tuple* $P = (X, K)$ *where* $X = (M_1, \ldots, M_k)$ *is a sequence of argument maps, and K is a decoding knowledgebase s.t. for each* $i \in \{1, \ldots, k-1\}$, $M_{i+1}$ *is obtained from* $M_i$ *using one application of a meta-level rule. We say that* $M_k$ *is* **obtained from** $M_1$ *using* $P$*. Let* $\mathcal{P}$ *be the set of metaproofs and* $\mathcal{P}^K \subseteq \mathcal{P}$ *be the set of metaproofs where K is the decoding knowledgebase.*

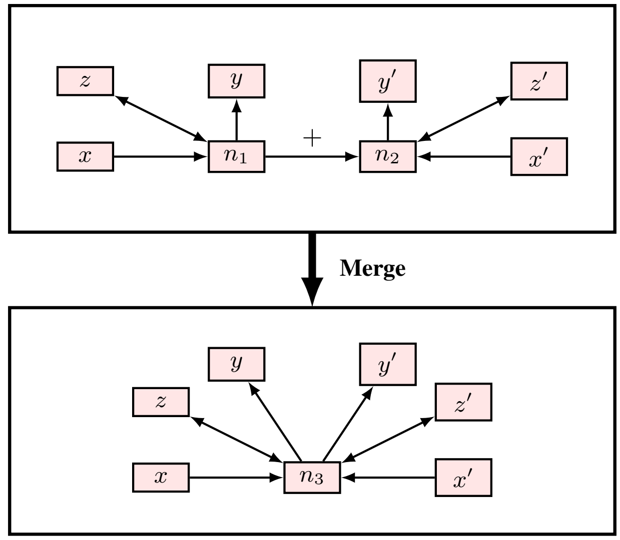


Figure 4: Schematic diagram of merge operation (Definition 3) on nodes $n_1$ and $n_2$ and the nodes that they connect to. As a result, $n_3$ introduced, with the connections of $n_1$ and $n_2$, reassigned to $n_3$. Everything else in the map remains the same.

Examples of metaproofs are given in Figures 1 and 2. In these examples, the resulting argument maps contain strict arguments, but this does not need to be the case.

For a metaproof $P = ((M_1, \ldots, M_k), K)$, we require the following subsidiary functions: Let $\mathsf{In}(P) = M_1$, $\mathsf{Out}(P) = M_k$, and $\mathsf{Len}(P) = k$.

The metaproof **connection** function $\oplus$ is defined as follows where $P_1 = ((M^1_1, \ldots, M^1_{n_1}), K_1)$ and $P_2 = ((M^2_1, \ldots, M^2_{n_2}), K_2)$ are metaproofs: If $M^1_{n_1} = M^2_1$, then $P_1 \oplus P_2$ is $((M^1_1, \ldots, M^1_{n_1}, M^2_2, \ldots, M^2_{n_2}), K_1 \cup K_2)$.

An **atomic metaproof** is a metaproof $((M_1, M_2), K)$. So it is a metaproof involving exactly one proof rule. For $P \in \mathcal{P}$, where $P = ((M_1, \ldots, M_k), K)$, let $\mathsf{Atomic}(P) = \{((M_i, M_{i+1}), K) \mid i \in \{1, \ldots, k\}\}$ be the set of atomic metaproofs in $P$. So for atomic metaproofs $P_1, \ldots, P_{k-1}$, $\mathsf{Atomic}(P) = \{P_1, \ldots, P_{k-1}\}$ iff $P = P_1 \oplus \ldots \oplus P_{k-1}$.

For an atomic metaproof $((M_1, M_2), K)$, if $M_1 = M_2$, then $M_2$ can be obtained (trivially) from $M_1$ using augment or shrink (with no premises added or deleted) or strengthen or weaken (with the claim unchanged), whereas if $M_1 \neq M_2$, then there is only one of the meta-level rules in Table 1 that can be applied to obtain $M_2$ from $M_1$.

We now consider some subsidiary definitions. For a metaproof $P = (X, K)$: $P$ is **finite** iff $X$ is a finite sequence; $P$ is **minimal** iff there is no metaproof $P'$ s.t. $\mathsf{In}(P) = \mathsf{In}(P')$ and $\mathsf{Out}(P) = \mathsf{Out}(P')$ and $\mathsf{Len}(P) > \mathsf{Len}(P')$; and $P$ is **non-repetitive** iff for all $n, m \in \{1, \ldots, k\}$, if $n \neq m$, then $M_n \neq M_m$.

**Proposition 1.** *If* $P \in \mathcal{P}$ *is minimal, then* $P$ *is non-repetitive.*

**Proposition 2.** *If* $K_1 \subseteq K_2$*, then* $\mathcal{P}^{K_1} \subseteq \mathcal{P}^{K_2}$.

The logical rules are reversible in the sense that for any argument, and sequence of rules, we can apply further rules to get the original argument back.

**Proposition 3.** *For each $P \in \mathcal{P}$ s.t. $\mathsf{In}(P) = A$, there is a metaproof $P'$ s.t. $\mathsf{In}(P \oplus P') = A$, and $\mathsf{Out}(P \oplus P') = A$.*

Different choices of proof system (i.e. different sets of meta-level rules) allow for different types of elucidation. Let $\mathcal{P}_{\mathsf{amend}}$ (respectively $\mathcal{P}_{\mathsf{edit}}$, and $\mathcal{P}_{\mathsf{ae}}$) be the metaproofs obtained only with the premise amendment proof rules, (respectively, only with the claim edit proof rules, and only with the premise amendment and claim edit proof rules).

**Proposition 4.** *If $P \in \mathcal{P}_{\mathsf{ae}}$, and $\mathsf{In}(P) = (N_1, R_1, I_1)$ and $\mathsf{Out}(P) = (N_2, R_2, I_2)$ then $(N_1, R_1)$ and $(N_2, R_2)$ are isomorphic.*

The next result shows that a claim can be changed into any other formula using strengthen and weaken.

**Proposition 5.** *For any arguments $\langle \Phi, \alpha \rangle, \langle \Phi, \beta \rangle \in \mathcal{A}$, there is a $P \in \mathcal{P}_{\mathsf{edit}}$ s.t. $\mathsf{In}(P) = (N_1, R_1, I_1)$ and $\mathsf{Out}(P) = (N_2, R_2, I_2)$ and there is an $n \in N_1 \cap N_2$ s.t. $I_1(n) = \langle \Phi, \alpha \rangle$, and $I_2(n) = \langle \Phi, \beta \rangle$.*

The next result shows that if all formulae are in the decoding knowledgebase, then using the augment, shrink, strengthen, and weaken, meta-level rules are sufficient for any argument to be turned into any other argument.

