Epistemic Graphs for Representing and Reasoning with Positive and Negative Influences of Arguments*

Anthony Hunter and Sylwia Polberg†

University College London, London, United Kingdom

Matthias Thimm

University of Koblenz-Landau, Koblenz, Germany

Abstract

This paper introduces epistemic graphs as a generalization of the epistemic approach to probabilistic argumentation. In these graphs, an argument can be believed or disbelieved up to a given degree, thus providing a more fine–grained alternative to the standard Dung's approaches when it comes to determining the status of a given argument. Furthermore, the flexibility of the epistemic approach allows us to both model the rationale behind the existing semantics as well as completely deviate from them when required. Epistemic graphs can model both attack and support as well as relations that are neither support nor attack. The way other arguments influence a given argument is expressed by the epistemic constraints that can restrict the belief we have in an argument with a varying degree of specificity. The fact that we can specify the rules under which arguments should be evaluated and we can include constraints between unrelated arguments permits the framework to be more context–sensitive. It also allows for better modelling of imperfect agents, which can be important in multi–agent applications.

Keywords— abstract argumentation, epistemic argumentation, bipolar argumentation

*Funding: Sylwia Polberg and Anthony Hunter were supported by EPSRC Project EP/N008294/1 "Framework for Computational Persuasion". Matthias Thimm was partially supported by the Deutsche Forschungsgemeinschaft (grant KE 1686/3-1).

†Corresponding author. E–mail: sylwia.polberg at gmail.com. University College London, Department of Computer Science, 66-72 Gower Street, London WC1E, United Kingdom.

Contents

1 Introduction

In real-world situations, argumentation is pervaded by uncertainty. There are various kinds of uncertainty that we may need to consider. Of particular importance is the notion of belief in arguments, which can be conceptualized in a number of ways. In this paper, we focus on belief as the degree to which an argument is regarded as being acceptable, and by that we mean the degree of belief in the combination of the premises being true, and the claim following from those premises. Such modelling is useful both in monological and dialogical argumentation. In monological argumentation, we might be uncertain about how much we believe an argument and how much this belief should influence the belief in other arguments. These issues are compounded when considering dialogical argumentation, where each participant might be uncertain about what other agents believe. In addition, there are further notions important for successful argumentation, such as the ability to take contextual information into account, to handle different perspectives that various agents can have about a given issue, or to model agents that are not perfectly rational reasoners or about whom we do not possess complete information.

Our aim in this paper is to present a new formalism for argumentation that takes belief into account and tackles the aforementioned challenges. In order to make our investigation more focused, we assume that for any scenario, we may have a set of arguments, and some relationships between these arguments. We will treat the arguments as abstract in this paper, but we can instantiate each argument with a textual description (and we will have examples of such instantiations) or a logical specification (for example as a deductive argument [14]). We also assume that the arguments and relationships between them can be represented by a directed graph, and so each node denotes an argument, and each arc denotes a relationship between a pair of arguments. In addition, to these assumptions, we have some requirements for the formalism, which we present in the next subsection.

Requirements

In this paper, we will focus on the following requirements for our new formalism. We will briefly delineate them first, and then motivate them through examples and a discussion of how meeting the requirements is of use.

Modelling fine–grained acceptability Typical semantics for argumentation frameworks focus on judging whether an argument should be accepted or rejected. However, in practical applications, there might be uncertainty as to the degree an argument is accepted or rejected. Knowing how strongly a person agrees or disagrees with a given concept may in certain situation be more desirable than just knowing that s/he agrees or disagrees with it. Various studies, including [77, 71], show that a two–valued perspective may be insufficient for modelling people's beliefs about arguments. So the requirement is that we should have a many–valued scale for belief in arguments. Recent interest in ranking-based semantics and the notion of argument strength, also points to the need for this requirement (see [17] for an overview).

Modelling positive and negative relations between arguments The notions of attack and support are clearly important aspects of argumentation, even though the formalization of the interaction between these two types of relationship is open to multiple interpretation and subject to some debate in the research community [25, 19, 74, 67, 71, 21, 54, 79]. Nevertheless, there are various studies showing the importance of support in real argumentation, such as works on argument mining [21] or dialogical argumentation [71]. Furthermore, in decision–making, it is common to consider not only the arguments against a given option, but also the arguments in favour of this option [7, 38]. Hence, this requirement is that we need to model how the beliefs in arguments can have a positive or negative influence on other arguments, and that the belief in an argument needs to take into account those influences.

Modelling context–sensitivity If we assume an argumentation situation is represented by a directed graph, then the belief that an individual has in the arguments can depend on how the arguments are instantiated by textual descriptions and on the background of this individual. The actual content of the arguments and the problem domain can affect the way a given graph is evaluated [26]. Two different instantiations can be interpreted differently by a single user depending on his or her knowledge or preferences [84]. Furthermore, the influence each argument has on another can depend on the instantiation. Hence, this requirement is that the context (i.e. how arguments are instantiated) can affect the belief in arguments and their influence on other arguments.

Modelling different perspectives It is common for different people to perceive the same information in different ways. In argumentation, not only can a given graph be evaluated differently by different people, but also its structure might not be uniformly perceived [71]. So if we have an argumentation situation represented by a single directed graph, with each argument instantiated with a text description, different people might have different belief in the individual arguments and in the influence the belief in one argument has on other arguments. Partly, this difference may occur because the text description of an argument is often in the form of an enthymeme (i.e. an argument that only has some of its premises and claim represented explicitly), and so different people may decode the enthymemes differently [15]. This difference may also occur because of differing background knowledge and experience. So this requirement is that the participant (i.e. the agent judging the argument graph) can have belief in arguments and their influences that is different to other participants.

Modelling imperfect agents People can exhibit a number of imperfections such as errors in their background knowledge, errors in the way they analyze information, and biases in how they analyze information in general. So when judging an argumentation situation represented by a single directed graph, with each argument instantiated with a text description, some people might make inappropriate or irrational judgments. This irrationality could be seen in terms of not adhering to argumentation semantics as well as in terms of reasoning fallacies or undesirable cognitive biases [61]. Since we want our formalism to be useful for real-world applications, we need the ability to model the imperfect agents in their assessment of belief in arguments and in their assessment of the influence between arguments.

Modelling incomplete situations An argumentation graph might not contain all of the arguments relevant to a given problem, in particular those that concern the agent(s). For example, a patient might withhold a certain embarrassing or private piece of information from the doctor, despite the fact that it can affect the diagnosis. However, this incomplete knowledge might also be a result of how the graph is obtained or updated. In dialogical argumentation, depending on the used protocol, an agent might not always be able to put forward all arguments relevant to the discussion. As a result, an agent may, for example, disbelieve an argument that is perceived by us as unattacked, even though the agent is privately aware of reasons to doubt the argument. Similarly, an agent can believe an argument despite it being attacked, simply because the graph does not contain the agent's supporting arguments. Such a behaviour would violate the majority of the argumentation semantics available in the literature. Furthermore, graph incompleteness in combination with fine–grained acceptability means that we might know that an agent believes or disbelieves a particular argument, but cannot precisely state to what degree. We therefore need an approach that is more resilient to potential incompleteness of the possessed information.

In the following examples, we consider simple scenarios where we might use monological argumentation to make sense of a situation, and possibly to make decisions. The examples highlight the value of implementing the aforementioned requirements.

Example 1. Imagine we have two passengers on a train, Jack and Jill, travelling to work. Jack is using this particular connection regularly and has some experience with the vagaries of the service. Jill, however, uses this connection for the first time, and has an important meeting to attend and wants to be on time. Let us assume that the their knowledge concerning whether the train is going to be late is represented in Figures 1 and 2 and let us first focus on Jack.

We can observe that arguments `B`, `C` and `D` are enthymemes. Their claims are not explicit and can be decoded in a number of ways. Since Jack is a regular client of this service, it is reasonable to assume that the missing claim for `B` (and `C`) is "therefore the train will arrive a bit late". He also has a live travel info app that says that this service will arrive on time. He has been using this app for a while and does not consider

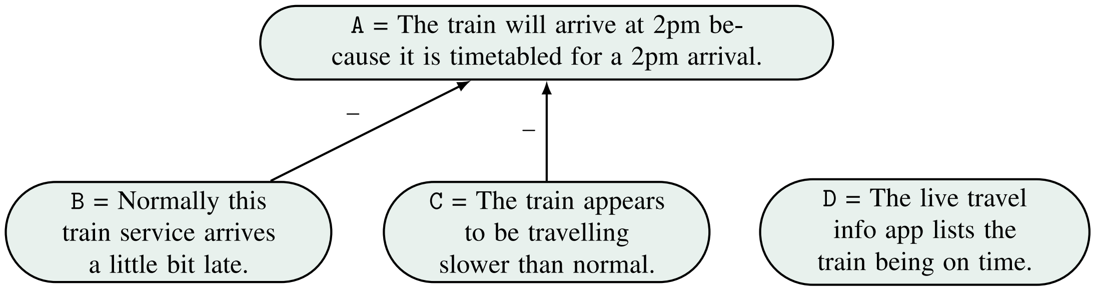


Figure 1: Jack's graph concerning the arrival time of a train journey. Edges labelled with − represent attack.

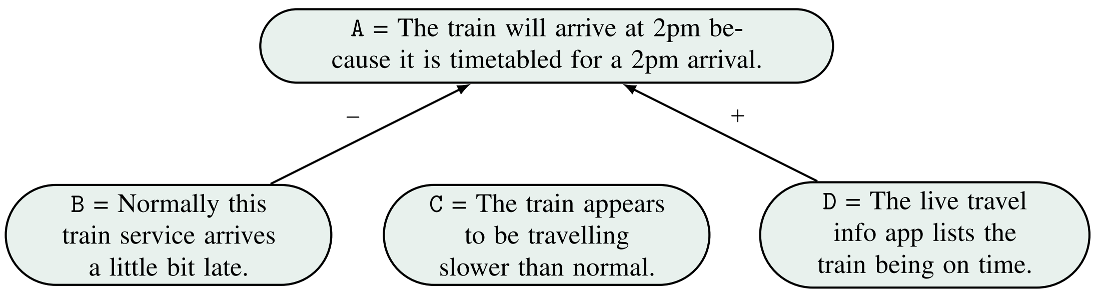


Figure 2: Jill's labelled graph concerning the arrival time of a train journey. Edges labelled with − represent attack and edges labelled with + represent support.

it reliable at all, and because of this experience he chooses to decode the claim of D as "the live travel info service predicts the train will be on time". So, Jack does not decode the claim of D as "therefore the train will arrive on time". Hence, he sees arguments B and C as attacking A and disregards the influence of D. Thus, Jack's belief in B and C suggests that A should be disbelieved, and the degree to which he disbelieves or believes C should be primarily affected by C, i.e. his current perception of the service. At the same time, he is certain of his eyes, i.e. that the info service predicts that the train will be on time, but his belief in that argument does not affect A.

Let us now focus on Jill, who is new to the service. She heard from a fellow passenger that this train normally arrives a little bit late and chooses to decode the claim of B as "therefore the train will arrive a bit late". This is the first time she has used this particular service, as she had only recently moved from a different town. She commuted by train before, but the connection she used from her previous town was a faster one. Therefore, she sees argument C as a comment on the new line when compared to the line she used before, not as a sign of problems happening right now on the train she has boarded. Thus, her claim for C is "therefore the tracks on this line must be in a worse condition than on the other line" and for her, arguments A and C are not particularly related. Finally, the live travel app she has been using has been very reliable in the past and she trusts it. She decodes the claim of D as "therefore, the train will be on time". Therefore, as long as she believes D more than she believes the complaints of a random stranger on the train (i.e. argument B), she will believe A.

The above example indicates how considering arguments, and beliefs in them, can be a useful part of sense-making and decision-making in monological argumentation. How we model the influence of positive and negative relations is an important part of this. Furthermore, we can see that there is context-sensitivity, in that how we interpret arguments (in particular how we decode enthymemes) and the relationships between them can affect this analysis. We can also see that it is reasonable for different agents having different views on how to decode a given enthymeme, different views on the influence of one argument on another, and different views on how to take multiple relationships into account.

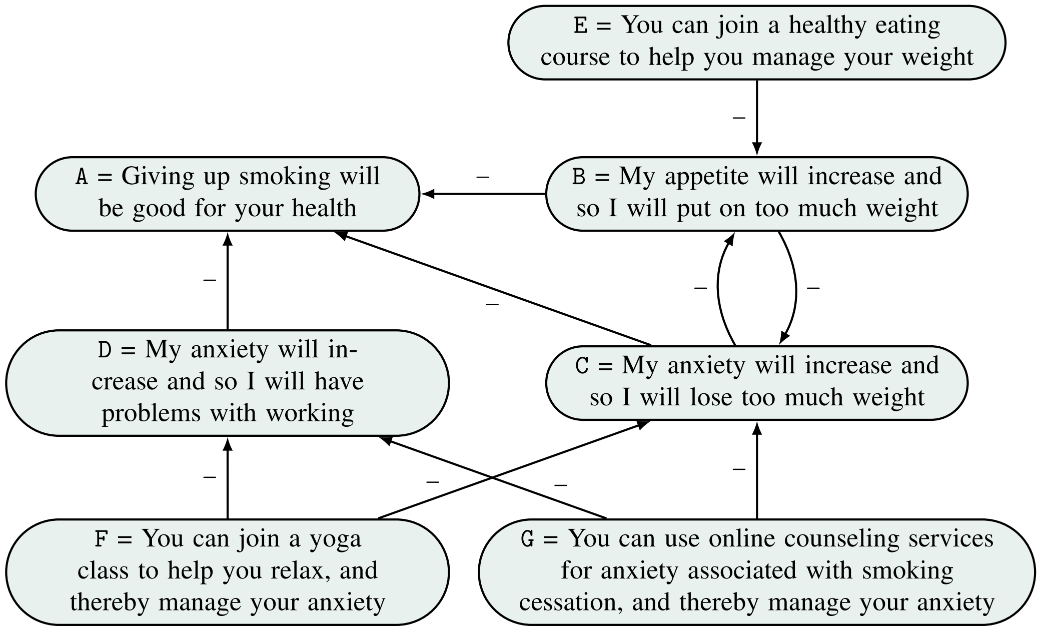


Figure 3: Example of argument graph for persuading someone to give up smoking. Edges labelled with − represent attack. The graph contains the arguments known (but not necessarily believed by) the artificial agent, and might not contain all arguments of Rachel, Robin or Morgan.

Example 2. Let us now assume that we have an artificial agent attempting to persuade the users Rachel, Robin and Morgan to stop smoking. The graph of the artificial agent is represented by Figure 3. The dialogue proceeds in turns and limits the ways the participants can respond. The artificial agent can state any of the arguments in the graph and the user is allowed to react in two ways. A user (be it Rachel, Robin or Morgan) can either select his/her counterargument from the list presented by the agent, or state how much (s)he agrees or disagrees with an argument presented by the agent. The user can end the dialogue at any time, the agent ends once there are no arguments to put forward or the user agreed to the desired arguments. After the dialogue is finished by any party, the participant is asked whether he or she agrees or disagrees with argument A. If the participant agrees, the dialogue is marked as successful.

Let us start with Rachel. The agent presents her with argument A in order to convince her to stop smoking and allows her to select from B, C and D as her potential arguments. Rachel selects B and D. In response to B, the agent puts forward E, and Rachel agrees. In response to D, the agent decides to first put forward F based on the experience with previous users. Unfortunately, Rachel strongly disagrees and ends the discussion. The dialogue is marked as unsuccessful. The agent was not aware that Rachel uses a wheelchair and that yoga classes did not suit her requirements, and the conversation ended before G could have been put forward.

Let us now consider Robin. The agent presents Robin with A and again allows B, C and D to be selected as counterarguments. Robin is afraid of any weight changes associated with smoking cessation and selects both B and C despite the fact that they are conflicting. Consequently, any counterarguments put forward by the artificial agent can be seen as at the same time indirectly conflicting with and promoting A. The agent puts forward E and F, to which Robin moderately agrees, and the dialogue ends successfully.

Finally, consider Morgan, who similarly to Rachel selects both B and D. However, in reality, Morgan is more afraid of weight gain than anxiety affecting his work, but wants to discuss both issues. The agent proposes solutions and Morgan moderately agrees with E, but somewhat disagrees with F. The agent decides to follow up with G, with which Morgan strongly disagrees. Nevertheless, the dialogue ends successfully due to the fact that Morgan's more pressing issue was addressed.