**Proposition 6.** *If decode knowledgebase $K$ is $\mathcal{L}$, then for any arguments $\langle \Phi, \alpha \rangle, \langle \Phi, \beta \rangle \in \mathcal{A}$, there is a $P \in \mathcal{P}_{\mathsf{ae}}$ s.t. $\mathsf{In}(P) = (N_1, R_1, I_1)$ and $\mathsf{Out}(P) = (N_2, R_2, I_2)$ and there is an $n \in N_1 \cap N_2$ s.t. $I_1(n) = \langle \Phi, \alpha \rangle$, and $I_2(n) = \langle \Psi, \beta \rangle$.*

The above result shows that metaproofs are liberal in what can be derived. Furthermore, there are potentially many metaproofs that can be generated. (c.f. natural deduction allows many possible proofs). Therefore, if we want to use metaproofs in automated reasoning, we need to impose constraints to control them so they give useful elucidations. We do this next by introducing the notion of a gain function.

## 6 Measuring Gain

So far we have assumed that a metaproof $P$ has been constructed without considering how rules are chosen. We now consider how we can direct the choice of rule using the following general notion of a gain function.

**Definition 5.** *Let $\mathcal{P}$ be the set of meta-proofs. A* **gain** *function $G : \mathcal{P} \to \mathbb{R}$ s.t. for all $P \in \mathcal{P}$, $G(P) = 0$ if $\mathsf{In}(P) = \mathsf{Out}(P)$.*

As an illustration, the **strict gain function** $G_{\mathsf{strict}} : \mathcal{P}_{\mathsf{ae}} \to \mathbb{Z}$ gives the additional number of arguments that are elucidated as strict arguments between the input and output argument maps: If $M_2 = (N, R, I_2)$ be obtained from $M_1 = (N, R, I_1)$ using $T$, then $G_{\mathsf{strict}}(T)$ is

$$|\{n \in N \mid I_2(n) \text{ is strict and } I_1(n) \text{ is not strict}\}|$$

In Defn 5, the constraint $G(P) = 0$ if $\mathsf{In}(P) = \mathsf{Out}(P)$ captures whether or not the output argument map is better than the input argument map. More specifically, if the input is the same as the output, then there is zero gain.

Further constraints that we may wish to adopt include the following which capture potentially desirable features of specific gain functions where $P \in \mathcal{P}$:

**C1** $G(P) > 0 \Leftrightarrow$ for all $P' \in \mathsf{Atomic}(P)$, $G(P') > 0$.

**C2** $G(P) > 0 \Rightarrow$ there is a $P' \in \mathsf{Atomic}(P)$, $G(P') > 0$.

**C3** $G(P) < 0 \Rightarrow$ there is a $P' \in \mathsf{Atomic}(P)$, $G(P') < 0$.

The first constraint above captures the requirement that for the metaproof to have positive gain, then every atomic step has to be positive. The second constraint is a weaker version of the first in that if the metaproof has a positive gain, then there has to be an atomic step that has positive gain. The third constraint is the counterpart to the second constraint in that if the metaproof has a negative gain, then there has to be an atomic step that has negative gain. We can also consider constraints on pairs of metaproofs such as follows, which captures a preference over metaproofs with the same input and output that are shorter, where $P, P' \in \mathcal{P}$:

**C4** $G(P) > G(P') \Leftrightarrow \mathsf{In}(P) = \mathsf{In}(P')$ and $\mathsf{Out}(P) = \mathsf{Out}(P')$ and $\mathsf{Len}(P) < \mathsf{Len}(P')$

To define specific gain functions, we will introduce measures of a specific dimension of an argument map (each of which is a type of *map measure*, denoted by $\mathsf{Map}_\sigma$, where $\sigma$ is the type). We define six types below, and discuss further variants.

**Degree of entailment** For $\Phi \subseteq \mathcal{L}$, let $W(\Phi)$ be the set of interpretations of $\Phi$ delineated by the atoms used in $\Phi$ (i.e. $W(\Phi) = \wp(\mathsf{Atoms}(\Phi))$). For an argument $A$, let $\Delta = \mathsf{S}(A)$, and $\Gamma = \{\mathsf{C}(A)\}$, let $M(\Delta, \Gamma)$ be the set of models of $\Delta$ that are in $W(\Gamma)$. So $M(\Delta, \Gamma) = \{w \models \wedge\Delta \mid w \in W(\Gamma)\}$ where $\models$ is classical entailment. In the following definition, the degree of argument entailment for $A$, denoted $\mathsf{Arg}_{\mathsf{Ent}}(A)$, is the number of models that $\Delta$ and $\Gamma$ have in common divided by the number of models for $\Delta$. If $\Delta \vdash \bot$, then $\mathsf{Arg}_{\mathsf{Ent}}(A) = 1$, otherwise:

$$\mathsf{Arg}_{\mathsf{Ent}}(A) = \frac{|M(\Delta, \Delta \cup \Gamma) \cap M(\Gamma, \Delta \cup \Gamma)|}{|M(\Delta, \Delta \cup \Gamma)|}$$

**Example 13.** $\mathsf{Arg}_{\mathsf{Ent}}(\langle \{\mathtt{a}\}, \mathtt{a} \wedge \mathtt{b} \rangle) = 1/2$, $\mathsf{Arg}_{\mathsf{Ent}}(\langle \{\mathtt{a}\}, \mathtt{a} \wedge \mathtt{b} \wedge \mathtt{c} \rangle) = 1/4$, $\mathsf{Arg}_{\mathsf{Ent}}(\langle \{\mathtt{a}\}, \mathtt{a} \wedge \mathtt{b} \wedge \mathtt{c} \wedge \mathtt{d} \rangle) = 1/8$, $\mathsf{Arg}_{\mathsf{Ent}}(\langle \{\mathtt{a} \wedge \mathtt{b}\}, \mathtt{a} \vee \mathtt{b} \rangle) = 1$, $\mathsf{Arg}_{\mathsf{Ent}}(\langle \{\mathtt{a} \wedge \mathtt{b} \wedge \mathtt{c}\}, \mathtt{a} \wedge \mathtt{e} \rangle) = 1/2$, $\mathsf{Arg}_{\mathsf{Ent}}(\langle \{\mathtt{a} \wedge \mathtt{b}\}, \mathtt{a} \wedge \neg\mathtt{b} \rangle) = 0$.