The above example indicates how beliefs in arguments and relations between them are important in dialogical argumentation. In particular, the same procedures applied to two agents expressing similar concerns can lead to different results based on the beliefs they have in arguments and their private knowledge. An agent not aware of another agent's arguments or beliefs can put forward unacceptable arguments and fail to persuade a given party to do or not to do something. One also has to be ready to put forward arguments that, possibly due to certain behaviours of the other party that can be deemed not rational, might work against the agent's goal.

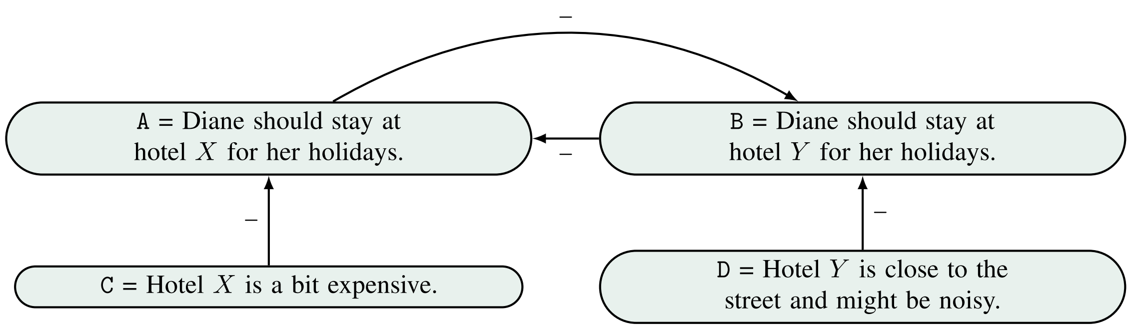


Figure 4: An argument graph representing Diane's hotel dilemma. Edges labelled with – represent attack.

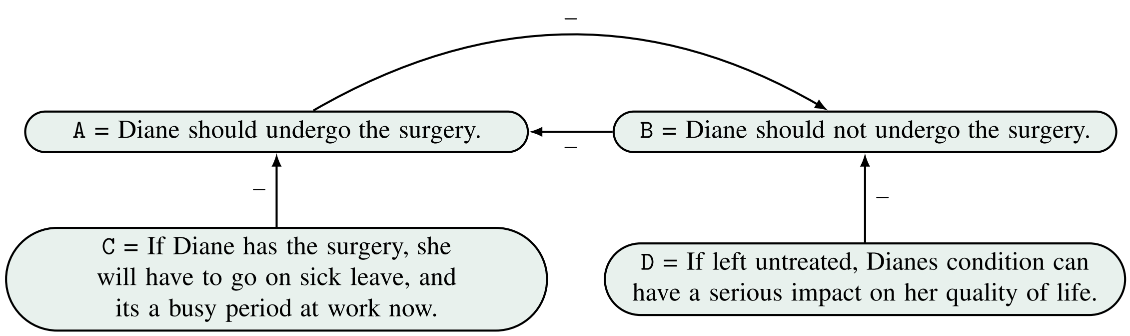


Figure 5: An argument graph representing Diane's surgery dilemma. Edges labelled with – represent attack.

Example 3. Let us consider two structurally identical argument graphs in Figures 4 and 5 representing two issues Diane had solved in the past, one concerning the choice of a hotel and one concerning undergoing the surgery. Let us first consider the hotel. Classical semantics would mark C and D as accepted arguments and reject both A and B. Even though Diane does believe C and D to a similar degree, the noise factor is not as important to her as is the price. Hence, she accepts argument B and rejects A.

Let us now consider the surgery graph in Figure 5. Again, both C and D are believed by Diane. However, in the end it is her health that is more important and D is a more compelling counterargument than C. Hence, she accepts argument A and rejects B.

This last example shows the importance of modelling context–sensitivity. In particular, equally believed arguments might not affect the arguments they are attacking to the same extent, and argumentation problems possessing structurally similar argument graphs may be evaluated differently by the same agent depending on his or her personal knowledge and preferences.

State of the Art

There are various approaches that attempt to tackle some of the above requirements. We can find a number of proposals in computational models of argument such as the postulates for argument weights, strengths or beliefs [23, 24, 55, 1, 5, 3, 2, 76, 17, 4, 6, 83, 46, 52], which offer a more fine–grained alternative for Dung's approach. Some of these works also permit certain forms of support or positive influences on arguments [23, 4, 3, 76, 55]. However, all of these approaches tend to introduce ways of aggregating influences that are general patterns applied to all nodes. For instance, for a graph where `A` is attacked by `B` and `C`, the way `B` and `C` influence `A` is based solely on this information about the structure of the graph, and takes no account of what arguments actually instantiate `A`, `B` and `C`. Hence, it is difficult for these proposals to meet the requirement for modelling context sensitivity, different perspectives or incomplete graphs. Certain flexibility is perhaps possible only with approaches that work with initial scoring assignment such as [76, 55, 4, 6]. Nevertheless, due to the way the influence aggregation methods are defined, dealing with imperfect agents still poses difficulties.

With the exception of the epistemic approach to probabilistic argumentation [83, 46, 52], the aforementioned proposals tend to consider unattacked arguments as the most desirable, thus creating issues in dealing with agents that we do not have a complete knowledge of. They also tend to aggregate the values assigned to the parents of a given argument into a single value for this argument. Thus, no margin of error or vagueness is possible, which adds to the difficulty of dealing with incomplete situations or imperfect agents. In these works, the values or weights assigned to arguments are either seen as a certain measure of strength or have no intrinsic meaning of their own. As a result, we can say that one argument is, for example, more acceptable or stronger than another based on their assigned weights, but we cannot say if they are acceptable at all in the first place unless an additional ad hoc function mapping scores to acceptance states is created for this purpose. Unfortunately, the epistemic approach is analyzed only in the context of attack relations.

So the above works allow fine–grained modelling and, apart from epistemic approach, also permit both supports and attacks in the framework. However, dealing with the context–sensitivity, different perspectives and agent imperfections pose difficulties. In contrast to them, there are frameworks that do allow us to specify the way one argument affects another locally, however, they are typically only equipped with standard two or three–valued semantics. In particular, abstract dialectical frameworks (ADFs) [20, 18, 67] allow us to specify various ways the incoming support and attack can affect a given argument. Then through the use of acceptance conditions, we can determine whether an argument is accepted or not based on the acceptance of its parents. The flexibility in specifying these conditions permits us certain context–sensitivity and modelling of imperfect agents in the system. Furthermore, the fact that the conditions also encode the nature of the relations present in the framework, it is possible for the users to express their different perspectives on the graph structure through different conditions. By creating an appropriate condition, we can also completely reject an argument despite not being aware of the reasons behind it, thus accommodating some form of incompleteness in our knowledge of the user. Unfortunately, ADFs need to be fully specified in order to be used, and conditions can be imposed only on arguments directly related to each other. Their semantics also adhere to the two or three–valued perspective and the condition associated with a given argument gives only a yes or no answer. Consequently, the degree to which a given argument is accepted or rejected is not considered. A recent proposal concerning weighted ADFs [41] allows values to be assigned to arguments and therefore offers a solution to the issue of fine–grained acceptability. Nevertheless, the conditions still need to be fully defined and can only refer to parents of a given argument. Furthermore, a combination of parent values in weighted ADFs is associated with one particular value of the target. This high degree of specificity can pose problems in modelling imperfect agents or incomplete graphs, as obtaining acceptance conditions describing such agent with this type of accuracy might not always be possible in a reasonable amount of time.

Finally, we also have the constraint–based argumentation framework [31], which permits external requirements among unrelated arguments to be imposed in the framework. Although it has only been analyzed in the context of attack–based graphs, certain positive relationships between arguments could potentially be simulated through the use of propositional formulae representing the external requirements. Nevertheless, this modelling is targeted mainly at two–valued semantics, and thus the framework does not deal with fine–grained acceptability.

Our Proposal

From our review of the state of the art, we believe that none of the currently available formalisms meets all of our requirements. Hence, there is a need to investigate argumentative approaches that would handle both attack and support relations, allow for fine–grained argument acceptability, and permit context-sensitivity, different perspectives, agents' imperfections and incomplete knowledge about agents' graphs.

To address these requirements, we introduce epistemic graphs as a generalization of the epistemic approach to probabilistic argumentation, which has already shown to be potentially valuable in modelling agents in persuasion dialogues [47, 49, 50, 42, 43]. In these graphs, an argument can be believed or disbelieved to a given degree, and an edge can represent negative as well as positive or mixed relations between arguments. The way other arguments influence a given argument is expressed by the epistemic constraints that can restrict the belief we have in an argument with a varying degree of specificity. An example of how constraints can represent agents' requirements can be seen below.

Example 4. Let us consider Example 1 once more and see how it could be formalized using the epistemic graphs. The original graphs are amended by the addition of special constraints that describe how beliefs in arguments influence each other. Let us assume that the belief is on a $[0, 1]$ scale, where 0 represents complete disbelief, 1 complete belief and 0.5 is seen as neither believing nor disbelieving something. Thus, a value greater than 0.5 is seen as belief and less than 0.5 as disbelief. For now, let $p(\mathtt{A})$ denote the belief we have in $\mathtt{A}$. For example, $p(\mathtt{A}) > 0.5$ means that "belief in $\mathtt{A}$ is greater than 0.5". From these elements, we can build formulae using the standard $\wedge$, $\vee$, $\neg$, $\rightarrow$ connectives.

Let us focus on Jack first. We can describe the way the belief in $\mathtt{A}$ is influenced by other arguments in the following way:

- if $\mathtt{B}$ and $\mathtt{C}$ are believed, then $\mathtt{A}$ is disbelieved
- if $\mathtt{B}$ is believed and $\mathtt{C}$ is strongly believed, then $\mathtt{A}$ is strongly disbelieved
- if $\mathtt{B}$ is not disbelieved and $\mathtt{C}$ is neither believed nor disbelieved, then $\mathtt{A}$ is somewhat believed
- if $\mathtt{B}$ and $\mathtt{C}$ are neither believed nor disbelieved, then $\mathtt{A}$ is neither believed nor disbelieved
- if $\mathtt{B}$ or $\mathtt{C}$ is disbelieved, then $\mathtt{A}$ is believed

Let us assume that strong belief is the one greater than 0.85, moderate belief is between 0.65 and 0.85, and that an argument is somewhat believed if its score is between 0.5 and 0.65 (we can create disbelief ranges in a similar fashion). Then, a possible way to write down Jack's constraints is as follows:

- $(p(\mathtt{B}) > 0.5 \wedge p(\mathtt{C}) > 0.5) \rightarrow p(\mathtt{A}) < 0.5$
- $(p(\mathtt{B}) > 0.5 \wedge p(\mathtt{C}) \geq 0.85) \rightarrow p(\mathtt{A}) \leq 0.15$
- $(\neg p(\mathtt{B}) < 0.5 \wedge p(\mathtt{C}) = 0.5) \rightarrow (p(\mathtt{A}) > 0.5 \wedge p(\mathtt{A}) < 0.65)$
- $(p(\mathtt{B}) = 0.5 \wedge p(\mathtt{C}) = 0.5) \rightarrow p(\mathtt{A}) = 0.5$
- $(p(\mathtt{B}) < 0.5 \vee p(\mathtt{C}) < 0.5) \rightarrow p(\mathtt{A}) > 0.5$

Let us now focus on Jill. A possible way to describe her views on how $\mathtt{A}$ is influenced by other arguments is as follows:

- if $\mathtt{D}$ is believed and is believed more than $\mathtt{B}$, then $\mathtt{A}$ is believed
- if $\mathtt{D}$ is believed but believed no more than $\mathtt{B}$, then $\mathtt{A}$ is neither believed nor disbelieved
- if $\mathtt{D}$ is disbelieved, then $\mathtt{A}$ is disbelieved
- if $\mathtt{D}$ is strongly disbelieved then $\mathtt{A}$ is at best moderately disbelieved
- if $\mathtt{D}$ is strongly disbelieved and $\mathtt{B}$ is believed then $\mathtt{A}$ is strongly disbelieved

A possible way to formalize this is as follows:

- $(p(\mathtt{D}) > 0.5 \wedge p(\mathtt{D}) - p(\mathtt{B}) > 0) \rightarrow p(\mathtt{A}) > 0.5$
- $(p(\mathtt{D}) > 0.5 \wedge p(\mathtt{D}) - p(\mathtt{B}) \leq 0) \rightarrow p(\mathtt{A}) = 0.5$
- $p(\mathtt{D}) < 0.5 \rightarrow p(\mathtt{A}) < 0.5$
- $p(\mathtt{D}) \leq 0.15 \rightarrow p(\mathtt{A}) \leq 0.35$
- $(p(\mathtt{D}) \leq 0.15 \vee p(\mathtt{B}) > 0.5) \rightarrow p(\mathtt{A}) \leq 0.15$

Epistemic graphs therefore provide a more fine–grained alternative to classical Dung's approaches when it comes to determining the status of a given argument. The flexibility of the epistemic approach allows us to both model the rationale behind the existing semantics as well as completely deviate from them, thus giving us a more appropriate formalism for practical situations including the modelling of imperfect agents. Epistemic graphs can model both attack and support as well as relations that are neither support nor attack, so far analyzed primarily in the context of abstract dialectical frameworks [18]. The freedom in defining the constraints allows us to easily express various interpretations of support at the same time and without the need to transform them, which is usually necessary in other types of argumentation frameworks [25, 73, 67]. The fact that we can specify the rules under which arguments should be evaluated and that we can include constraints between unrelated arguments allows the framework to be more context–sensitive and more accommodating when it comes to dealing with imperfect agents. Additionally, the ability to leave certain relations unspecified lets us deal with cases when the system has insufficient knowledge about the situation.

Outline of the Paper

We start with a brief background on argumentation in Section 2. In Section 3 we introduce two formats of epistemic constraints and graphs, present their semantics and show how they can be used to model a real–life scenario. Then, in Section 4, we provide a sound and complete proof theory for reasoning with constraints as well as various normal forms for them. Section 5 is devoted to the analysis of epistemic graphs. We present a number of possible types of graphs, analyze the ways arguments and relations are covered by the constraints and how the nature of a given relation can be inferred from the constraints. In Section 6 we provide a study on computational issues related to epistemic graphs, while in Section 7 we compare our work to related state-of-the-art formalisms. We close the paper with a discussion of our contributions and future work.

2 Preliminaries

In its simplest form, an argument graph is a directed graph in which nodes represent arguments and arcs represent relations. In conflict–based graphs, such as the ones created by Dung [32], arcs stand for attacks. In graphs such as those in [3], arcs are supports, while in bipolar graphs they can be either supports or attacks [25, 16, 62, 60, 73]. In some frameworks, such as abstract dialectical frameworks, an arc may also represent a dependence relation in case it cannot be strictly classified as neither supporting nor attacking [20, 18, 67]. Argument graphs can be extended in various ways in order to account for additional preferences, recursive relations [1], group relations [2] and more. For an overview, we refer the reader to [19]. We will also discuss some of these structures more in Section 7. For now, we will focus on recalling some basic notions and introducing the notation we will use throughout the text.

By an argument graph we will understand a directed graph and we will use a labelling function that assigns to every arc a label representing its nature – supporting, attacking, or dependent, where dependency is understood as a relation that is neither positive nor negative. Hence, unless stated otherwise, we will

[1] Frameworks with recursive relations are represented as generalizations of directed graphs where edges point at other edges.

[2] Frameworks with group relations are often represented by B–graphs, i.e. directed hypergraphs where the head of the edge is a single node.

assume we are working with a label set $\Omega = \{+, *, -\}$, which can be adjusted if needed. Given that in many graphs allowing more than a single relation to be represented it can happen that two arguments are connected in more way than one, we allow a single arc to possess more than just one label:

Definition 2.1. Let $\mathcal{G} = (V, A)$, where $A \subseteq V \times V$, be a directed graph. A **labelled graph** is a tuple $X = (\mathcal{G}, \mathcal{L})$ where $\mathcal{L} : \mathsf{Arcs}(\mathcal{G}) \to 2^{\Omega}$ is labelling function and Ω is a set of possible labels. X is **fully labelled** iff for every $\alpha \in \mathsf{Arcs}(\mathcal{G})$, $\mathcal{L}(\alpha) \neq \emptyset$. X is **uni-labelled** iff for every $\alpha \in \mathsf{Arcs}(\mathcal{G})$, $|\mathcal{L}(\alpha)| = 1$.