For an argument map $M = (N, R, I)$, the degree of map entailment is the average of the degree of argument entailment: $\mathsf{Map}_{\mathsf{Ent}}(M) = 1/|N| \times (\sum_{n \in N} \mathsf{Arg}_{\mathsf{Ent}}(I(n)))$.

**Degree of consistency** For this, we can use an inconsistency measure. For instance, we can use the cardinality of the set of minimal inconsistent subsets of a set $\Delta$ of propositional formulae denoted $|\mathsf{MinInc}(\Delta)|$. The degree of argument consistency, denoted $\mathsf{Arg}_{\mathsf{Con}}(A)$, is defined below, where $n$ is the cardinality of $\Delta$ and $\binom{n}{\lfloor n/2 \rfloor}$ is the upper bound on $|\mathsf{MinInc}(\Delta)|$ as it is the maximum number of subsets of $\Delta$ for which none is a subset of another (Sperner's theorem). If $\mathsf{S}(A) = \emptyset$, then $\mathsf{Arg}_{\mathsf{Con}}(A) = 1$, otherwise:

$$\mathsf{Arg}_{\mathsf{Con}}(A) = 1 - \frac{|\mathsf{MinInc}(\mathsf{S}(A))|}{\binom{n}{\lfloor n/2 \rfloor}}$$

**Example 14.** *Consider argument* $\mathtt{A}$ *where* $\mathsf{S}(\mathtt{A}) = \{\mathtt{a} \vee \mathtt{b}, \neg\mathtt{a}, \neg\mathtt{b} \wedge \mathtt{c}, \neg\mathtt{c}, \mathtt{d}, \mathtt{e}\}$. *Since* $\mathsf{MinInc}(\mathsf{S}(\mathtt{A})) = \{\{\mathtt{a} \vee \mathtt{b}, \neg\mathtt{a}, \neg\mathtt{b} \wedge \mathtt{c}\}, \{\neg\mathtt{b} \wedge \mathtt{c}, \neg\mathtt{c}\}\}$, *and* $\binom{6}{3} = 20$, $\mathsf{Arg}_{\mathsf{Con}}(\mathtt{A}) = 1 - (2/20) = 9/10$.

For an argument map $M = (N, R, I)$, the degree of map consistency is the average of the degree of argument consistency: $\mathsf{Map}_{\mathsf{Con}}(M) = 1/|N| \times (\sum_{n \in N} \mathsf{Arg}_{\mathsf{Con}}(I(n)))$.

The degree of argument consistency is the complement of its degree of inconsistency. There are various options for measuring inconsistency in a set of formulae (for a review, see (Thimm 2016)) with counting the number of minimal inconsistent subsets being the basis of many of them. We use the normalized version to give a value in the unit interval (Besnard and Grant 2020).

**Degree of minimality** For an argument $A = \langle \Phi, \alpha \rangle$, $\mathsf{MinProof}(\Phi, \alpha)$ is $\{\Psi \subseteq \Phi \mid \Psi \vdash \alpha$ and there is not a $\Psi' \subset \Psi$ s.t. $\Psi' \vdash \alpha\}$ and let $\mathsf{Nec}(A)$ be the following set:

$$\mathsf{Max}(\{|\Psi| \mid \Psi \in \mathsf{MinProof}(\mathsf{S}(A), \mathsf{C}(A))\})$$

For argument $A$, we define the degree of argument minimality, denoted $\mathsf{Arg}_{\mathsf{Min}}(A)$, as follows: If $A$ is not valid, then $\mathsf{Arg}_{\mathsf{Min}}(A) = 0$, otherwise:

$$\mathsf{Arg}_{\mathsf{Min}}(A) = 1 - \frac{|\mathsf{S}(A)| - \mathsf{Nec}(A)}{|\mathsf{S}(A)|}$$

Essentially, $\mathsf{Arg}_{\mathsf{Min}}$ is a measure of the proportion of the support of an argument that is necessary for the entailment.

**Example 15.** *Let* $\mathtt{A} = \langle \{\neg\mathtt{a}, \mathtt{a} \vee \mathtt{b}, \mathtt{b} \to \mathtt{c}, \neg\mathtt{a} \vee \mathtt{b} \to \mathtt{c}\}, \mathtt{c} \rangle$. *So* $|\mathsf{S}(\mathtt{A})| = 4$ *and* $\mathsf{Nec}(\mathtt{A}) = 3$. *Hence,* $\mathsf{Arg}_{\mathsf{Min}}(\mathtt{A}) = 2/3$.

For an argument map $M = (N, R, I)$, the degree of map entailment is the average of the degree of argument entailment: $\mathsf{Map}_{\mathsf{Min}}(M) = 1/|N| \times (\sum_{n \in N} \mathsf{Arg}_{\mathsf{Min}}(I(n)))$.

**Degree of agreement** This is intended to capture the degree of agreement between the expected relation and the logical relation, and so we use the expected relation $R$ as what is required (i.e. the gold standard). Let $M = (N, R, I)$ be an argument map and $n$ the cardinality of $N$. The degree of agreement of $M$ is defined as follows: If $N = \emptyset$, then $\mathsf{Map}_{\mathsf{Agr}}(M) = 1$, otherwise:

$$\mathsf{Map}_{\mathsf{Agr}}(M) = \frac{|\{(x, y) \in N^2 \mid R(x, y) = \mathsf{LR}^M(x, y)\}|}{n^2}$$

**Example 16.** *For Figure 2, for the top (respectively bottom) argument map* $M_1$ *(respectively* $M_2$*),* $\mathsf{Map}_{\mathsf{Agr}}(M_1) = 7/9$ *and* $\mathsf{Map}_{\mathsf{Agr}}(M_2) = 1$.

**Degree of detail** With the degree of detail, $\mu$ denotes the desired number of formulae in the support of each argument. For any $x \in \mathbb{R}$, let $\mathsf{absolute}(x)$ be the absolute value of $x$. For argument $A$, the degree of argument detail, denoted $\mathsf{Arg}_{\mathsf{Det}^\mu}(A)$, is defined below:

$$\mathsf{Arg}_{\mathsf{Det}^\mu}(A) = \frac{1}{\mathsf{absolute}(|\mathsf{S}(A)| - \mu) + 1}$$

When $|\mathsf{S}(A)| = \mu$, $\mathsf{Arg}_{\mathsf{Det}^\mu}(A)$ is 1, and as the difference between $|\mathsf{S}(A)|$ and $\mu$ increases, $\mathsf{Arg}_{\mathsf{Det}^\mu}(A)$ approaches 0.

**Example 17.** *Let* $\mathsf{S}(\mathtt{A}) = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{a} \wedge \mathtt{b} \wedge \mathtt{c} \to \mathtt{d}, \neg\mathtt{d} \vee \mathtt{e}, \neg\mathtt{f} \to \neg\mathtt{e}, \neg(\mathtt{g} \wedge \mathtt{h})\}$, *with* $\mu = 3$. *So* $\mathsf{Arg}_{\mathsf{Det}^\mu}(\mathtt{A}) = 1/5$.

For an argument map $M = (N, R, I)$, the degree of map detail is the average of the degree of argument detail: $\mathsf{Map}_{\mathsf{Det}^\mu}(M) = 1/|N| \times (\sum_{n \in N} \mathsf{Arg}_{\mathsf{Det}^\mu}(I(n)))$.

We could define variants of the degree of argument detail such as a measure that would have the value 1 when the number of premises below or equal to $\mu$, otherwise the degree decreases as the number of premises increases.

**Degree of context** This measure aims to ensure that the claim of an argument is elucidated in a way that is relevant to the arguments attacked or supported by that argument. For an argument map $M = (N, R, I)$ and an argument $A$, the degree of context is defined as follows:

$$\mathsf{Arg}_{\mathsf{Ctx}}(A) = \begin{cases} 1 & \text{if } A \text{ attacks/supports nothing;} \\ \frac{\tau_1}{\tau_2} & \text{otherwise,} \end{cases}$$

where $\tau_1$ is the number of atoms that appear in the claim of $A$ and in at least one argument attacked or supported by $A$ (i.e. $\tau_1 = |\{a \in \mathsf{Atoms}(\mathsf{C}(A)) \mid$ there is an $n \in N$ s.t. $A$ supports or attacks $I(n)$ and $a \in \mathsf{Atoms}(\mathsf{S}(I(n)))\}|$), and $\tau_2$ is the number of atoms appearing in the claim of $A$ (i.e. $\tau_2 = |\mathsf{Atoms}(\mathsf{C}(A))|$). This degree is maximal when all the atoms appearing in the claim also appear in a formula in the premise of an argument attacked or supported by it.

**Example 18.** *Let* $\mathtt{a}$ = *"John has an umbrella",* $\mathtt{b}$ = *"It rains", and* $\mathtt{c}$ = *"John travels light" be atoms. Let* $\mathtt{A}$ *be the argument* $\langle \{\neg\mathtt{a}\}, \mathtt{c} \rangle$, *let* $\mathtt{B}$ *be the argument* $\langle \{\mathtt{b}\}, \mathtt{a} \rangle$, *and let* $\mathtt{C}$ *be the argument* $\langle \{\mathtt{b}, \mathtt{b} \to \mathtt{a}\}, \mathtt{a} \rangle$ *that is an augmentation of* $\mathtt{B}$. *Both* $\mathtt{B}$ *and* $\mathtt{C}$ *attack* $\mathtt{A}$ *but not vice versa. Therefore,* $\mathsf{Arg}_{\mathsf{Ctx}}(\mathtt{A}) = \mathsf{Arg}_{\mathsf{Ctx}}(\mathtt{B}) = \mathsf{Arg}_{\mathsf{Ctx}}(\mathtt{C}) = 1$.

We define the degree of context of an argument map $M$ as: $\mathsf{Map}_{\mathsf{Ctx}}(M) = 1/|N| \times (\sum_{n \in N} \mathsf{Arg}_{\mathsf{Ctx}}(I(n)))$.

**Example 19.** *Continuing Example 18, let* $M$ *be the argument map consisting of* $\mathtt{A}$ *and* $\mathtt{C}$, *and so* $\mathsf{Map}_{\mathsf{Ctx}}(M) = 1$. *Now let* $\mathtt{d}$ = *"Snails go out" be an atom and* $\mathtt{D}$ *be the argument* $\langle \{\mathtt{b}, \mathtt{b} \to \mathtt{a}, \mathtt{b} \to \mathtt{d}\}, \mathtt{a} \wedge \mathtt{d} \rangle$. *Let* $M'$ *be the argument map consisting of* $\mathtt{A}$ *and* $\mathtt{D}$. *Then,* $\mathsf{Map}_{\mathsf{Ctx}}(M') = \frac{1+0.5}{2} = 0.75$. *Hence, given argument* $\mathtt{A}$ *and the counter-argument* $\mathtt{B}$ *(which are enthymemes), our gain function prefers the elucidation "It rains and John always takes an umbrella when it rains, thus John has one" (i.e.* $\mathtt{C}$*) over the one that also speaks about snails (i.e.* $\mathtt{D}$*).*

The degree of agreement guarantees a connection between an argument and its target (contradiction in case of attack, entailment in case of support). The degree of context ensures that this connection is minimal. This is analogous to that between the degrees of entailment and minimality.

**General comments** All the map measures $\mathsf{Map}_\sigma$ that we have considered above are in the unit interval, with 0 denoting the worst case and 1 denoting the best case.