Unless stated otherwise, from now on we assume that we are working with fully labelled graphs. With $\mathsf{Nodes}(\mathcal{G})$ we denote the set of nodes V in the graph $\mathcal{G}$ and with $\mathsf{Arcs}(\mathcal{G})$ we denote the set of arcs A in $\mathcal{G}$. For a graph $\mathcal{G}$ and a node $\mathtt{B} \in \mathsf{Nodes}(\mathcal{G})$, the parents of $\mathtt{B}$ are $\mathsf{Parent}(\mathtt{B}) = \{\mathtt{A} \mid (\mathtt{A}, \mathtt{B}) \in \mathsf{Arcs}(\mathcal{G})\}$. With $\mathcal{L}^x(\mathcal{G}) = \{\alpha \in \mathsf{Arcs}(\mathcal{G}) \mid x \in \mathcal{L}(\alpha)\}$ we denote the set of relations labelled with x by $\mathcal{L}$, where $x \in \{+, *, -\}$. In a similar fashion, by $\mathsf{Parent}^x(\mathtt{B}) = \{\mathtt{A} \mid (\mathtt{A}, \mathtt{B}) \in \mathsf{Arcs}(\mathcal{G}) \wedge x \in \mathcal{L}((\mathtt{A}, \mathtt{B}))\}$ we will denote the set of parents of an argument B s.t. the relation between the two is labelled with x by $\mathcal{L}$.

On an arc from a parent to the target, a positive label denotes a positive influence, a negative label denotes a negative influence, and a star label denotes an influence that is neither strictly positive nor negative. If $\mathcal{L}$ is assigns only the $-$ label to every arc in a graph, then the graph is a conflict–based argument graph, and if $\mathcal{L}$ is assigns $+$ or $-$ (or both) to every arc in a graph, then the graph is a bipolar argument graph [60, 25, 73]. Following the analysis in [71], a graph making use of all three labels will be referred to as tripolar. In Figure 3 we can see an example of a conflict–based argument graph, Figure 1 shows an example of a bipolar argument graph and Figure 6 of a tripolar one. In the last case, we can observe that while E and F are necessary for A, only one of them can be accepted at a time in order for A to be accepted, as having both of them would lead to rejecting the argument. This mutually exclusive requirement for A is neither an attacking nor a supporting relation, and thus it is classified as a dependency.

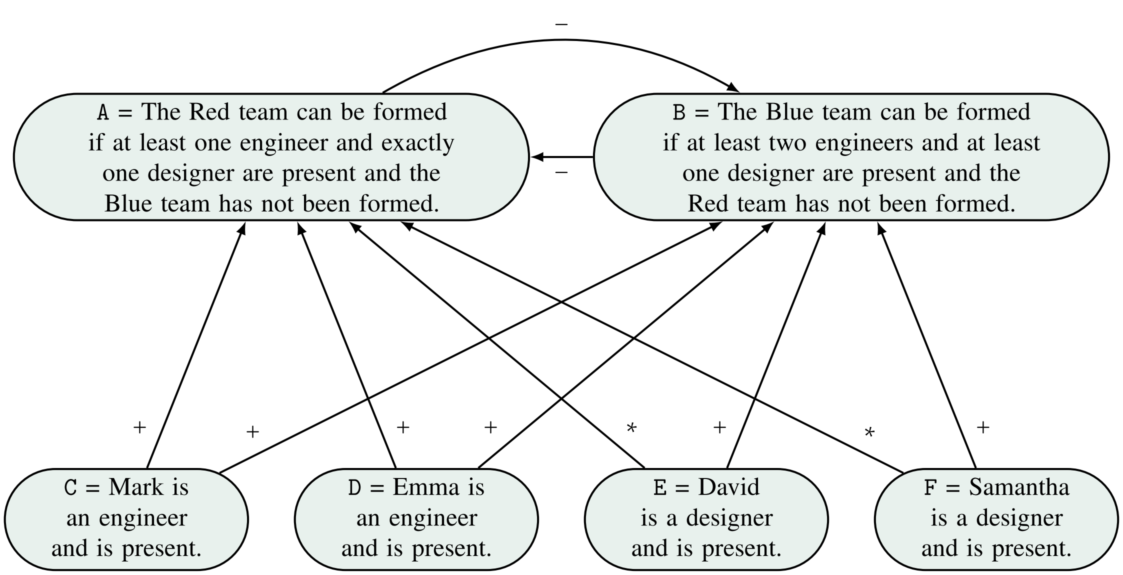


Figure 6: A tripolar graph example. Edges labelled with $-$, $+$ and $*$ represent attack, support and dependency respectively. Forming of the Red and Blue teams requires particular specialists. Arguments C, D, E and F support the creation of the Blue team. However, only C and D strictly support the creation of the Red team. If we accept E, then acceptance of F leads to the rejection of the Red team, and if we accept F, then the acceptance of E leads to the rejection of the Red team. At the same time, one of E and F has to be accepted. Thus, the relations from E and F are in some cases attacking, in some supporting, and hence they can only be classified as dependent.

A given argument graph is evaluated with the use of semantics, which are meant to represent what can be considered "reasonable". The most basic type of semantics – the extension–based ones – associates a given graph with sets of arguments, called extensions, formed from acceptable arguments. A more refined version, the labeling–based semantics, tell us whether an argument is accepted, rejected, or neither [22, 9]. However, when it comes to user modelling, these two and three–valued perspectives turned out to be insufficient to express the extent to which the user agrees or disagrees with a given argument [77, 71]. Consequently, a variety of weighted, ranking–based and probabilistic approaches were proposed [1, 2, 3, 5, 76, 17, 4, 6, 46, 47, 49, 50, 42, 43, 75, 13, 23, 24, 55]. To put it simply, a semantics is a function that associates an argument graph with "answers", which can follow various formats:

Definition 2.2. Let $X = (\mathcal{G}, \mathcal{L})$ be a labelled argument graph.

- an **extension** is a set of arguments $Ext \subseteq \mathsf{Nodes}(\mathcal{G})$
- an **argument labeling** is a total function $Lab : \mathsf{Nodes}(\mathcal{G}) \to \{\mathsf{in}, \mathsf{out}, \mathsf{und}\}$
- a **ranking** on arguments is a binary relation $\preceq$ on $\mathsf{Nodes}(\mathcal{G})$ that is total and transitive
- a **weighting** on arguments is a function $w : \mathsf{Nodes}(\mathcal{G}) \to [0, 1]$
- a **belief distribution** on arguments is a function $P : 2^{\mathsf{Nodes}(\mathcal{G})} \to [0, 1]$ s.t. $\sum_{\Gamma \subseteq \mathsf{Nodes}(\mathcal{G})} P(\Gamma) = 1$.

Definition 2.3. Let $X = (\mathcal{G}, \mathcal{L})$ be a labelled argument graph.

- an **extension–based semantics** associates X with a set $Y \subseteq 2^{\mathsf{Nodes}(\mathcal{G})}$
- a **labeling–based semantics** associates X with a set $Y \subseteq \mathcal{U}_X^{Lab}$, where $\mathcal{U}_X^{Lab}$ is the set of all argument labelings on X
- a **ranking–based semantics** associates X with a ranking on $\mathsf{Nodes}(\mathcal{G})$
- a **weight–based semantics** associates X with a weighting on $\mathsf{Nodes}(\mathcal{G})$
- an **epistemic semantics** associates X with a set $\mathcal{R} \subseteq \mathsf{Dist}(\mathcal{G})$, where $\mathsf{Dist}(\mathcal{G})$ is the set of all belief distributions over $\mathcal{G}$

We say that argument $\mathtt{A}$ is accepted in an extension if it is contained in it; otherwise, it is rejected. In the case of a labeling, the argument is in (i.e. accepted), out (i.e. rejected), or und (i.e. undecided). By $\mathsf{in}(Lab), \mathsf{out}(Lab)$ and $\mathsf{und}(Lab)$ we denote the arguments mapped respectively to $\mathsf{in}, \mathsf{out}$ and und by a labeling Lab. We will often write a labeling as a triple (I, O, U), where I, O and U are the sets of arguments mapped to in, out and und. In the case of a ranking, $\mathtt{A} \preceq \mathtt{B}$ means that $\mathtt{A}$ is at least as acceptable as $\mathtt{B}$. In the case of the epistemic semantics, we say that an argument $\mathtt{A}$ has a given probability, which is seen as the sum of the probabilities (beliefs) associated with sets of arguments containing it:

$$P(\mathtt{A}) = \sum_{\Gamma \subseteq \mathsf{Nodes}(\mathcal{G}) \text{ s.t. } \mathtt{A} \in \Gamma} P(\Gamma)$$

The probability of a single argument is understood as the belief that an agent has in it, i.e. we say that an agent believes an argument $\mathtt{A}$ to some degree when $P(\mathtt{A}) > 0.5$, disbelieves an argument to some degree when $P(\mathtt{A}) < 0.5$, and neither believes nor disbelieves an argument when $P(\mathtt{A}) = 0.5$. It is possible to relate belief distributions to extensions and labelings by taking as accepted arguments those that are believed, as rejected those that are disbelieved, and as undecided those that are neither believed nor disbelieved.

Given that a semantics is supposed to reflect what is considered rational, arbitrary assignments are not desirable and we usually expect the semantics to meet certain properties. These properties depend on the output format of the semantics and on the types of relations we have in the graph. Although in some cases it is possible to draw connections between some of these semantics (for example, between extension and labeling–based ones for Dung's graph, and between the ranking and weighting–based semantics), they tend to follow different design choices and intuitions and can lead to different results despite potential structural similarities. We exemplify some of the approaches below and refer the readers to previously cited works and Section 7 for further properties and analysis. In the following, we are working with a labelled graph $X = (\mathcal{G}, \mathcal{L})$ s.t. for every $\alpha \in \mathsf{Arcs}(\mathcal{G})$, $\mathcal{L}(\alpha) = \{-\}$.

Definition 2.4. A set $Ext \subseteq \mathsf{Nodes}(\mathcal{G})$ is:

- **conflict–free** iff for no $\mathtt{A}, \mathtt{B} \in Ext$, $\mathtt{A}$ is an attacker of $\mathtt{B}$.
- **admissible** iff it is conflict–free and defends all of its members.
- **complete** iff it is admissible and all arguments defended by Ext are contained in Ext.

Example 5. Consider the framework $(\mathcal{G}, \mathcal{L})$ where $\mathcal{G} = (\{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}, \mathtt{E}\}, \{(\mathtt{A},\mathtt{B}), (\mathtt{C},\mathtt{B}), (\mathtt{C},\mathtt{D}), (\mathtt{D},\mathtt{C}), (\mathtt{D},\mathtt{E}), (\mathtt{E},\mathtt{E})\})$ and $\mathcal{L}$ assigns only the $-$ label to every edge, as depicted in Figure 7. This framework has eight conflict–free extensions in total: $\{\mathtt{A},\mathtt{C}\}$,$\{\mathtt{A},\mathtt{D}\}$, $\{\mathtt{B},\mathtt{D}\}$, $\{\mathtt{A}\}$, $\{\mathtt{B}\}$, $\{\mathtt{C}\}$, $\{\mathtt{D}\}$ and $\varnothing$. As $\mathtt{B}$ is attacked by an unattacked argument, it cannot be defended against it and will not be in any admissible extension. From this $\{\mathtt{A},\mathtt{C}\}$, $\{\mathtt{A},\mathtt{D}\}$ and $\{\mathtt{A}\}$ are complete.

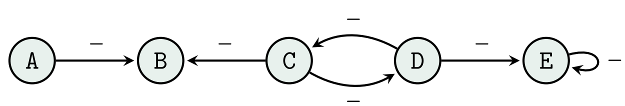


Figure 7: A conflict–based argument graph

Definition 2.5 ([9])**.** Let $Lab : \mathsf{Nodes}(\mathcal{G}) \to \{\mathsf{in}, \mathsf{out}, \mathsf{und}\}$ be a labeling. Then Lab is:

- **conflict-free** iff every $\mathtt{A} \in \mathsf{out}(Lab)$ has an attacker in $\mathsf{in}(Lab)$ and there is no $\mathtt{B}, \mathtt{C} \in \mathsf{in}(Lab)$ s.t. $(\mathtt{B}, \mathtt{C}) \in \mathsf{Arcs}(\mathcal{G})$.
- **admissible** iff it is conflict–free and every $\mathtt{A} \in \mathsf{in}(Lab)$ has all of its attackers in $\mathsf{out}(Lab)$.
- **complete** if it is admissible and for every $\mathtt{A} \in \mathsf{und}(Lab)$, not all of its attackers are in $\mathsf{out}(Lab)$ and it does not have an attacker in $\mathsf{in}(Lab)$.

Example 6. Let us come back to the framework from Example 5. The admissible labelings of our framework are visible in Table 1. We can observe that one admissible extension can be associated with more than just a single labeling. However, out of the possible interpretations, only Lab_3, Lab_9 and Lab_{13} are complete. They are now also in one–to–one relation with the complete extensions.

Extension		$\varnothing$	$\{\mathtt{A}\}$	$\{\mathtt{A}\}$	$\{\mathtt{C}\}$	$\{\mathtt{C}\}$	$\{\mathtt{D}\}$	$\{\mathtt{D}\}$	$\{\mathtt{A},\mathtt{C}\}$	$\{\mathtt{A},\mathtt{C}\}$	$\{\mathtt{A},\mathtt{D}\}$	$\{\mathtt{A},\mathtt{D}\}$	$\{\mathtt{A},\mathtt{D}\}$	$\{\mathtt{A},\mathtt{D}\}$
	#	Lab_1	Lab_2	Lab_3	Lab_4	Lab_5	Lab_6	Lab_7	Lab_8	Lab_9	Lab_{10}	Lab_{11}	Lab_{12}	Lab_{13}
Labeling	A	und	in	in	und	und	und	und	in	in	in	in	in	in
	B	und	und	out	out	und	und	und	und	out	und	und	out	out
	C	und	und	und	in	in	out	out	in	in	out	out	out	out
	D	und	und	und	out	out	in	in	out	out	in	in	in	in
	E	und	und	und	und	und	und	out	und	und	und	out	und	out

Table 1: Admissible labelings of the framework from Example 5.

Definition 2.6 ([1])**.** A ranking $\preceq$ is **burden–based** iff for every $\mathtt{A}, \mathtt{B} \in \mathsf{Nodes}(\mathcal{G})$, $\mathtt{A} \preceq \mathtt{B}$ iff one of the two following cases holds:

- $\forall i \in \{0, 1, \ldots\}, Bur_i(\mathtt{A}) = Bur_i(\mathtt{B})$
- $\exists i \in \{0, 1, \ldots\}, Bur_i(\mathtt{A}) < Bur_i(\mathtt{B})$ and $\forall j \in \{0, 1, \ldots, i-1\}, Bur_j(\mathtt{A}) = Bur_j(\mathtt{B})$

where the burden number of an argument $\mathtt{C}$ on the i-th step $Bur_i(\mathtt{C})$is defined as

$$Bur_i(\mathtt{C}) = \begin{cases} 1 & \text{if } i = 0 \\ 1 + \sum_{\mathtt{D} \in \mathsf{Parent}^-(\mathtt{C})} 1/Bur_{i-1}(\mathtt{D}) & \text{otherwise} \end{cases}$$

Example 7. An example of how the burden numbers for the framework from Example 5 can be computed is visible in Table 2. Although we can continue analysing further steps, it becomes clear that A will be ranked better than any other argument, C and D will be ranked equally and better than B and E, and B will be ranked the least. We therefore obtain the ranking $\mathtt{A} \preceq \mathtt{C}, \mathtt{D} \preceq \mathtt{E} \preceq \mathtt{B}$.

#	A	B	C	D	E
$i = 0$	1	1	1	1	1
$i = 1$	1	3	2	2	3
$i = 2$	1	2.5	1.5	1.5	$\frac{11}{6} \approx 1.83$
$i = 3$	1	$\frac{8}{3} \approx 2.67$	$\frac{5}{3} \approx 1.67$	$\frac{5}{3} \approx 1.67$	$\frac{73}{33} \approx 2.12$
$i = 4$	1	2.6	1.6	1.6	$\frac{749}{365} \approx 2.05$
			⋮		

Table 2: Example of computation of burden numbers on the argument graph from Example 5.

Definition 2.7 ([13, 75])**.** A weighting w is an **h–categorizer** iff it satisfies the following:

$$w(\mathtt{A}) = \begin{cases} 1 & \mathsf{Parent}^-(\mathtt{A}) = \varnothing \\ \frac{1}{1+\sum_{\mathtt{B}\in\mathsf{Parent}^-(\mathtt{A})} w(\mathtt{B})} & \mathsf{Parent}^-(\mathtt{A}) \neq \varnothing \end{cases}$$

Example 8. The h–categorizer weighting for the framework from Example 5 produces the following weights: $w(\mathtt{A}) = 1$, $w(\mathtt{B}) \approx 0.38$, $w(\mathtt{C}) = w(\mathtt{D}) \approx 0.62$ and $w(\mathtt{E}) \approx 0.48$.

Finally, we recall some examples of epistemic postulates, which can be used to refine the belief distributions:

Definition 2.8 ([83, 71])**.** A belief distribution P is:

- **rational** if for every $\mathtt{A}, \mathtt{B} \in \mathsf{Nodes}(\mathcal{G})$ s.t. $(\mathtt{A}, \mathtt{B}) \in \mathcal{L}^-(\mathcal{G})$, $P(\mathtt{A}) > 0.5$ implies $P(\mathtt{B}) \leq 0.5$
- **trusting** if for every $\mathtt{B} \in \mathsf{Nodes}(\mathcal{G})$, if $P(\mathtt{A}) < 0.5$ for all $\mathtt{A} \in \mathsf{Parent}^-(\mathtt{B})$, then $P(\mathtt{B}) > 0.5$
- **optimistic** if $P(\mathtt{A}) \geq 1 - \sum_{\mathtt{B}\in\mathsf{Parent}^-(\mathtt{A})} P(\mathtt{B})$ for every $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$
- **ternary** if $P(\mathtt{A}) \in \{0, 0.5, 1\}$ for every $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$

Example 9. Technically speaking, we can create infinitely many belief distributions on the graph from Example 5 that satisfy the properties we have recalled. We will therefore provide only a few examples, visible in Table 3. We can observe that the sets of arguments believed in the rational distributions are in fact conflict–free in our graph. For example, P_2 and P_6 produce the set $\{\mathtt{C}\}$, while P_4 and P_5 produce the extension $\{\mathtt{A}, \mathtt{C}\}$. Further postulates could be assumed in order to retrieve various extensions or labelings of our graph [71, 72].

3 Epistemic Graphs

In the introduction, we have discussed the value of being able to model beliefs in arguments, various types of relations between arguments, context–sensitivity, and more. Our proposal, capable of meeting the postulated requirements, comes in the form of epistemic graphs, which can be equipped with particular formulae specifying the beliefs in arguments and the interplay between them. In this section we formalize the idea of epistemic graphs and constraints, previously briefly described in Example 4. We explain how constraints can be specified and interpreted, define epistemic graphs and their semantics, and provide an example of how our proposal can be used in practical applications.

#	A	B	C	D	E	Rational	Trusting	Optimistic	Ternary
P_1	0.6	0.5	0.2	0.4	0.8	×	×	×	×
P_2	0.3	0.5	0.6	0.2	0.5	✓	×	×	×
P_3	0.6	1	0.4	0.7	0.5	×	✓	×	×
P_4	0.7	0.4	0.6	0.4	0.5	✓	✓	×	×
P_5	1	0	1	0	0.5	✓	✓	✓	✓
P_6	0.5	0	1	0	0.5	✓	×	×	✓
P_7	1	1	1	1	1	×	✓	✓	✓
P_8	1	0.7	0.9	0.3	0.7	×	✓	✓	×

Table 3: Satisfaction of epistemic postulates on the argument graph from Example 5.

3.1 Epistemic Constraints

In order to show how constraints can be harnessed in epistemic graphs, we first need to specify their representation. A **term** is a Boolean combination of arguments. If $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, then $\mathtt{A}$ is a positive literal, and $\neg\mathtt{A}$ is a negative literal. Let $\mathsf{Terms}(\mathcal{G})$ denote all the terms that can be formed from the arguments in $\mathcal{G}$ using $\vee$, $\wedge$ and $\neg$ as connectives in the usual way (we can derive secondary connectives, such as implication $\rightarrow$, in the usual manner). In order to formalize the satisfaction of a term, we treat each subset of $\mathsf{Nodes}(\mathcal{G})$ as a model (i.e. a possible world).

Definition 3.1. The **term satisfaction relation**, denoted $\models$, is defined as follows where $X \subseteq \mathsf{Nodes}(\mathcal{G})$, $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, and $\alpha, \beta \in \mathsf{Terms}(\mathcal{G})$:

- $X \models \mathtt{A}$ when $\mathtt{A} \in X$;
- $X \models \alpha \wedge \beta$ iff $X \models \alpha$ and $X \models \beta$;
- $X \models \alpha \vee \beta$ iff $X \models \alpha$ or $X \models \beta$; and
- $X \models \neg\alpha$ iff $X \not\models \alpha$.

Essentially $\models$ is a classical satisfaction relation. So if α is a classical tautology (or $\top$), then $X \models \alpha$ for all $X \subseteq \mathsf{Nodes}(\mathcal{G})$, and if α is a classical contradiction (or $\bot$), then $X \not\models \alpha$ for all $X \subseteq \mathsf{Nodes}(\mathcal{G})$. For $\alpha \in \mathsf{Terms}(\mathcal{G})$, let $\mathsf{Models}(\alpha) = \{X \subseteq \mathsf{Nodes}(\mathcal{G}) \mid X \models \alpha\}$. For each graph $\mathcal{G}$, we assume an ordering over the arguments $\langle \mathtt{A}_1, \dots, \mathtt{A}_n \rangle$ so that we can encode each model by a binary number: for a model X, if the ith argument is in X, then the ith digit is 1, otherwise it is 0. For example, for $\langle \mathtt{A}, \mathtt{B}, \mathtt{C} \rangle$, the model $\{\mathtt{A}, \mathtt{C}\}$ is represented by 101.

Since in our constraints, we want to express how the degree of belief in one argument affects another, we need to define what is the probability of a term. In our approach, the probability of a term α is the sum of the probability of each model satisfying the term:

Definition 3.2. For $P \in \mathsf{Dist}(\mathcal{G})$, the **probability of term** $\alpha \in \mathsf{Terms}(\mathcal{G})$ is $P(\alpha) = \sum_{X \in \mathsf{Models}(\alpha)} P(X)$.

Suppose $\alpha \in \mathsf{Terms}(\mathcal{G})$ and P is a belief distribution. If α is a contradiction of classical logic, then $P(\alpha) = 0$, and if α is a tautology of classical logic, then $P(\alpha) = 1$. Also, if $\{\alpha\} \vdash \beta$, then $P(\alpha) \leq P(\beta)$, and if $\neg(\alpha \wedge \beta)$ is a classical tautology, then $P(\alpha \vee \beta) = P(\alpha) + P(\beta)$, where by $\vdash$ we understand the classical consequence relation.

With $\mathsf{Args}(\alpha)$, where $\alpha \in \mathsf{Terms}(\mathcal{G})$, we will denote the set of all arguments appearing in α. By the abuse of notation, with $\mathsf{Args}(\Gamma)$ we will denote the set of all arguments appearing in a set of terms $\Gamma \subseteq \mathsf{Terms}(\mathcal{G})$.

We can now formally define what we understand as an epistemic constraint and distinguish between the two formats, namely the basic and valued one. Epistemic semantics, briefly exemplified in Definition 2.8, comprise of a wide range of possible postulates [71]. Many of them follow the format of the rational and trusting semantics, i.e. the fact that the belief in one argument was greater, less or equal to a given numerical value implied that the belief in another argument had to be greater, less or equal to a given numerical value. Thus, the basic components of such postulates are of the form $P(\ldots)\#x$, where $x \in [0,1]$ and $\# \in \{=, \neq, \geq, \leq, >, <\}$, and by creating appropriate formulas using such components we can express a number of the existing properties. However, there are also semantics for which we need to create a formula that combines the probabilities of arguments using arithmetic operators. For example, one can consider the optimistic postulate, which demands that every argument satisfies the formula $P(\texttt{A}) + \sum_{\texttt{B} \in \mathsf{Parent}^-(\texttt{A})} P(\texttt{B}) \geq 1$. This calls for a more advanced constraint format to which we will refer as *valued*. In what follows, we therefore choose to investigate both of these forms.

3.1.1 Basic Constraints

Previously, we have noted that many epistemic postulates (see Definition 2.8 and [71]), are built of components of the form $P(\gamma)\#x$, where $x \in [0,1]$ and $\# \in \{=, \neq, \geq, \leq, >, <\}$. By allowing γ to represent any boolean formula on arguments, we obtain the basic epistemic atom, from which the basic formulae can be built:

Definition 3.3. A **basic epistemic atom** is of the form $p(\alpha)\#x$ where $\alpha \in \mathsf{Terms}(\mathcal{G})$ is a term, $\# \in \{=, \neq, \geq, \leq, >, <\}$, and $x \in [0,1]$. A **basic epistemic formula** is a Boolean combination of basic epistemic atoms (i.e. if ϕ is an atom, then it is a basic epistemic formula, and if ϕ and ψ are basic epistemic formulae, then each of $\phi \wedge \psi$, $\phi \vee \psi$ and $\neg\phi$ is a basic epistemic formula). Let $\mathsf{BFormulae}(\mathcal{G})$ be the set of basic epistemic formulae that can be formed from the arguments in $\mathcal{G}$.

Given a basic formula $\psi \in \mathsf{BFormulae}(\mathcal{G})$, with $\mathsf{FTerms}(\psi)$ we denote the set of terms appearing in ψ and with $\mathsf{FArgs}(\psi) = \mathsf{Args}(\mathsf{FTerms}(\psi))$ the set of arguments appearing in ψ. With $\mathsf{Num}(\psi)$ we denote the collection of all numerical values x appearing in ψ.

Example 10. For $\texttt{A}, \texttt{B} \in \mathsf{Nodes}(\mathcal{G})$, $\psi : (p(\texttt{A} \wedge \texttt{B}) > 0.9) \vee (p(\neg\texttt{A} \wedge \neg\texttt{B}) < 0.5)$ is an example of a basic epistemic formula. The terms of that formula are $\mathsf{FTerms}(\psi) = \{\texttt{A} \wedge \texttt{B}, \neg\texttt{A} \wedge \neg\texttt{B}\}$, the arguments appearing in them are $\mathsf{FArgs}(\psi) = \{\texttt{A}, \texttt{B}\}$, and the numerical values of the formula are $\mathsf{Num}(\psi) = \{0.9, 0.5\}$.

Definition 3.4. The **satisfying distributions** for an atom $p(\alpha)\#x$ is $\mathsf{Sat}(p(\alpha)\#x) = \{P' \in \mathsf{Dist}(\mathcal{G}) \mid P'(\alpha)\#x\}$, where $\# \in \{=, \neq, \geq, \leq, >, <\}$. The set of satisfying distributions for a basic epistemic formula is as follows where ϕ and ψ are basic epistemic formulae:

- $\mathsf{Sat}(\phi \wedge \psi) = \mathsf{Sat}(\phi) \cap \mathsf{Sat}(\psi)$;
- $\mathsf{Sat}(\phi \vee \psi) = \mathsf{Sat}(\phi) \cup \mathsf{Sat}(\psi)$; and
- $\mathsf{Sat}(\neg\phi) = \mathsf{Sat}(\top) \setminus \mathsf{Sat}(\phi)$.

Also, for a set of basic formulae $\Phi = \{\phi_1, \ldots, \phi_m\}$, the set of satisfying distributions is $\mathsf{Sat}(\Phi) = \mathsf{Sat}(\phi_1) \cap \ldots \cap \mathsf{Sat}(\phi_n)$.

Example 11. For a graph s.t. $\mathsf{Nodes}(\mathcal{G}) = \{\texttt{A}, \texttt{B}\}$ and order $\langle \texttt{A}, \texttt{B} \rangle$, if $P_1(11) = 1$ and $P_2(00) = 1$, then $P_1, P_2 \in \mathsf{Sat}(p(\texttt{A} \wedge \texttt{B}) = 1) \vee p(\neg\texttt{A} \wedge \neg\texttt{B}) = 1)$. For a graph s.t. $\mathsf{Nodes}(\mathcal{G}) = \{\texttt{C}\}$, if $P_3(1) = 0.5$ and $P_4(1) = 0.6$, then $P_3 \notin \mathsf{Sat}(p(\texttt{C}) > 0.5)$ and $P_4 \in \mathsf{Sat}(p(\texttt{C}) > 0.5)$.

The following two propositions show the semantical equivalence between the basic epistemic formulae and show certain simple relations between the formulae and the distributions satisfying them:

Proposition 3.5. *The following hold:*

- $\mathsf{Sat}(p(\alpha) \geq x) = \mathsf{Sat}((p(\alpha) = x) \vee (p(\alpha) > x))$,
- $\mathsf{Sat}(p(\alpha) \leq x) = \mathsf{Sat}((p(\alpha) = x) \vee (p(\alpha) < x))$,
- $\mathsf{Sat}(p(\alpha) \neq x) = \mathsf{Sat}(\neg(p(\alpha) = x))$,
- $\mathsf{Sat}(p(\alpha) \not> x) = \mathsf{Sat}(\neg(p(\alpha) > x))$,

- $\mathsf{Sat}(p(\alpha) \nless x) = \mathsf{Sat}(\neg(p(\alpha) < x))$.

Proposition 3.6. *The following hold:*

- *For* $x \in (0,1]$, $\mathsf{Sat}(p(\bot) = x) = \varnothing$.
- $\mathsf{Sat}(p(\top) = 1) = \mathsf{Dist}(\mathcal{G})$.
- *For any term* α, $\mathsf{Sat}(p(\alpha) \leq 1) = \mathsf{Dist}(\mathcal{G})$ *and* $\mathsf{Sat}(p(\alpha) \geq 0) = \mathsf{Dist}(\mathcal{G})$.
- *When* $x \neq y$, $\mathsf{Sat}(p(\alpha) = x \wedge p(\alpha) = y) = \varnothing$.
- *When* $\vdash \alpha \leftrightarrow \beta$, $\mathsf{Sat}(p(\alpha) = x) = \mathsf{Sat}(p(\beta) = x)$.

We can now define various constraints. A constraint is, in principle, any basic formula that concerns at least one argument. However, it also makes sense to distinguish formulae that have certain patterns. In particular, we can consider target and source constraints. The first grasp situations in which we want to constrain the belief in one argument based on its (possibly many) parents, while the second handle cases in which we constrain the belief in (possibly many) arguments based on a shared parent.

Definition 3.7. Let $\psi \in \mathsf{BFormulae}(\mathcal{G})$ be a basic epistemic formula and $X = \mathsf{FArgs}(\psi)$ the set of arguments appearing in ψ. A **basic constraint** is a basic formula $\psi \in \mathsf{BFormulae}(\mathcal{G})$ s.t. $X \neq \varnothing$. A **basic target constraint** is a basic constraint ψ for which there exists $\mathtt{A} \in X$ s.t. $(X \setminus \{\mathtt{A}\}) \subseteq \mathsf{Parent}(\mathtt{A})$. A target constraint is **full** iff there exists $\mathtt{A} \in X$ s.t. $X \setminus \{\mathtt{A}\} = \mathsf{Parent}(\mathtt{A}) \setminus \{\mathtt{A}\}$. A **basic source constraint** is a basic constraint ψ for which there exists $\mathtt{A} \in X$ s.t. $\mathtt{A} \in \bigcap_{\mathtt{B} \in (X \setminus \{\mathtt{A}\})} \mathsf{Parent}(\mathtt{B})$. A source constraint is **full** iff there exists $\mathtt{A} \in X$ s.t. $X \setminus \{\mathtt{A}\} = \{\mathtt{B} \mid (\mathtt{A},\mathtt{B}) \in \mathsf{Arcs}(\mathcal{G}), \mathtt{A} \neq \mathtt{B}\}$.

From now on we will denote the set of basic constraints associated with a given graph with $\mathsf{BCon}(\mathcal{G})$.