**Proposition 7.** *For each* $M \in \mathcal{M}$, *and for each* $\sigma \in \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}, \mathsf{Ctx}, \mathsf{Det}^\mu\}$, $\mathsf{Map}_\sigma(M) \in [0, 1]$.

| Step | Map | Metarule | Ent | Con | Min | Agr | Ctx | Average | $G^{\Pi}_{\mathsf{av}}$ |
|---|---|---|---|---|---|---|---|---|---|
| 1 | $\langle\{\}, \mathtt{a} \land \mathtt{b}\rangle \xrightarrow{-} \langle\{\}, \mathtt{c}\rangle$ | - | 0.375 | 1 | 0 | 0.75 | 0.5 | 0.525 | - |
| 2 | $\langle\{\}, \mathtt{a} \land \mathtt{b}\rangle \xrightarrow{-} \langle\{\mathtt{d}, \mathtt{d} \to \mathtt{c}\}, \mathtt{c}\rangle$ | Augment | 0.625 | 1 | 0.5 | 0.75 | 0.5 | 0.675 | 0.15 |
| 3 | $\langle\{\mathtt{a}, \mathtt{a} \to \mathtt{b}\}, \mathtt{a} \land \mathtt{b}\rangle \xrightarrow{-} \langle\{\mathtt{d}, \mathtt{d} \to \mathtt{c}\}, \mathtt{c}\rangle$ | Augment | 1 | 1 | 1 | 0.75 | 0.5 | 0.85 | 0.175 |
| 4 | $\langle\{\mathtt{a}, \mathtt{a} \to \mathtt{b}\}, \mathtt{a} \land \mathtt{b}\rangle \xrightarrow{-} \langle\{\}, \mathtt{c}\rangle$ | Shrink | 0.75 | 1 | 0.5 | 0.75 | 0.5 | 0.7 | -0.15 |
| 5 | $\langle\{\mathtt{a}, \mathtt{a} \to \mathtt{b}\}, \mathtt{a} \land \mathtt{b}\rangle \xrightarrow{-} \langle\{\neg\mathtt{b}, \mathtt{b} \lor \mathtt{c}\}, \mathtt{c}\rangle$ | Augment | 1 | 1 | 1 | 1 | 0.75 | 0.95 | 0.25 |
| 6 | $\langle\{\mathtt{a}, \mathtt{a} \to \mathtt{b}\}, \mathtt{b}\rangle \xrightarrow{-} \langle\{\neg\mathtt{b}, \mathtt{b} \lor \mathtt{c}\}, \mathtt{c}\rangle$ | Weaken | 1 | 1 | 1 | 1 | 1 | 1 | 0.05 |

Table 2: Example for calculating average gain function for map measures $\Pi = \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}, \mathsf{Ctx}\}$ and decoding knowledgebase $K = \{\mathtt{a}, \mathtt{a} \to \mathtt{b}, \neg\mathtt{b}, \mathtt{b} \lor \mathtt{c}, \mathtt{d}, \mathtt{d} \to \mathtt{c}\}$. Each argument map contains two arguments where the left argument is attacking the right argument. The metarule used to obtain the argument map from its previous step is given for steps 2 to 6. The metaproof first elucidates the right argument, and then the left argument. But then the left argument fails to attack the right argument. So the shrink step is a backtrack step allowing for an alternative elucidation of the right argument. With this second elucidation, the left argument does attack the right argument, and so the degree of agreement increases. In the final step, the claim is weakened thereby increasing the degree of context. The average column is the average of the five columns to its left. The gain value is the change in the average between the current step and its previous step.

We have just given a few options for map measures. There are some interesting further options. For instance, splitting degree of agreement so that we separately measure the number of expected attacks that are logical attacks, the number of logical attacks that are not expected attacks, etc, and similarly for supports (Amgoud and David 2018). Other options include evaluating the complexity of formulae in premises.

**Average gain function** We assume a set of map measures $\Pi$ s.t. for each $\sigma \in \Pi$, and for each $P \in \mathcal{P}$, $\mathsf{Map}_\sigma(P) \in [0, 1]$. For instance, we may let $\Pi$ be a particular subset of $\{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}, \mathsf{Ctx}, \mathsf{Det}^\mu\}$. For a metaproof $P$, let $\mathsf{In}(P) = M_1 = (M_1, R_1, I_1)$, $\mathsf{Out}(P) = M_2 = (M_2, R_2, I_2)$. The **average gain function**, denoted $G^{\Pi}_{\mathsf{av}}$, is defined as follows: $G^{\Pi}_{\mathsf{av}}(P) =$

$$(\frac{1}{|\Pi|} \times \sum_{\sigma\in\Pi} \mathsf{Map}_\sigma(M_2)) - (\frac{1}{|\Pi|} \times \sum_{\sigma\in\Pi} \mathsf{Map}_\sigma(M_1))$$

For example, for Figures 1 and 2, each metaproofs $P$ is such that $G^{\Pi}_{\mathsf{av}}(P) > 0$ when $\Pi = \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}\}$. Note, the average gain function is only one option for aggregating the values from a set of map measures.

**Proposition 8.** *The average gain function is a gain function (i.e. satisfies Definition 5) that returns a value in* $[-1, 1]$. *If* $\Pi \subseteq \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}, \mathsf{Ctx}, \mathsf{Det}^\mu\}$, *then it satisfies constraints 2 and 3, but not constraints 1 or 4.*

We illustrate the use of the average gain function in the example in Table 2 for $\Pi = \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}, \mathsf{Ctx}\}$, and in Table 3 with also $\mathsf{Det}^\mu$. In these examples, we see that there is backtracking (i.e. moves reversed). In particular, an augment is reversed by a shrink move, with a reversing of the gain, which is generalized in the following result.

**Proposition 9.** *Let* $\Pi \subseteq \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}, \mathsf{Ctx}, \mathsf{Det}^\mu\}$. *If metaproofs* $P = ((M_1, M_2), K)$ *and* $P' = ((M_2, M_1), K)$, *then* $G^{\Pi}_{\mathsf{av}}(P) = -G^{\Pi}_{\mathsf{av}}(P')$.

In practice, we are using the gain function to direct the search for a better elucidation in a space of elucidations. For instance in Table 2, we see how steps 1, 2, and 3 improve

| Map | Ent | Con | Min | Agr | Ctx | $\mathsf{Det}^\mu$ | $G^{\Pi}_{\mathsf{av}}$ |
|---|---|---|---|---|---|---|---|
| $M_1$ | 0.5 | 1 | 0 | 1 | 1 | 0.5 | - |
| $M_2$ | 1 | 1 | 1 | 1 | 1 | 0.25 | 0.208 |
| $M_3$ | 0.5 | 1 | 0 | 1 | 1 | 0.5 | -0.208 |
| $M_4$ | 1 | 1 | 1 | 1 | 1 | 1 | 0.333 |