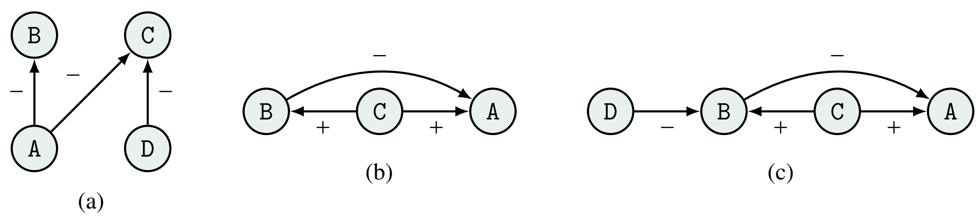


Figure 8: Examples of labelled graphs

Example 12. Let us consider the graph depicted in Figure 8a. The formula $(p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{B}) \leq 0.5) \wedge (p(\mathtt{D}) > 0.5 \rightarrow p(\mathtt{C}) \leq 0.5)$ is a constraint, but neither a target nor a source one. The formula $p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{B}) \leq 0.5 \vee p(\mathtt{C}) \leq 0.5$ is a source constraint as A is a parent of both B and C. It is also full. The formula $p(\mathtt{C}) > 0.5 \rightarrow p(\mathtt{A}) = 0 \wedge p(\mathtt{D}) \leq 0.5$ is a full target constraint – in this case, A and D are parents of C. Finally, $p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{C}) \leq 0.5$ is both a source and a target constraint, though not a full source or a full target one.

Example 13. Consider Figure 8b. For this, we could consider the following set of constraints. But if we assume $p(\mathtt{C}) > 0.5$ holds, then the set of constraints has no satisfying distributions.

$$\{p(\mathtt{B}) > 0.5 \rightarrow p(\mathtt{A}) \leq 0.5 \quad p(\mathtt{C}) > 0.5 \rightarrow p(\mathtt{A}) > 0.5 \quad p(\mathtt{C}) > 0.5 \rightarrow p(\mathtt{B}) > 0.5\}$$

Now consider Figure 8c . For this, we could consider the following set of constraints. Hence, if we assume $p(\mathtt{C}) > 0.5$ holds, then there are satisfying distributions.

$$\{p(\mathtt{B}) > 0.5 \wedge p(\mathtt{D}) \leq 0.5 \rightarrow p(\mathtt{A}) \leq 0.5 \quad p(\mathtt{C}) > 0.5 \rightarrow p(\mathtt{A}) > 0.5 \quad p(\mathtt{C}) > 0.5 \rightarrow p(\mathtt{B}) > 0.5\}$$

3.1.2 Valued Constraints

Although there is a wide range of postulates [71] that the basic constraints can grasp, there are also those that cannot be handled. Consider two arguments $\mathtt{A}, \mathtt{B} \in \mathsf{Nodes}(\mathcal{G})$ s.t. $\mathtt{A}$ is an attacker of $\mathtt{B}$. The rational postulate can then be easily represented as $p(\mathtt{A}) \geq 0.5 \rightarrow p(\mathtt{B}) < 0.5$. However, the optimistic postulate, which can be written as $p(\mathtt{A}) + p(\mathtt{B}) \geq 1$, does not fit the pattern of the basic constraint (see Definition 2.8). Unless a fixed set of values is assumed and we enumerate every possible case, expressing this property is extremely difficult due to the fact that the probabilities can be any real values from the $[0,1]$ interval. In principle, the valued family of postulates is not handled by the previously introduced language. Therefore, we also introduce the notion of a valued constraint, allowing us to express such properties. We start by creating the definitions of operational formulae, which, to put it simply, express subtraction and addition of probabilities. From these formulae we can create the valued epistemic atoms. To put it simply, the $p(\alpha)$ component of a basic epistemic atom is replaced by an operational formulae:

Definition 3.8. An **operational formula** φ is of the form $\varphi := \{p(\alpha) \mid \varphi{+}\varphi' \mid \varphi{-}\varphi'\}$ where $\alpha \in \mathsf{Terms}(\mathcal{G})$. The set of operational formulae of $\mathcal{G}$ is denoted $\mathsf{OFormulae}(\mathcal{G})$.

Definition 3.9. A **valued epistemic atom** is of the form $\alpha \# x$ where $\# \in \{=, \neq, \geq, \leq, >, <\}$, $x \in [0,1]$ and $\alpha \in \mathsf{OFormulae}(\mathcal{G})$. A **valued epistemic formula** is a Boolean combination of valued atoms (i.e. if ϕ is a valued epistemic atom, then it is a valued epistemic formula, and if ϕ and ψ are valued epistemic formulae, then each of $\phi \wedge \psi$, $\phi \vee \psi$ and $\neg\phi$ is a valued epistemic formula). Let $\mathsf{VFormulae}(\mathcal{G})$ be the set of valued epistemic formulae that can be formed from the arguments in $\mathcal{G}$.

Example 14. For $\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D} \in \mathsf{Nodes}(\mathcal{G})$, $\psi : p(\mathtt{A} \wedge \mathtt{B}) - p(\mathtt{C}) - p(\mathtt{D}) > 0$ is an example of a valued epistemic formula. The terms of that formula are $\mathsf{FTerms}(\psi) = \{\mathtt{A} \wedge \mathtt{B}, \neg\mathtt{A} \wedge \neg\mathtt{B}, \mathtt{C}, \mathtt{D}\}$, the arguments appearing in them are $\mathsf{FArgs}(\psi) = \{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}\}$, and the numerical values of the formula are $\mathsf{Num}(\psi) = \{0\}$.

Another example of a valued epistemic formula is $\phi : p(\mathtt{A} \vee \mathtt{B}) - p(\mathtt{A}) - p(\mathtt{B}) + p(\mathtt{A} \wedge \mathtt{B}) = 0$, which is one of the laws of probability. The terms of that formula are $\mathsf{FTerms}(\phi) = \{\mathtt{A} \wedge \mathtt{B}, \mathtt{A} \vee \mathtt{B}, \mathtt{A}, \mathtt{B}\}$, the arguments appearing in them are $\mathsf{FArgs}(\phi) = \{\mathtt{A}, \mathtt{B}\}$, and the numerical values of the formula are $\mathsf{Num}(\phi) = \{0\}$.

Additionally, we define a function for retrieving the sequence of arithmetic operators appearing in a given epistemic atom:

Definition 3.10. Let $\varphi = p(\alpha_1) *_1 p(\alpha_2) *_2 \ldots *_{m-1} p(\alpha_m)$ be an operational formula. Then $\mathsf{AOp}(\varphi) = (*_1, *_2, \ldots, *_{m-1})$ is the, possibly empty, **sequence of arithmetic operators** appearing in the operational formula f.

By abuse of notation, by $\mathsf{AOp}(\varphi)$ for an epistemic atom φ we will understand the sequence of operators of the operational formula of φ.

Definition 3.11. Let $\varphi : p(\alpha_1) *_1 p(\alpha_2) *_2 \ldots *_{m-1} p(\alpha_m)$, where $\alpha_i \in \mathsf{Terms}(\mathcal{G})$ and $*_i \in \{+, -\}$, be an operational formula. Let $x \in [0,1]$ and $\# \in \{=, \neq, \geq, \leq, >, <\}$. The **satisfying distributions** for a valued atom $\varphi \# x$ is defined as $\mathsf{Sat}(\varphi \# x) = \{P' \in \mathsf{Dist}(\mathcal{G}) \mid P'(\alpha_1) *_1 P'(\alpha_2) *_2 \ldots *_{m-1} P'(\alpha_m) \# x\}$.

The set of satisfying distributions for a valued formula is as follows where ϕ and ψ are valued formulae:

- $\mathsf{Sat}(\phi \wedge \psi) = \mathsf{Sat}(\phi) \cap \mathsf{Sat}(\psi)$;
- $\mathsf{Sat}(\phi \vee \psi) = \mathsf{Sat}(\phi) \cup \mathsf{Sat}(\psi)$; and
- $\mathsf{Sat}(\neg\phi) = \mathsf{Sat}(\top) \setminus \mathsf{Sat}(\phi)$.

Also, for a set of valued formulae $\Phi = \{\phi_1, \ldots, \phi_n\}$, the set of satisfying distributions is $\mathsf{Sat}(\Phi) = \mathsf{Sat}(\phi_1) \cap \ldots \cap \mathsf{Sat}(\phi_n)$.

Example 15. Let us consider the constraint $\psi : p(\mathtt{A} \wedge \mathtt{B}) - p(\mathtt{C}) - p(\mathtt{D}) > 0$ on a graph s.t. $\{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}\} = \mathsf{Nodes}(\mathcal{G})$ with the order $\langle \mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D} \rangle$. Consider a probability distribution P_1 s.t. $P_1(1100) > 0.7$. In this case, the total probabilities of $\mathtt{C}$ and $\mathtt{D}$ cannot exceed 0.3, which means that $P_1 \in \mathsf{Sat}(\psi)$. On the other hand, given a distribution P_2 s.t. $P_2(1100) = 0$, we can never find an assignment for $\mathtt{C}$ and $\mathtt{D}$ s.t. ψ is satisfied, hence $P_2 \notin \mathsf{Sat}(\psi)$.

Similarly as in the case of basic constraints, we distinguish between "general", target and source valued constraints. Again, a constraint is a formula that contains at least one argument, in the target constraint we can find an argument s.t. the remaining ones are its parents, and in the source we have an argument that is a parent of all the remaining arguments:

Definition 3.12. Let $\psi \in \mathsf{VFormulae}(\mathcal{G})$ be a valued formula and $X = \mathsf{FArgs}(\psi)$ the set of arguments appearing in ψ. A **valued constraint** is a valued formula $\psi \in \mathsf{VFormulae}(\mathcal{G})$ s.t. $X \neq \emptyset$. A **valued target constraint** is a valued constraint ψ for which there exists $\mathtt{A} \in X$ s.t. $(X \setminus \{\mathtt{A}\}) \subseteq \mathsf{Parent}(\mathtt{A})$. A target constraint is **full** iff there exists $\mathtt{A} \in X$ s.t. $X \setminus \{\mathtt{A}\} = \mathsf{Parent}(\mathtt{A}) \setminus \{\mathtt{A}\}$. A **valued source constraint** is a valued constraint ψ for which there exists $\mathtt{A} \in X$ s.t. $\mathtt{A} \in \bigcap_{\mathtt{B} \in (X \setminus \{\mathtt{A}\})} \mathsf{Parent}(\mathtt{B})$. A source constraint is **full** iff there exists $\mathtt{A} \in X$ s.t. $X \setminus \{\mathtt{A}\} = \{\mathtt{B} \mid (\mathtt{A},\mathtt{B}) \in \mathsf{Arcs}(\mathcal{G}), \mathtt{A} \neq \mathtt{B}\}$.

From now on we will denote the set of valued constraints associated with a given graph with $\mathsf{VCon}(\mathcal{G})$.

Example 16. Let $(\mathcal{G},\mathcal{L})$ be a labelled graph s.t. $\mathcal{G} = (\{\mathtt{A},\mathtt{B},\mathtt{C},\mathtt{D}\}, \{(\mathtt{A},\mathtt{B}),(\mathtt{A},\mathtt{C}),(\mathtt{D},\mathtt{C})\})$ and $\mathcal{L}$ assigns only the $-$ label, as depicted in Figure 8a. The formula $p(\mathtt{D})+p(\mathtt{B})+p(\mathtt{C})+p(\mathtt{A}) \leq 0.9$ is a valued constraint, but neither a target nor a source one. The formula $p(\mathtt{C}) + p(\mathtt{A}) + p(\mathtt{D}) \leq 1$ is a (full) target constraint, but not a source constraint. We can observe that $p(\mathtt{B}) + p(\mathtt{A}) \leq 1$ is both a source and a target constraint. Additionally, when seen as a target constraint, it is also full. Finally, $p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{B}) + p(\mathtt{C}) < 1$ is a (full) source constraint, but not a target one.

We can observe that a basic atom is a special case of a valued atom. Given that the construction of the epistemic formulae from these atoms is the same, every basic formula is a valued formula, and every basic constraint is a valued constraint. Please note that this relation is, in general, one way, as it is easy to create valued atoms not conforming to the basic atom patterns:

Proposition 3.13. *Let $(\mathcal{G},\mathcal{L})$ be a labelled graph. The following hold:*

- *every basic epistemic atom is a valued epistemic atom, but not necessarily vice versa*
- *every basic epistemic formula is a valued epistemic formula, but not necessarily vice versa*
- *every basic constraint is a valued constraint, but not necessarily vice versa*
- *every basic target constraint is a valued target constraint, but not necessarily vice versa*
- *every basic source constraint is a valued source constraint, but not necessarily vice versa*

Example 17. We can consider a graph consisting of two arguments $\mathtt{A},\mathtt{B}$ s.t. $\mathcal{L}((\mathtt{A},\mathtt{B})) = \{-\}$. A valued constraint expressing the optimistic postulate would be $p(\mathtt{B}) + p(\mathtt{A}) \geq 1$. Unfortunately, it does not fit the format of the basic epistemic atom, and trying to create an equivalent basic constraint would lead to creating (uncountably) infinitely many postulates for all $x \in [0,1]$ that enumerate all the values that $p(\mathtt{B})$ may take and constraining $p(\mathtt{A})$ appropriately, i.e. for every $x \in [0,1]$, we have a constraint $p(\mathtt{B}) = x \rightarrow p(\mathtt{A}) \geq y$ where $y = 1 - x$.

3.2 Epistemic Graphs

In the previous section we have introduced two types of epistemic constraints, the basic and the valued one. With this, we can now introduce our new framework – the epistemic graph – which, to put it simply, is a labelled graph equipped with a set of such constraints:

Definition 3.14. An **epistemic graph** is a tuple $(\mathcal{G},\mathcal{L},\mathcal{C})$ where $(\mathcal{G},\mathcal{L})$ is a labelled graph, and $\mathcal{C} \subseteq \mathsf{VCon}(\mathcal{G})$ is a set of epistemic constraints associated with the graph.

Please note that the graph (and its labelling, which we will discuss in Section 5.3) are not necessarily induced by the constraints and therefore contain additional information. The actual direction of the edges in the graph is not necessarily derivable from $\mathcal{C}$. For example, if we had two arguments $\mathtt{A}$ and $\mathtt{B}$ connected by an edge, a constraint of the form $p(\mathtt{A}) < 0.5 \vee p(\mathtt{B}) < 0.5$ would not tell us the direction of this edge.

The constraints may also involve unrelated arguments, similarly as in [31]. Furthermore, the constraints can be (full) target–style, (full) parent–style, or neither.

We will now present a number of examples of how epistemic graphs can be defined. This includes the cases in which parent arguments influence target arguments and vice versa, the constraints with Boolean combinations of terms in the atoms, and the cases in which constraints may be between arguments that are not directly connected.

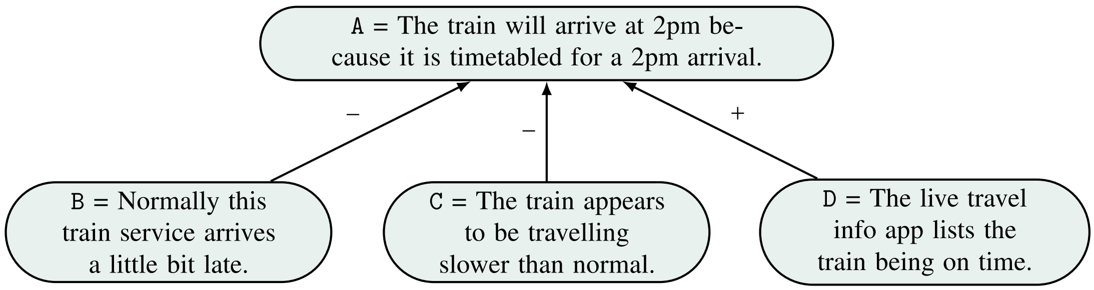


Figure 9: Argument graph concerning the arrival time of a train journey. Edges labelled with − represent attack and edges labelled with + represent support.

Example 18. Let us continue Example 1 and consider two more passengers, Annie and Alfred, travelling by the train. Their graph is presented in Figure 9. Let us assume they decode the claims of arguments B and C as "therefore, the train might be late" and of D as "therefore, the train will arrive on time".

We first consider Annie, who is a very skeptical person. Therefore, she will have a reasonable belief in the train arriving on time if and only if she strongly believes in the information provided by the service app and strongly disbelieves either of the attackers being right. Thus, she can see B, C and D influencing A in the following manner:

$$p(\mathtt{D}) \geq 0.9 \land p(\mathtt{B} \lor \mathtt{C}) \leq 0.1 \leftrightarrow p(\mathtt{A}) \geq 0.7$$

Let us now consider Alfred. As long as he jointly believes B and C (i.e. that the train is normally late and that it appears to be travelling slowly), he will disbelieve that the train arrives on time, irrespective of what the app says. However, if he jointly disbelieves the attackers, then he will believe the train being on time as much as he believes the app being right:

$$p(\mathtt{B} \land \mathtt{C}) > 0.5 \rightarrow p(\mathtt{A}) < 0.5$$
$$p(\mathtt{B} \lor \mathtt{C}) \leq 0.5 \rightarrow p(\mathtt{A}) - p(\mathtt{D}) = 0$$

Example 19 (Adapted from [58])**.** Let us consider the following arguments:

- A = "Patient X needs to undergo medical procedure P."
- B = "Medical procedure P is very expensive."
- C = "The procedure is necessary to save X's life."
- D = "The patient's condition can be managed with drug D, therefore the procedure is not necessary."

Arguments B and A can be seen as mutually attacking, and D attacking C. However, we can observe that C is not entirely a counterargument for B, as the necessity of the procedure does not contradict the fact that it is expensive. It is perhaps more accurately modelled using attacks on attacks [10, 39, 58], where the importance of the surgery overrides the attack from B to A. However, one can also model it via a particular form of support, as done in abstract dialectical frameworks [67, 68], which is also possible in epistemic graphs. Thus, the relations between the aforementioned arguments can be written down as epistemic constraints in the following manner, where φ_1 and φ_3 state that a B and C should be disbelieved if their attackers are believed, and φ_2 states that A should be disbelieved if B is believed and the argument that could "overrule" its attack is disbelieved:

- $\varphi_1 : p(\texttt{A}) > 0.5 \rightarrow p(\texttt{B}) < 0.5$
- $\varphi_2 : (p(\texttt{B}) > 0.5 \wedge p(\texttt{C}) < 0.5) \rightarrow p(\texttt{A}) < 0.5$
- $\varphi_3 : p(\texttt{D}) > 0.5 \rightarrow p(\texttt{C}) < 0.5$

This scenario can be then represented with the labelled graph in Figure 10 which does not use recursive edges.

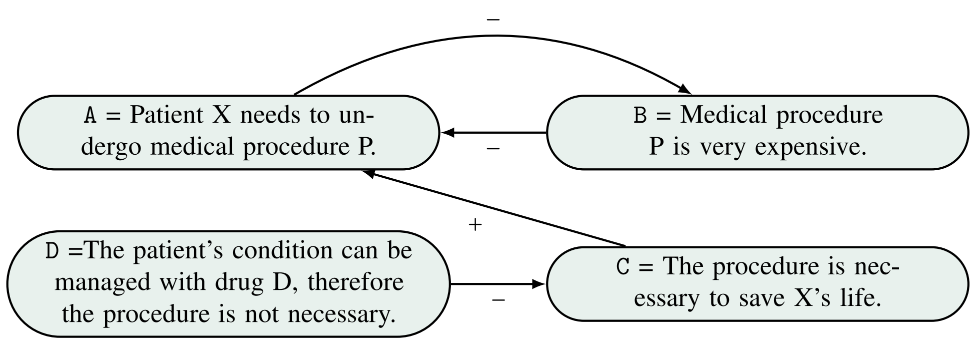


Figure 10: Medical procedure graph. The + labels denote support and − denote attack.

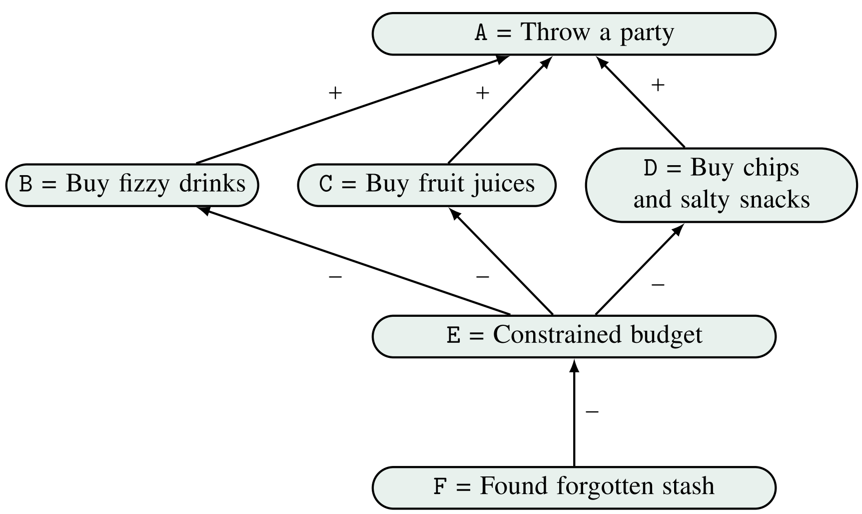


Figure 11: Party organization graph. The + labels denote support and − denote attack.

Example 20. Let us consider an example in which Mary and Jane are organizing a small party at the student dormitory. Although the guests will bring some beer, Mary and Jane need to buy some non–alcoholic drinks and snacks. This can be represented with arguments A, B, C and D as seen in Figure 11 and expressed with the following constraints:

- $\varphi_1 = (p(\texttt{B}) > 0.5 \vee p(\texttt{C}) > 0.5) \wedge p(\texttt{D}) > 0.5 \rightarrow p(\texttt{A}) > 0.5$
- $\varphi_2 = (p(\texttt{B}) < 0.5 \wedge p(\texttt{C}) < 0.5) \vee p(\texttt{D}) < 0.5 \rightarrow p(\texttt{A}) < 0.5$

We can observe that B, C and D are supporters of A in the sense that the acceptance of A requires the acceptance of D and B or C.

Let us assume that Mary and Jane realize that their budget is somewhat limited. We could create a constraint stating that at least one of the items has to be rejected:

- $\varphi_3 : p(\mathtt{A}) < 0.5 \lor p(\mathtt{B}) < 0.5 \lor p(\mathtt{C}) < 0.5$

However, instead of this, we can also decide to represent the budget limitations as an argument E and replace φ_3 with φ_3':

- $\varphi_3' = p(\mathtt{E}) > 0.5 \rightarrow p(\mathtt{B}) < 0.5 \lor p(\mathtt{C}) < 0.5 \lor p(\mathtt{D}) < 0.5$
- $\varphi_4' = p(\mathtt{E}) > 0.5$

We can observe that in this case, the relation between E and B, C and D is more attacking, in the sense that acceptance of E leads to the rejection of at least one of B, C and D.

Although the former solution is more concise, the latter also has its benefits. Let us assume that Mary now finds some spare money in her backpack and they can afford to buy all of the items. Thus, we add argument F, and the constraint φ_4' will need to be replaced:

- $\varphi_4'' = p(\mathtt{F}) > 0.5 \rightarrow p(\mathtt{E}) < 0.5$
- $\varphi_5'' = p(\mathtt{F}) > 0.5$

Clearly, the relation between F and E is conflicting.

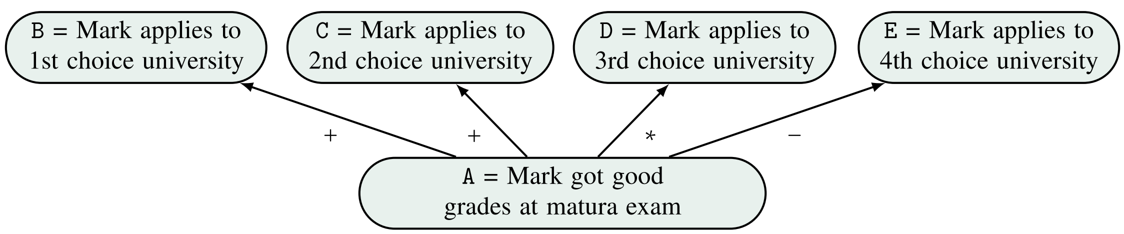


Figure 12: Mark's university choice graph. The + labels denote support, − attack, and $*$ dependency.

Example 21. Let us consider the graph depicted in Figure 12. Given the rules in his country, Mark has written the matura exam (national exam after high school allowing a person to apply to a university) and can now register for up to two universities that interest him. He will be accepted or rejected once the exam results are in. We create the following constraints expressing what Mark plans to do:

- If Mark strongly disbelieves that he will get good grades, he will apply only to his 4th choice university:

  $p(\mathtt{A}) \leq 0.2 \rightarrow p(\mathtt{B}) < 0.5 \land p(\mathtt{C}) < 0.5 \land p(\mathtt{D}) < 0.5 \land p(\mathtt{E}) > 0.5$

- If Mark moderately does not believe that he will get good grades, he will apply only to his 3rd and 4th choice universities:

  $p(\mathtt{A}) > 0.2 \land p(\mathtt{A}) \leq 0.5 \rightarrow p(\mathtt{B}) < 0.5 \land p(\mathtt{C}) < 0.5 \land p(\mathtt{D}) > 0.5 \land p(\mathtt{E}) > 0.5$

- If Mark moderately believes his grades will be good, he will apply only to his 2nd and 3rd choice universities:

  $p(\mathtt{A}) > 0.5 \land p(\mathtt{A}) < 0.8 \rightarrow p(\mathtt{B}) < 0.5 \land p(\mathtt{C}) > 0.5 \land p(\mathtt{D}) > 0.5 \land p(\mathtt{E}) < 0.5$

- If Mark strongly believes he will get good grades, he will apply only to his 1st and 2nd choice universities:

  $p(\mathtt{A}) \geq 0.8 \rightarrow p(\mathtt{B}) > 0.5 \land p(\mathtt{C}) > 0.5 \land p(\mathtt{D}) < 0.5 \land p(\mathtt{E}) < 0.5$

We can consider the relation between A and E to be conflicting, as once A is believed we disbelieve E. Given that believing A (to a sufficiently high degree) also leads to believing B and C, the relations between these arguments can be seen as supporting. However, the interaction between A and D cannot be clearly classified as supporting or attacking, given that as the belief in A increases, D can be disbelieved, believed, and then disbelieved again.

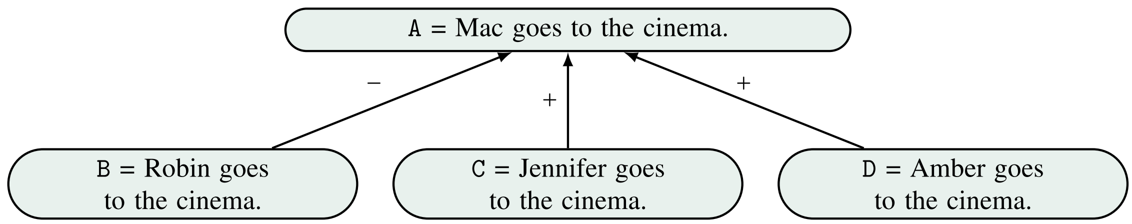


Figure 13: Argument graph concerning Mac going to the cinema. Edges labelled with − represent attack and edges labelled with + represent support.

Example 22. Let us consider a very simple example concerning Mac going to the cinema (see Figure 13). Mac will go to the cinema if Jennifer and Amber go and Robin does not join them. This could, for example, be expressed with the following constraint:

$$\varphi : (p(\mathtt{B}) < 0.5 \wedge p(\mathtt{C}) > 0.5 \wedge p(\mathtt{D}) > 0.5) \rightarrow p(\mathtt{A}) > 0.5$$

However, the following requirement can be seen as stronger, ensuring that not only we believe that Jennifer and Amber will come to the cinema independently of each other, but that they will come together:

$$\psi : (p(\mathtt{B}) < 0.5 \wedge p(\mathtt{C} \wedge \mathtt{D}) > 0.5) \rightarrow p(\mathtt{A}) > 0.5$$

We can observe that if ψ is satisfied, then so is φ, but not necessarily the other way around.

These examples have shown that the epistemic graphs are quite flexible in representing various restrictions on beliefs. However, given the freedom we have in defining constraints, we can create epistemic graphs in which the constraints do not really reflect the structure of the graph and vice versa. Moreover, a probability distribution satisfying our requirements may be further refined in various ways, independently of the graph in question. Thus, in the next section we would like to explore various types of epistemic semantics which, depending on the situation, may or may not be desirable.

3.3 Epistemic Semantics

The examples from the previous section show us that the epistemic graphs offer us a number of ways in which we can decide how much a given argument should be believed or disbelieved depending on the remaining arguments. Although the main aim of the semantics of the epistemic graphs is to select those probability distributions that satisfy our requirements, one can also enforce additional restrictions for refining the sets of acceptable distributions, on which we will focus in this section. First of all, the simplest possible semantics is the one that associates a given graph with the set of distributions satisfying its constraints:

Definition 3.15. For an epistemic graph $(\mathcal{G}, \mathcal{L}, \mathcal{C})$, a distribution $P \in \mathsf{Dist}(\mathcal{G})$ meets the **satisfaction semantics** iff $P \in \mathsf{Sat}(\mathcal{C})$.

We will say that a framework is constraint consistent iff $\mathsf{Sat}(\mathcal{C}) \neq \emptyset$, i.e. the satisfaction semantics produces at least one distribution for this graph.

Example 23. There is no distribution satisfying the set of constraints $\{p(\mathtt{A}) > 0.8, p(\mathtt{A}) \leq 0.7\}$, independently of the graph in question.

Given that an inconsistent graph is not particularly interesting, we will aim at specifying epistemic graphs that have consistent constraints. However, we would like to note that this may not always be possible and that inconsistency does not necessarily mean that the constraints are not rational. For example, the stable semantics [28] for argumentation graphs does not always produce any extensions, and this is a result of the restrictive nature of this semantics. We can therefore expect that epistemic graphs aiming to emulate these graphs and stability may have inconsistent sets of constraints.

Various properties which can be quite useful concern minimizing or maximizing certain aspects of a distribution. Similarly as in other types of argumentation semantics, we can aim to maximize or minimize the set of arguments that are believed up to any degree, disbelieved up to any degree, or both, which corresponds to comparing the undecided assignments. These approaches can be further refined to take the actual degrees into account as well. For example, in some scenarios a distribution s.t. $P(\mathtt{A}) = P(\mathtt{B}) = 1$ and $P(\mathtt{C}) = 0.49$ might be preferable to one s.t. $P(\mathtt{A}) = P(\mathtt{B}) = P(\mathtt{C}) = 0.51$, even if the actual number of believed arguments is smaller. Thus, we also consider the belief maximizing and minimizing approaches, based on the notion of entropy:

Definition 3.16. For a probability distribution P, the **entropy** $H(P)$ of P is defined as

$$H(P) = -\sum_{\Gamma\subseteq \mathsf{Nodes}(\mathcal{G})} P(\Gamma) log P(\Gamma)$$

with $0\log 0 = 0$.

The entropy measures the amount of indeterminateness of a probability distribution P. A probability function P_1 that describes absolute certain knowledge, i. e. $P_1(\Gamma) = 1$ for some $\Gamma \subseteq \mathsf{Nodes}(\mathcal{G})$ and $P_1(\Gamma') = 0$ for every other $\Gamma' \subseteq \mathsf{Nodes}(\mathcal{G})$, yields minimal entropy $H(P_1) = 0$. The uniform probability function P_0 with $P_0(\Gamma) = \frac{1}{|2^{\mathsf{Nodes}(\mathcal{G})}|}$ for every $\Gamma \subseteq \mathsf{Nodes}(\mathcal{G})$ yields maximal entropy $H(P_0) = -\log {}^1/_{|2^{\mathsf{Nodes}(\mathcal{G})}|}$.