Table 3: Let $K = \{\mathtt{a} \to \mathtt{b}, \mathtt{c}, \mathtt{c} \to \mathtt{d}, \mathtt{b} \land \mathtt{d} \to \mathtt{e}, \neg\mathtt{a} \lor \mathtt{e}\}$ and $\Pi = \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}, \mathsf{Ctx}, \mathsf{Det}^\mu\}$ where $\mu = 2$. Consider the argument map $M_1$ containing just the enthymeme $\langle\{\mathtt{a}\}, \mathtt{e}\rangle$. Let $M_2$ be obtained by augmenting $\langle\{\mathtt{a}, \mathtt{a} \to \mathtt{b}, \mathtt{c}, \mathtt{c} \to \mathtt{d}, \mathtt{b} \land \mathtt{d} \to \mathtt{e}\}, \mathtt{e}\rangle$. This increases the degree of entailment but reduces the degree of detail. Let $M_3$ be the result of backtracking to $\langle\{\mathtt{a}\}, \mathtt{e}\rangle$, and let $M_4$ be an alternative augmentation $\langle\{\mathtt{a}, \neg\mathtt{a} \lor \mathtt{e}\}, \mathtt{e}\rangle$ that improves the degree of detail. The gain value is the change in the average between the current step and its previous step

the gain, and then we need to backtrack at step 4, and then proceed with steps 5 and 6 to obtain the optimal solution. If we were to look ahead, and thereby compare alternative sequences of steps in order to identify those that only have positive gain for each step, we could have for instance identified the sequence of steps 1, 4, 5, and 6.

## 7 Directed Elucidation

The aim of a metaproof is to improve the quality of the argument map by making arguments explicit and clearer. Next, we provide some options for defining this.

**Definition 6.** *For argument maps* $M_1 = (N, R, I_1)$ *and* $M_2 = (N, R, I_2)$, $M_2$ *is a* $K$**-decoding** *of* $M_1$ *w.r.t. decoding knowledgebase* $K$ *iff for all* $n \in N$, $I_2(n) \setminus I_1(n) \subseteq K$.

**Definition 7.** *For argument maps* $M_1 = (N, R, I_1)$ *and* $M_2 = (N, R, I_2)$, $M_2$ *is a* **complete elucidation** *of* $M_1$ *w.r.t. decoding knowledgebase* $K$ *iff* $M_2$ *is a K-decoding of* $M_1$ *and* $R = \mathsf{LR}^M_2$ *and for all* $n \in N$, $I_2(n)$ *is a strict argument.*

Next, we show that when the input and output argument maps are isomorphic, and $\Pi = \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}\}$, then the average gain function ensures that the enthymemes are elucidated as strict arguments, and the structure of the input argument map is respected.

**Proposition 10.** *For argument maps $M$ and $M'$, $M'$ is a complete elucidation of $M$ w.r.t. decoding knowledgebase $K$ iff $M'$ is a $K$-decoding of $M$ and $\Pi = \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}\}$ and $(\frac{1}{|\Pi|} \times \sum_{\sigma \in \Pi} \mathsf{Map}_\sigma(M')) = 1$.*

Since a complete elucidation may be regarded as capturing some key requirements for an ideal elucidation of an argument map, the above result shows that we can capture this when the degrees of entailment, consistency, minimality, and agreement, are all 1.

**Theorem 1.** *For argument maps $M = (N, R, I_1)$ and $M_2 = (N, R, I_2)$, $M'$ is a complete elucidation of $M$ w.r.t. decoding knowledgebase $K$ iff there is a finite $P \in \mathcal{P}^K$ s.t. $M'$ is obtained from $M$ using $P$ and for all $P' \in \mathcal{P}^K$ s.t. $M''$ is obtained from $M$ using $P''$, and for $\Pi = \{\mathsf{Ent}, \mathsf{Con}, \mathsf{Min}, \mathsf{Agr}\}$, $G^{\Pi}_{\mathsf{av}}(P'') \leq G^{\Pi}_{\mathsf{av}}(P)$.*

When there is not a complete elucidation, then we need to find the best result as a Pareto optimal solution as defined below where condition 1 ensures that the output is a solution and condition 2 ensures that there is no better solution.

**Definition 8.** *For argument maps $M_1$ and $M_2$, $M_2$ is an* **optimal elucidation** *of $M_1$ w.r.t. decoding knowledgebase $K$ and map measures $\Pi$ iff (1) $M_2$ is a $K$-decoding of $M_1$ and (2) there is no argument map $M_3$ s.t. $M_3$ is a $K$-decoding of $M_1$ and there is a $\sigma \in \Pi$ for which $(\mathsf{Map}_\sigma(M_3) - \mathsf{Map}_\sigma(M_2)) > 0$ and for all $\sigma \in \Pi$, $(\mathsf{Map}_\sigma(M_3) - \mathsf{Map}_\sigma(M_2)) \geq 0$.*

So given a decoding knowledgebase, and a set of map measures, an optimal elucidation of an argument map $M_1$ is an argument map $M_2$ for which there are no further elucidation steps that can improve it (i.e. there are no steps that can increase one map measures without decreasing another). Therefore, as shown next, if a metaproof cannot be extended to improve its output argument map, then the output argument map is an optimal elucidation.

**Theorem 2.** *For a metaproof $P \in \mathcal{P}^K$, where $\mathsf{In}(P) = M_1$ and $\mathsf{Out}(P) = M_2$, and map measures $\Pi$, if $G^{\Pi}_{\mathsf{av}}(P) \geq 0$ and there is no metaproof $P \oplus P' \in \mathcal{P}^K$ s.t. $G^{\Pi}_{\mathsf{av}}(P') > 0$, then $M_2$ is an optimal elucidation of $M_1$ w.r.t. $K$ and $\Pi$.*

The advantage of Theorem 2 is that we can obtain an optimal elucidation by a metaproof. Moreover, rather than randomly search for the optimal solutions, we can use proof steps, with the steering of the gain function, to find the optimal solutions in a goal-directed way.

## 8 Derived Proof Rules

A set of natural deduction proof rules for classical logic does not include all the natural proof rules that are valid for classical logic. In other words, from a finite set of natural deduction rules for classical logic, there are further proof rules that can be derived from the set. We have the same situation with our meta-rules. Here we consider the following derived meta-rules that are valid. They capture restrictions on the preconditions and/or they capture multiple steps in one rule. We assume each derived proof rule can be used directly as a meta-rule in a metaproof.