With this at hand, we can now define the following properties:

Definition 3.17. Given a set of distributions $S \subseteq \mathsf{Dist}(\mathcal{G})$, a distribution $P \in \mathsf{Dist}(\mathcal{G})$ is:

- **acceptance maximizing** w.r.t. S iff there is no $P' \in S$ s.t. $\{\mathtt{A} \mid P(\mathtt{A}) > 0.5\} \subset \{\mathtt{A} \mid P'(\mathtt{A}) > 0.5\}$
- **acceptance minimizing** w.r.t. S iff there is no $P' \in S$ s.t. $\{\mathtt{A} \mid P'(\mathtt{A}) > 0.5\} \subset \{\mathtt{A} \mid P(\mathtt{A}) > 0.5\}$
- **rejection maximizing** w.r.t. S iff there is no $P' \in S$ s.t. $\{\mathtt{A} \mid P(\mathtt{A}) < 0.5\} \subset \{\mathtt{A} \mid P'(\mathtt{A}) < 0.5\}$
- **rejection minimizing** w.r.t. S iff there is no $P' \in S$ s.t. $\{\mathtt{A} \mid P'(\mathtt{A}) < 0.5\} \subset \{\mathtt{A} \mid P(\mathtt{A}) < 0.5\}$
- **information maximizing** w.r.t. S iff there is no $P' \in S$ s.t. $\{\mathtt{A} \mid P(\mathtt{A}) \neq 0.5\} \subset \{\mathtt{A} \mid P'(\mathtt{A}) \neq 0.5\}$, $\{\mathtt{A} \mid P(\mathtt{A}) < 0.5\} \subseteq \{\mathtt{A} \mid P'(\mathtt{A}) < 0.5\}$ and $\{\mathtt{A} \mid P(\mathtt{A}) > 0.5\} \subseteq \{\mathtt{A} \mid P'(\mathtt{A}) > 0.5\}$
- **information minimizing** w.r.t. S iff there is no $P' \in S$ s.t. $\{\mathtt{A} \mid P'(\mathtt{A}) \neq 0.5\} \subset \{\mathtt{A} \mid P(\mathtt{A}) \neq 0.5\}$, $\{\mathtt{A} \mid P'(\mathtt{A}) < 0.5\} \subseteq \{\mathtt{A} \mid P(\mathtt{A}) < 0.5\}$ and $\{\mathtt{A} \mid P'(\mathtt{A}) > 0.5\} \subseteq \{\mathtt{A} \mid P(\mathtt{A}) > 0.5\}$
- **undecided maximizing** w.r.t. S iff there is no $P' \in S$ s.t. $\{\mathtt{A} \mid P(\mathtt{A}) = 0.5\} \subset \{\mathtt{A} \mid P'(\mathtt{A}) = 0.5\}$
- **undecided minimizing** w.r.t. S iff there is no $P' \in S$ s.t. $\{\mathtt{A} \mid P'(\mathtt{A}) = 0.5\} \subset \{\mathtt{A} \mid P(\mathtt{A}) = 0.5\}$
- **belief maximizing** w.r.t. S iff there is no $P' \in S$ s.t. $H(P') < H(P)$
- **belief minimizing** w.r.t. S iff there is no $P' \in S$ s.t. $H(P) < H(P')$

Given that the purpose of an epistemic semantics is to grasp various optional properties whenever and however they are needed, a new semantics can be defined “on top” of a previous semantics, such as the satisfaction semantics. We can therefore propose the following, parameterized definition:

Definition 3.18. Let $(\mathcal{G},\mathcal{L},\mathcal{C})$ be an epistemic graph and $\mathcal{R}$ the set of distributions associated with it according to a given semantics σ. A distribution $P \in \mathsf{Dist}(\mathcal{G})$ meets the **σ–acceptance maximizing semantics** iff $P \in \mathcal{R}$ and P is acceptance maximizing w.r.t. $\mathcal{R}$.

Further minimizing and maximizing semantics can be created in the same manner, as we will show in Exampled 24 to 26.

There are, of course, additional properties we may want to impose in order to refine the distributions produced by the constraints associated with a framework. In particular, we may want to limit the values that the distribution may take. Certain epistemic postulates already did that, for example the ternary one as recalled in Definition 2.8. With some exceptions, most of them can be expressed as basic constraints in the epistemic graphs. However, one has to observe that in a sense, they are completely independent of the underlying structure of the graph. Thus, we believe it is more appropriate to view them as additional, optional properties:

Definition 3.19. A distribution P is:

- **minimal** iff for every $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, $P(\mathtt{A}) = 0$
- **maximal** iff for every $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, $P(\mathtt{A}) = 1$
- **neutral** iff for every $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, $P(\mathtt{A}) = 0.5$
- **ternary** iff for every $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, $P(\mathtt{A}) \in \{0, 0.5, 1\}$
- **non–neutral**[3] iff for every $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, $P(\mathtt{A}) \neq 0.5$
- **n–valued** iff $|\{x \mid \exists \mathtt{A} \in \mathsf{Nodes}(\mathcal{G}),\ P(\mathtt{A}) = x\}| = n$

We can therefore observe that there are various ways of refining probability distributions. We have proposed a number ways we can minimize or maximize different aspects of the distributions, and it is possible that on certain epistemic graphs they will coincide. However, as the following examples will show, all of the methods are in principle distinct.

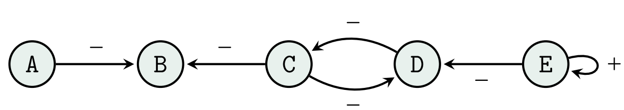


Figure 14: A labelled graph

Example 24. Let us consider the graph depicted in Figure 14 and the following set of constraints $\mathcal{C}$:

- $p(\mathtt{A}) \geq 0.5$
- $p(\mathtt{B}) + p(\mathtt{A}) \leq 1 \wedge p(\mathtt{B}) + p(\mathtt{C}) \leq 1$
- $p(\mathtt{B}) + p(\mathtt{A}) + p(\mathtt{C}) \geq 1$
- $p(\mathtt{C}) + p(\mathtt{D}) = 1$
- $p(\mathtt{D}) + p(\mathtt{C}) \leq 1 \wedge p(\mathtt{D}) + p(\mathtt{E}) \leq 1$
- $p(\mathtt{D}) + p(\mathtt{C}) + p(\mathtt{E}) \geq 1$
- $p(\mathtt{C}) > 0.5 \vee p(\mathtt{E}) > 0.5 \rightarrow p(\mathtt{D}) < 0.5$

In this case, a possible constraint for E would read "if E is believed, then E is believed and if E is disbelieved, then E is disbelieved", which is a tautology and its inclusion is not necessary.

There are infinitely many distributions satisfying our set of constraints, one of the reasons behind it being the fact that the belief in E is not constrained in any particular way. Let us therefore consider what happens when we add the following restrictions:

- ternary – the distributions satisfying the constraints and the ternary restriction are visible in Table 4
- non–neutral – there are again infinitely many distributions that satisfy the constraints and the non–neutral restriction. If we also consider the ternary one, then P_9, P_{11} and P_{14} meet our requirements.
- minimal – the minimal distribution (i.e. assigning 0 to every argument) does not meet the constraints

[3] Please note that this property was previously referred to as *binary* [71].

- maximal – the maximal distribution (i.e. assigning 1 to every argument) does not meet the constraints
- neutral – the neutral distribution meets the constraints (see P_1)
- n–valued – we can find an n–valued distribution for any $1 \leq n \leq 5$. For $n = 1$, we can consider P_1, for $n = 2$ we can consider P_2, P_3, P_4, P_9, P_{11} and P_{14}, and so on. For $n \geq 2$, there are infinitely many distributions that are n–valued – for example, any distribution in which all arguments but E are assigned 0.5 and the $0 \leq P(\mathtt{E}) < 0.5$ will satisfy the constraints and be 2–valued.

Let us now consider those distributions that meet the satisfaction semantics and are additionally:

- acceptance maximizing – although there are infinitely many distributions meeting these requirements, every such distribution follows one of the two patterns: either A, C and E are believed, or A and D are believed (see e.g. P_9 and P_{14})
- acceptance minimizing – there are again infinitely many distributions meeting these requirements, however, in all of them the set of believed arguments is empty (see e.g. P_1 to P_4)
- rejection maximizing – there are infinitely many distributions meeting these requirements, however, everyone of them follows one of the two patterns: either B, C and E are disbelieved, or B, D and E are disbelieved (see e.g. P_8, P_9, P_{11} and P_{14})
- rejection minimizing – a rejection minimizing distribution that at the same time satisfies the constraints will always lead to an neutral assignment as in P_1, i.e. one in which every argument is assigned 0.5
- undecided maximizing – an undecided maximizing distribution that at the same time satisfies our constraints will always lead to an neutral assignment as in P_1, i.e. one in which every argument is assigned 0.5
- undecided minimizing – there are infinitely many distributions meeting these requirements, but the patterns of which arguments are believed or disbelieved will be the same as in distributions P_9, P_{11} and P_{14}.
- information maximizing – in this particular example, they correspond to undecided minimizing distributions
- information minimizing – in this particular example, they correspond to undecided maximizing distributions
- belief maximizing – there are three belief maximizing distributions P_9, P_{11}, and P_{14}. In fact, whenever a set of arguments with each having probability 1 and all other arguments having probability 0 can be identified then this distribution can be modelled by setting the probability of this set to 1 and all other sets to 0. For example, P_9 can be defined as $P_9(\{\mathtt{A},\mathtt{D}\}) = 1$ and $P_9(\Gamma) = 0$ for all remaining sets, yielding minimal entropy $H(P_9) = 0$.
- belief minimizing – a belief minimizing distribution has maximal entropy among all distributions. Intuitively speaking, a belief minimizing distribution is as close as possible to the neutral distribution as possible. In fact, if the neutral distribution—i. e. the one assigning 0.5 to all arguments—is consistent with the constraints, as it is in this example, then it is also the unique belief minimizing distribution.

We can observe that, for example, P_{11} is information but not acceptance maximizing, P_8 is rejection but not information maximizing, and acceptance minimizing is not the same as rejection maximizing. Although in this particular case the information minimizing and rejection minimizing distributions coincide, this does not have to hold in general. We can easily imagine a graph with a single argument and a constraint stating that the argument can take any value from $\{0.1, 0.3, 0.5, 0.7\}$. Only the distributions assigning 0.5 to this argument are information minimizing, while both 0.5 and 0.7 probabilities are good for rejection minimizing.

Example 25. Let us come back to the framework $(\mathcal{G}, \mathcal{L})$ from Example 5, where $\mathcal{G} = (\{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}, \mathtt{E}\}, \{(\mathtt{A},\mathtt{B}), (\mathtt{C},\mathtt{B}), (\mathtt{C},\mathtt{D}), (\mathtt{D},\mathtt{C}), (\mathtt{D},\mathtt{E}), (\mathtt{E},\mathtt{E})\})$ and $\mathcal{L}$ assigns only the $-$ label to every edge. We can consider the following set of constraints $\mathcal{C}$:

- $\varphi_1 : p(\mathtt{A}) > 0.5$
- $\varphi_2 : (p(\mathtt{B}) > 0.5 \leftrightarrow (p(\mathtt{A}) < 0.5 \wedge p(\mathtt{C}) < 0.5)) \wedge (p(\mathtt{B}) < 0.5 \leftrightarrow (p(\mathtt{A}) > 0.5 \vee p(\mathtt{C}) > 0.5))$
- $\varphi_3 : (p(\mathtt{C}) > 0.5 \leftrightarrow p(\mathtt{D}) < 0.5) \wedge (p(\mathtt{C}) < 0.5 \leftrightarrow p(\mathtt{D}) > 0.5)$
- $\varphi_4 : (p(\mathtt{E}) > 0.5 \leftrightarrow (p(\mathtt{E}) < 0.5 \wedge p(\mathtt{D}) < 0.5)) \wedge (p(\mathtt{E}) < 0.5 \leftrightarrow (p(\mathtt{E}) > 0.5 \vee p(\mathtt{D}) > 0.5))$

We obtain three ternary satisfying distributions; P_1 s.t. $P_1(\mathtt{A}) = 1$, $P_1(\mathtt{B}) = 0$, $P_1(\mathtt{C}) = P_1(\mathtt{D}) = P_1(\mathtt{E}) = 0.5$, P_2 s.t. $P_2(\mathtt{A}) = P(\mathtt{C}) = 1$, $P_2(\mathtt{B}) = P_2(\mathtt{D}) = 0$, $P_2(\mathtt{E}) = 0.5$, and P_3 s.t. $P_3(\mathtt{A}) = P_3(\mathtt{D}) = 1$, $P_3(\mathtt{B}) = P_3(\mathtt{C}) = P_3(\mathtt{E}) = 0$. We can observe that P_2 and P_3 are information maximizing, but only P_3 is undecided minimizing.

We can now add the constraint $p(\mathtt{C}) > 0.5 \vee p(\mathtt{D}) > 0.5$ to $\mathcal{C}$, which eliminates P_1 as a satisfying distribution. We are therefore left with P_2 and P_3, both of which are information minimizing, but only P_2 is undecided maximizing.

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}	P_{13}	P_{14}
A	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	1	1	1	1	1	1
B	0.5	0.5	0	0	0	0	0.5	0	0	0	0	0	0	0
C	0.5	0.5	0.5	0.5	1	1	0	0	0	1	1	0.5	0.5	1
D	0.5	0.5	0.5	0.5	0	0	1	1	1	0	0	0.5	0.5	0
E	0.5	0	0.5	0	0.5	0	0	0	0	0.5	0	0.5	0	1

Table 4: Probability distributions meeting the ternary and satisfaction semantics from Example 24.

Figure 15: A labelled graph showing the distinction between undecided minimizing and belief maximizing

Example 26. Consider the graph depicted in Figure 15 and the following set of constraints $\mathcal{C} = \{p(\mathtt{A}) \geq 0.5, p(\mathtt{B}) + p(\neg\mathtt{B}) = 1\}$. For this graph, the distribution P'_1 defined via:

$$P'_1(\varnothing) = 0 \qquad P'_1(\{\mathtt{A}\}) = 0.5 \qquad P'_1(\{\mathtt{B}\}) = 0 \qquad P'_1(\{\mathtt{A},\mathtt{B}\}) = 0.5$$

satisfies the constraints and is both undecided minimizing (only B is undecided) and belief maximizing. However, the distribution P'_2 defined via:

$$P'_2(\varnothing) = 0.5 \qquad P'_2(\{\mathtt{A}\}) = 0 \qquad P'_2(\{\mathtt{B}\}) = 0 \qquad P'_2(\{\mathtt{A},\mathtt{B}\}) = 0.5$$

is belief maximizing but not undecided minimizing (both A and B are undecided).

3.4 Case Study

In order to illustrate how epistemic graphs could be acquired for an application, we consider using them as domain models in persuasion dialogue systems. Recent developments in computational argumentation are leading to a new generation of persuasion technologies [48]. An automated persuasion system (APS) is a system that can engage in a dialogue with a user (the persuadee) in order to convince the persuadee to do (or not do) some action or to believe (or not believe) something. The system achieves this by putting forward arguments that have a high chance of influencing the persuadee. In real-world persuasion, in particular in applications such as behaviour change, presenting convincing arguments, and presenting counterarguments

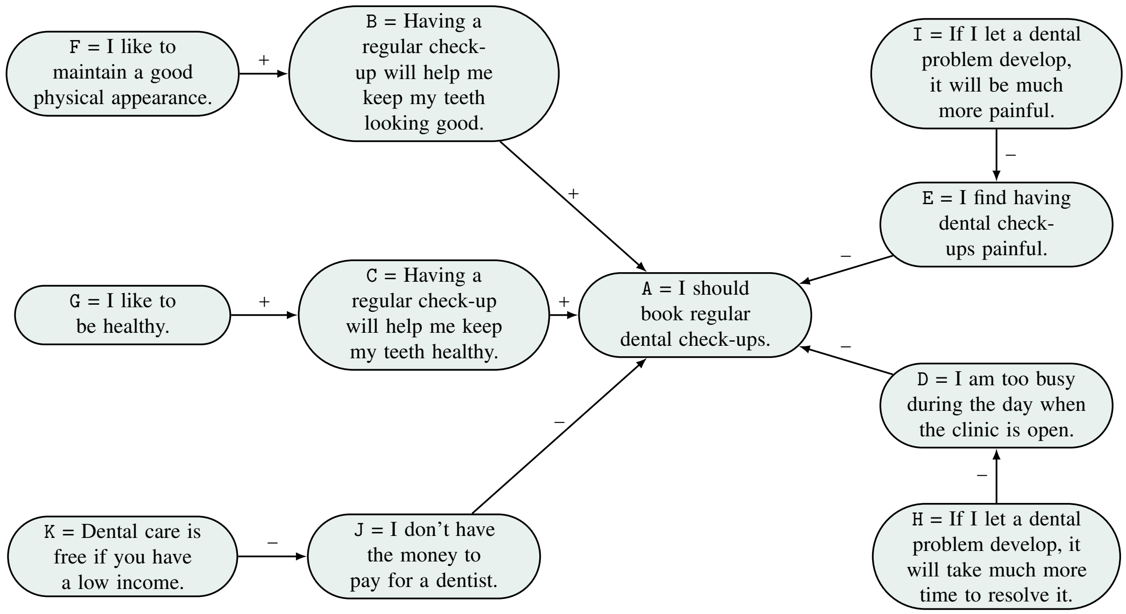


Figure 16: Epistemic graph for the domain model for a case study on encouraging people to take regular dental check-ups.

to the user's arguments, is critically important. For example, for a doctor to persuade a patient to drink less alcohol, that doctor has to give good arguments why it is better for the patient to drink less, and how (s)he can achieve this.