**Flip proof rule.** An argument can present a claim based on some evidence in the premises. For example, consider $\mathtt{s}$ = *"John is sneezing"*, and $\mathtt{c}$ = *"John has a cold"*. Initially, we may consider this as enthymeme $\langle\{\mathtt{s}\}, \mathtt{c}\rangle$. However, when elucidating this enthymeme, we may wish to understand it as a causal relationship in the form of the argument $\langle\{\mathtt{c}\}, \mathtt{s}\rangle$. For an argument map $M = (N, R, I)$, we can capture this as the following derived proof rule where the precondition (resp. postcondition) is above (resp. below) the line.

$$\frac{I(n) = \langle\{\alpha\}, \beta\rangle \text{ where } n \in N}{I(n) = \langle\{\beta\}, \alpha\rangle}$$

**Refinement proof rule.** The aim of refinement is to change the premises in order to get the right level of granularity for the user. For example, consider the atoms $\mathtt{j}$ = *"John has a parrot"*, and $\mathtt{c}$ = *"John has good company"*. Also, consider the argument $\langle\{\mathtt{j}, \mathtt{j} \to \mathtt{c}\}, \mathtt{c}\rangle$. For some audiences, an argument map containing this argument would need elucidating by refining the argument as follows. Let $\mathtt{p}$ = *parrots are intelligent animals that are good company for their owners*. Now, with the formulae $\mathtt{p}$ and $\mathtt{j} \wedge \mathtt{p} \to \mathtt{c}$, we can refine the argument as $\langle\{\mathtt{j}, \mathtt{j} \wedge \mathtt{p} \to \mathtt{c}\}, \mathtt{c}\rangle$. We assume a preference ordering $\preceq$ over subsets of the knowledgebase $K$ where for $\Phi, \Psi \subseteq K$, $\Phi \preceq \Psi$ denotes that $\Phi$ is preferred to $\Psi$ as premises for an argument. Now we can capture this refinement as the following derived proof rule.

$$\frac{I(n) = \langle\Psi, \alpha\rangle \text{ where } n \in N \text{ and } \Phi \preceq \Psi \text{ and } \Phi \vdash \alpha}{I(n) = \langle\Phi, \alpha\rangle}$$

For the above derived proof rule, we are assuming extra information about the decoding knowledgebase, and therefore make the metaproof more constrained. Whilst specifying $\preceq$ over all subsets of $K$ would be impractical except for small decoding knowledgebases, there may be efficient heuristics that could be used to generate this ordering using, for example, a ranking over propositional letters.

We formally define a **derived proof rule** as a proof rule that satisfies the following: For all argument maps $M$, if $M$ satisfies the precondition of the proof rule w.r.t. decoding knowledgebase $K$, and $M'$ is the result of applying the postcondition of the proof rule to $M$, then using the meta-level rules in Table 1, there is a metaproof $P = (X, K)$ s.t. $\mathsf{In}(P) = M$ and $\mathsf{Out}(P) = M'$. By this definition, each of the meta-rules in Table 1 is trivially a derived proof rule.

**Proposition 11.** *The flip proof rule and the refinement proof rule are derived proof rules.*

Advantages of using derived proof rules include: (1) we can add restrictions to the preconditions such as about the decoding knowledgebase; (2) we can reduce multiple proof step into a single step as in the flip proof rule; and (3) we can define derived rules that have positive gain (depending on the choice of gain function) for a single step but that wouldn't have positive gain for each step if we used the original set of meta-level rules. To harness these advantages, we can restrict metaproofs to specific derived proof rules (as we did with meta-level rules in Propositions 4 to 6).

## 9 Discussion

In this paper, we make the following contributions: (1) Characterization of the elucidation of enthymemes in propositional logic; (2) Presentation of meta-level rules for manipulating arguments in propositional logic; and (3) Presentation of gain as a way of directing the use of meta-level rules. Our proposal provides a clearer understanding of the process of elucidation of enthymemes in argument maps, and it provides tools to support this process.

When applied to argument maps generated from natural language argumentation, a metaproof is an explicit and therefore explainable derivation of the elucidation. The user can see exactly how the output argument map has been obtained from the input argument map, and this may facilitate analysis, user-generated corrections, and also improved user confidence in the process. We therefore see the proposal in this paper being used to automate the process of elucidation, possibly with a human-in-the-loop to steer elucidation. We also see that the elucidation process could learn from the human-in-the-loop by refining the gain functions to provide the type of elucidations required by the user.

This paper complements evaluation of the quality of each candidate for a decoding of an enthymeme on a pairwise basis (Ben-Naim, David, and Hunter 2025). Different quality criteria are considered, including inference, coherence, similarity between formulae in the enthymeme and in the candidate. These could provide alternative options for defining gain functions in our proposal. However, no consideration is given to optimization of decoding, and it only considers individual enthymemes, and so it does not consider argument maps. In another proposal, optimization of decoding of enthymemes in argument maps is considered but only with respect to what we refer to as agreement (David and Hunter 2025). Furthermore, no consideration is given in either of the above cited papers as to how decoding candidates are obtained (such as logical operations to obtain candidates, or measures of them).

We focus on propositional logic in this paper to better understand the notion of elucidation. However, key aspects of our paper (including the meta-level rules and some measures of gain) apply to first-order predicate logic.

In our proposal, the relationship (i.e. attack, neutral, or support) that one argument has with another is dictated by the logical instantiation. In contrast, in (Corsi 2024), arguments can be manipulated in terms of their logical structure. For example, if argument $X$ attacks an argument $A \wedge B$, then $X$ attacks $A$ and $X$ attacks $B$. Whilst, the proposal does not deal with enthymemes, it would be interesting to investigate how that approach could be harnessed in our proposal for elucidating enthymemes in argument maps.

In future work, we will investigate further proof rules, further measures of gain (new map measures and alternatives to $G^{\Pi}_{\mathsf{av}}$), and alternative notions of optimal elucidations, including merging similar arguments (based on similarity measures (Amgoud and David 2018)). We will also investigate automated reasoning for implementing the derivation of metaproofs, and evaluate the approach with real-data such as the microtexts corpus (Peldszus and Stede 2016; Skeppstedt, Peldszus, and Stede 2018).

## AI Declaration

The authors have not employed any Generative AI tools.

## Acknowledgements

The work by Victor David was supported by the French government, managed by the Agence Nationale de la Recherche under the Plan d'Investissement France 2030, as part of the Initiative d'Excellence d'Université Côte d'Azur under the reference ANR-15-IDEX-01. Anthony Hunter is grateful for travel funding from the Université Côte d'Azur to facilitate collaboration on this research.

## References

Amgoud, L., and David, V. 2018. Measuring similarity between logical arguments. In *Proceedings of KR'18*, 98–107. AAAI Press.

Atkinson, K.; Baroni, P.; Giacomin, M.; Hunter, A.; Prakken, H.; Reed, C.; Simari, G. R.; Thimm, M.; and Villata, S. 2017. Towards artificial argumentation. *AI Magazine* 38(3):25–36.

Ben-Naim, J.; David, V.; and Hunter, A. 2025. An axiomatic study of a modular evaluation of enthymeme decoding in weighted structured argumentation. In *Proceedings of KR'25*. IJCAI Organization.

Ben-Naim, J.; David, V.; and Hunter, A. 2026. Elucidating arguments maps in propositional logic: Proof appendix. http://www0.cs.ucl.ac.uk/staff/A.Hunter/papers/kr26appendix.pdf.

Besnard, P., and Grant, J. 2020. Relative inconsistency measures. *Artificial Intelligence* 280:103231.

Besnard, P., and Hunter, A. 2001. A logic-based theory of deductive arguments. *Artificial Intelligence* 128(1-2):203–235.

Black, E., and Hunter, A. 2008. Using enthymemes in an inquiry dialogue system. In *Proceedings of AAMAS'08*, 437–444. IFAAMAS.

Black, E., and Hunter, A. 2012. A relevance-theoretic framework for constructing and deconstructing enthymemes. *Journal of Logic and Computation.* 22(1):55–78.

Cayrol, C. 1995. On the relation between argumentation and non-monotonic coherence-based entailment. In *Proceedings of IJCAI'95*, 1443–1448. Morgan Kaufmann.

Corsi, E. A. 2024. Attack principles in sequent-based argumentation theory. *Journal of Logic and Computation* 35(3):exad080.

David, V., and Hunter, A. 2025. A logic-based framework for decoding enthymemes in argument maps involving implicitness in premises and claims. In *Proceedings of IJCAI'25*. IJCAI Organization.

de Saint-Cyr, F. D. 2011a. A first attempt to allow enthymemes in persuasion dialogs. In Morvan, F.; Tjoa, A. M.; and Wagner, R. R., eds., *Proceedings of DEXA'11*, 332–336. IEEE Computer Society.

de Saint-Cyr, F. D. 2011b. Handling enthymemes in time-limited persuasion dialogs. In *Proceedings of SUM'11*, volume 6929 of *LNCS*, 149–162. Springer.

Feng, X., and Hunter, A. 2025. Formalizing simple natural language arguments using abstract meaning representation and approximate propositional reasoning. In *Proceedings of the IEEE Conference on Tools with Artificial Intelligence ICTAI'25*, 238–245. IEEE Press.

Hosseini, S. A.; Modgil, S.; and Rodrigues, O. 2014. Enthymeme construction in dialogues using shared knowledge. In *Proc. of COMMA'14*, volume 266 of *FAIA*, 325–332. IOS Press.

Hunter, A. 2007. Real arguments are approximate arguments. In *Proceedings of AAAI'07*, 66–71. MIT Press.

Hunter, A. 2022. Understanding enthymemes in deductive argumentation using semantic distance measures. In *Proceedings of AAAI'22*, 5729–5736. AAAI Press.

Lalwani, A.; Chopra, L.; Hahn, C.; Trippel, C.; Jin, Z.; and Sachan, M. 2024. NL2FOL: translating natural language to first-order logic for logical fallacy detection. *CoRR* abs/2405.02318.

Leiva, D. S. O.; Gottifredi, S.; and García, A. J. 2023. Automatic knowledge generation for a persuasion dialogue system with enthymemes. *International Journal of Approximate Reasoning* 160:108963.

Liu, J. 2025. Few-shot natural language to first-order logic translation via code generation. In *Proceedings of NAACL'25*, 10939–10960. Association for Computational Linguistics.

Lu, X.; Liu, J.; Gu, Z.; Tong, H.; Xie, C.; Huang, J.; Xiao, Y.; and Wang, W. 2022. Parsing natural language into propositional and first-order logic with dual reinforcement learning. In *Proceedings of COLING'22*, 5419–5431. International Committee on Computational Linguistics.

Olausson, T.; Gu, A.; Lipkin, B.; Zhang, C. E.; Solar-Lezama, A.; Tenenbaum, J. B.; and Levy, R. 2023. LINC: A neurosymbolic approach for logical reasoning by combining language models with first-order logic provers. In *Proceedings of EMNLP'23*, 5153–5176. Association for Computational Linguistics.

Pan, L.; Albalak, A.; Wang, X.; and Wang, W. 2023. Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning. In *Findings of EMNLP'23*, 3806–3824. Association for Computational Linguistics.

Panisson, A. R.; McBurney, P.; and Bordini, R. H. 2022. Towards an enthymeme-based communication framework in multi-agent systems. In *Proceedings of KR'22*. IJCAI Organization.

Peldszus, A., and Stede, M. 2016. An annotated corpus of argumentative microtexts. In *Proceedings of European Conference on Argumentation*. College Publications.

Sadeddine, Z., and Suchanek, F. M. 2025. Verifying the steps of deductive reasoning chains. In *Findings of ACL'25*, 456–475. Association for Computational Linguistics.

Skeppstedt, M.; Peldszus, A.; and Stede, M. 2018. More or less controlled elicitation of argumentative text: Enlarging a microtext corpus via crowdsourcing. In *Proceedings of Workshop in Argumentation Mining (at EMNLP'18)*. Association of Computational Linguistics.

Sviridova, E.; Cabrio, E.; and Villata, S. 2025. Mining implicit arguments for reasoning: A survey. *Argument and Computation* 17:3–27.

Thimm, M. 2016. On the expressivity of inconsistency measures. *Artifical Intelligence* 234:120–151.

Walton, D. 2001. Enthymemes, common knowledge and plausible inference. *Philosophy and Rhetoric* 34(2):93–112.

Xydis, A.; Hampson, C.; Modgil, S.; and Black, E. 2020. Enthymemes in dialogues. In *Proceedings of COMMA'20*, volume 326 of *FAIA*, 395–402. IOS Press.

Yang, Y.; Xiong, S.; Payani, A.; Shareghi, E.; and Fekri, F. 2024. Harnessing the power of large language models for natural language to first-order logic translation. In *Proceedings of ACL'24*, 6942–6959. Association for Computational Linguistics.