Two important features of an APS are the domain model and the user model, which are closely related, and together are harnessed by the APS strategy for optimizing the choice of move in a persuasion dialogue.

Domain model This contains the arguments that can be presented in the dialogue by the system, and it also contains the arguments that the user may entertain. Some arguments will attack other arguments, and some arguments will support other arguments. As we will see, the domain model can be represented by an epistemic graph.

User model This contains information about the user that can be utilized by the system in order to choose the most beneficial actions. The information in the user model is what the system believes is true about that user. A key dimension that we consider in the user model is the belief that the user may have in the arguments, and as the dialogue proceeds, the model can be updated [47] based on the results of the queries and of the arguments posited.

By using an epistemic graph to represent the domain model, and a probability distribution over arguments to represent the user model, we can have a tight coupling of the two kinds of model. Furthermore, the probability distribution can be harnessed directly in a decision-theoretic approach to optimize the choice of move [43].

To illustrate the use of epistemic graphs for domain/user modelling, we consider a case study in behaviour change. The aim of this behaviour change application is to persuade users to book a regular dental check-up.

Example 27. The following are a set of constraints for the epistemic graph in Figure 16. We explain these constraints as follows:

1. This constraint states that if the attacker $\mathtt{J}$ is disbelieved, and the supporters $\mathtt{B}$ and $\mathtt{C}$ are disbelieved, then $\mathtt{A}$ is in $(0.5, 1]$.

$$(p(\mathtt{B}) \leq 0.5 \wedge p(\mathtt{C}) \leq 0.5 \wedge p(\mathtt{J}) \leq 0.5) \rightarrow p(\mathtt{A}) > 0.5$$

2. This constraint states that if either of the attackers D or E is disbelieved, and attacker J is disbelieved, and a supporter (either B or C) is believed greater than 0.7, then A is in $(0.7, 1]$.

$$((p(\texttt{C}) > 0.7 \vee p(\texttt{B}) > 0.7) \wedge (p(\texttt{D}) \leq 0.5 \vee p(\texttt{E}) \leq 0.5) \wedge p(\texttt{J}) \leq 0.5) \rightarrow p(\texttt{A}) > 0.7$$

3. This constraint states that if attackers D and E are believed, but attacker J is not believed, then the belief in A is in $(0.4, 0.5]$.

$$(p(\texttt{D}) > 0.5 \wedge p(\texttt{E}) > 0.5 \wedge p(\texttt{J}) \leq 0.5) \rightarrow (p(\texttt{A}) > 0.4 \wedge p(\texttt{A}) \leq 0.5)$$

4. This constraint states that if attacker J is believed, then the belief in A is in $[0, 0.4)$.

$$p(\texttt{J}) > 0.5 \rightarrow p(\texttt{A}) \leq 0.4$$

5. This constraint states that if belief in attacker $\mathcal{G}$ is in $(0.5, 0.7]$, then the belief in C is in $(0.5, 0.7]$.

$$p(\texttt{G}) > 0.5 \wedge p(\texttt{G}) \leq 0.7 \rightarrow p(\texttt{C}) > 0.5 \wedge p(\texttt{C}) \leq 0.7$$

6. This constraint states that if belief in attacker $\mathcal{G}$ is in $(0.7, 1.0]$, then the belief in C is in $(0.7, 1.0)$.

$$p(\texttt{G}) > 0.7 \rightarrow p(\texttt{C}) > 0.7$$

7. This constraint states that if the belief in supporter F is in $(0.5, 0.7]$, then the belief in B is in $(0.5, 0.7]$.

$$p(\texttt{F}) > 0.5 \wedge p(\texttt{F}) \leq 0.7 \rightarrow p(\texttt{B}) > 0.5 \wedge p(\texttt{B}) \leq 0.7$$

8. This constraint states that if the belief in supporter F is in $(0.7, 1]$, then the belief in B is in $(0.7, 1]$.

$$p(\texttt{F}) > 0.7 \rightarrow p(\texttt{B}) > 0.7$$

9. This constraint states that if the attacker K is believed, then J is not believed.

$$p(\texttt{K}) > 0.5 \rightarrow p(\texttt{J}) \leq 0.5$$

10. This constraint states that if the attacker I is believed, then E is not believed.

$$p(\texttt{I}) > 0.5 \rightarrow p(\texttt{E}) \leq 0.5$$

11. This constraint states that if the attacker H is believed, then D is not believed.

$$p(\texttt{H}) > 0.5 \rightarrow p(\texttt{D}) \leq 0.5$$

Now suppose that this set of constraints reflects the behaviour of a particular agent. So we can use these constraints together with the epistemic graph and probability distribution over the subsets of arguments to model the agent in a persuasion dialogue. We assume that we want to persuade the agent to have a strong belief in argument A. In the following table, let P_0 denote the initial belief that we think the agent has in the arguments.

	A	B	C	D	E	F	G	H	I	J	K
P_0	0.45	0.5	0.5	0.8	0.5	0.5	0.5	0.2	0.5	0.1	0.9
P_1	0.45	0.5	0.5	0.8	0.5	0.5	0.5	0.2	0.8	0.1	0.9
P_2	0.75	0.5	0.7	0.8	0.5	0.5	0.8	0.2	0.5	0.1	0.9

Given P_0, and the need to get a change in belief in A so that it is strongly believed, we can use constraint 2 as a guide. By getting a change in belief in B so that it is disbelieved, and by getting a change in belief in C so that it is strongly believed, we can get strong belief in A. This calls for positing argument I which through constraint 10 causes the required change in E (as illustrated by the intermediate probability function P_1), and then for positing argument C which through constraint 6 causes the required change in C. These changes will then cause the required change in A as specified by constraint 2 (as illustrated by the final probability function P_2.

Note, we do not consider here how the precise value is picked for the update in the belief in each argument, but rather direct the interested reader to [47, 51] for some options.

The above case study illustrates how the framework in this paper can be incorporated in a user model, and then used to guide the choice of moves in a persuasion dialogue.

4 Reasoning with Constraints

Previously, we have introduced the notions of epistemic constraints and graphs, and showed how they can be used to represent the requirements for believing or disbelieving a given argument. However, we have not yet explained how two epistemic formulae can be related based on their satisfying distributions, or what can be logically inferred from a given formula. We would like to address this here by first introducing the notion of epistemic entailment and, in the next subsections, providing a consequence relation for the basic and valued constraints. We will primarily consider working with the restricted case, i.e. one where the sets of values that the probability function can take on and that can appear as numerical values in the constraints are fixed and finite. This restricted case may be useful for handling Likert scales, which commonly appear in opinion surveys. For example, when asked about a statement, a respondent may be asked whether they *strongly agree*, *agree*, *neither agree nor disagree*, *disagree*, or *strongly agree* with it. This scale may, for example, be represented by the values 1, 0.75, 0.5, 0.25 and 0, respectively. We will show that in the restricted case, the valued constraints can be replaced by the basic ones. Nevertheless, before we do so, we need to introduce some basic notions first. From now on, unless stated otherwise, we will assume that the argumentation framework we are dealing with is finite and nonempty (i.e. the set of arguments in the graph is finite and nonempty).

Let us start with the unrestricted epistemic entailment relation. Given the semantics for constraints explained in Section 3.1, epistemic entailment can be defined in the following manner:

Definition 4.1. Let $\{\phi_1, \dots, \phi_n\} \subseteq \mathsf{VFormulae}(\mathcal{G})$ be a set of epistemic formulae, and $\psi \in \mathsf{VFormulae}(\mathcal{G})$ be a epistemic formula. The **epistemic entailment relation**, denoted $\Vdash$, is defined as follows.

$$\{\phi_1, \dots, \phi_n\} \Vdash \psi \text{ iff } \mathsf{Sat}(\{\phi_1, \dots, \phi_n\}) \subseteq \mathsf{Sat}(\psi)$$

Example 28. The following are some instances of epistemic entailment.

- $\{p(\mathtt{A}) < 0.2\} \Vdash p(\mathtt{A}) < 0.3$
- $\{p(\mathtt{A}) < 0.2\} \Vdash p(\mathtt{A} \wedge \mathtt{B}) < 0.2$
- $\{p(\mathtt{A}) < 0.9, p(\mathtt{A}) > 0.7\} \Vdash p(\mathtt{A}) \geq 0.7 \wedge \neg(p(\mathtt{A}) > 0.9)$

Let us now focus on the restricted entailment. We start by defining the restricted value set, which has to be closed under addition and subtraction (assuming the resulting value is still in the $[0,1]$ interval). We can then create subsets of this set according to a given inequality and "threshold" value, as well as sequences of values that can be seen as satisfying a given arithmetical formula:

Definition 4.2. Let Π be a finite set of rational numbers form the unit interval. Π is a **restricted value set** iff for any two $x, y \in \Pi$ it holds that if $x + y \leq 1$, then $x + y \in \Pi$, and if $x - y \geq 0$, then $x - y \in \Pi$. Given a nonempty restricted value set Π, with $\Pi^x_\# = \{y \in \Pi \mid x \# y\}$ we denote the subset of Π obtained according to the value x and relationship $\# \in \{=, \neq, \geq, \leq, >, <\}$. Given a nonempty restricted value set Π, a sequence of arithmetic operations $(*_1, \ldots, *_k)$ where $*_i \in \{+, -\}$ and $k \geq 0$, the **combination set** of sequences of values of Π is defined as:

$$\Pi_{\#}^{x,(*_1,\ldots,*_k)} = \begin{cases} \{(v) \mid v \in \Pi_{\#}^{x}\} & k = 0 \\ \{(v_1, \ldots, v_{k+1}) \mid v_i \in \Pi,\ v_1 *_1 \ldots *_k v_{k+1} \# x\} & \text{otherwise} \end{cases}$$

Example 29. Let $\Pi_1 = \{0, 0.5, 0.75, 1\}$. We can observe that it is not a restricted value set, since $0.75 - 0.5 = 0.25$ is missing from Π_1. Its modification, $\Pi_2 = \{0, 0.25, 0.5, 0.75, 1\}$, is a restricted value set. Similarly, it is easy to show that $\Pi_3 = \{0, \frac{1}{3}, \frac{2}{3}, \frac{3}{3}\}$ and $\Pi_4 = \{0, \frac{2}{5}, \frac{4}{5}\}$ are also restricted value sets.

Let us now focus on subsets of Π_2 according to a given inequality and assume that $x = 0.25$. We obtain the following subsets: ${\Pi_2}_{>}^{x} = \{0.5, 0.75, 1\}$, ${\Pi_2}_{<}^{x} = \{0\}$, ${\Pi_2}_{\geq}^{x} = \{0.25, 0.5, 0.75, 1\}$, ${\Pi_2}_{\leq}^{x} = \{0, 0.25\}$, ${\Pi_2}_{\neq}^{x} = \{0, 0.5, 0.75, 1\}$, and ${\Pi_2}_{=}^{x} = \{0.25\}$.

Finally, let us consider combination sets and assume we get a sequence of operations $(+, -)$, operator $=$ and a value $x = 1$. We are therefore looking for triples of values (τ_1, τ_2, τ_3) s.t. $x + y - z = 1$. For example, we can observe that $0 + 1 - 0 = 1$, $0.25 + 0.75 - 0 = 1$, $1 + 1 - 1 = 1$ and so on. By collecting such combinations of values from Π_2, we obtain fifteen possible value sequences, i.e. ${\Pi_2}_{=}^{1,(+,-)} = \{(0, 1, 0)$, $(0.25, 0.75, 0)$, $(0.25, 1, 0.25)$, $(0.5, 0.5, 0)$, $(0.5, 0.75, 0.25)$, $(0.5, 1, 0.5)$, $(0.75, 0.25, 0)$, $(0.75, 0.5, 0.25)$, $(0.75, 0.75, 0.5)$, $(0.75, 1, 0.75)$, $(1, 0, 0)$, $(1, 0.25, 0.25)$, $(1, 0.5, 0.5)$, $(1, 0.75, 0.75)$, $(1, 1, 1)\}$.

On the basis of a given restricted value set, we can now constrain our basic and valued approaches both in a syntactic and in a semantic way:

Definition 4.3. Let Π be a restricted value set. A basic formula $\psi \in \mathsf{BFormulae}(\mathcal{G})$ is **restricted** w.r.t. Π iff $\mathsf{Num}(\psi) \subseteq \Pi$. Let $\mathsf{BFormulae}(\mathcal{G}, \Pi)$ denote this set of restricted basic formulae.

A valued formula $\psi \in \mathsf{VFormulae}(\mathcal{G})$ is **restricted** w.r.t. Π iff $\mathsf{Num}(\psi) \subseteq \Pi$. Let $\mathsf{VFormulae}(\mathcal{G}, \Pi)$ denote this set of restricted valued formulae.

Definition 4.4. Let Π be a restricted value set. A probability distribution $P \in \mathsf{Dist}(\mathcal{G})$ is **restricted** w.r.t. Π iff for every $X \subseteq \mathsf{Nodes}(\mathcal{G})$, $P(X) \in \Pi$ and for every argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, $P(\mathtt{A}) \in \Pi$. Let $\mathsf{Dist}(\mathcal{G}, \Pi)$ denote the set of restricted distributions of $\mathcal{G}$.

Definition 4.5. Let Π be a restricted value set. For $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$, the **restricted satisfying distribution** w.r.t. Π, denoted $\mathsf{Sat}(\psi, \Pi)$, is

$$\mathsf{Sat}(\psi, \Pi) = \mathsf{Sat}(\psi) \cap \mathsf{Dist}(\mathcal{G}, \Pi)$$

Due to the properties of $\cap$, $\cup$ and $\setminus$, we can observe that restricted satisfying distributions can be manipulated similarly to the unrestricted ones, i.e. the following hold for formulae ψ and ϕ:

- $\mathsf{Sat}(\phi \wedge \psi, \Pi) = \mathsf{Sat}(\phi, \Pi) \cap \mathsf{Sat}(\psi, \Pi)$;
- $\mathsf{Sat}(\phi \vee \psi, \Pi) = \mathsf{Sat}(\phi, \Pi) \cup \mathsf{Sat}(\psi, \Pi)$; and
- $\mathsf{Sat}(\neg\phi, \Pi) = \mathsf{Sat}(\top, \Pi) \setminus \mathsf{Sat}(\phi, \Pi)$.

Example 30. Let $\Pi = \{0, 0.5, 1\}$. So the restricted basic language w.r.t. Π contains the restricted atoms $p(\alpha)\#0$, $p(\alpha)\#0.5$, and $p(\alpha)\#1$, where $\# \in \{=, \neq, \geq, \leq, >, <\}$ and the basic formulae that can be composed from them using the Boolean connectives. Let us assume we have a formula $p(\mathtt{A}) \geq 0.5$ on a graph s.t. $\{\mathtt{A}\} = \mathsf{Nodes}(\mathcal{G})$. We can create only two restricted satisfying distributions, namely P_1 s.t. $P_1(0) = 0$ and $P_1(1) = 1$ and P_2 s.t. $P_2(0) = 0.5$ and $P_2(1) = 0.5$.

In the valued language restricted w.r.t. Π, we can only have atoms of the form $\beta\#0$, $\beta\#0.5$, and $\beta\#1$, where $\beta \in \mathsf{OFormulae}(\mathcal{G})$. From these formulae we can built the valued atoms, and from these atoms we compose valued epistemic formulae using the Boolean connectives. Let us assume we have a formula $p(\mathtt{A}) + p(\mathtt{B}) \leq 0.5$ on a graph s.t. $\{\mathtt{A}, \mathtt{B}\} = \mathsf{Nodes}(\mathcal{G})$.

We can create three restricted satisfying distributions, namely P_1 s.t. $P_1(00) = 1$, $P_1(10) = 0$, $P_1(01) = 0$ and $P_1(11) = 0$, P_2 s.t. $P_2(00) = 0.5$, $P_2(10) = 0.5$, $P_2(01) = 0$ and $P_2(11) = 0$, and P_3 s.t. $P_3(00) = 0.5$, $P_3(10) = 0$, $P_3(01) = 0.5$ and $P_3(11) = 0$.

We can observe that depending on the graph and the restricted value set, it might not be possible to create a restricted distribution. For example, we can consider the set $\{0, 0.9\}$. Although it meets the restricted value set requirements, there is no way to add or subtract 0 and 0.9 such that they add up to 1. This means that it is not possible to define a distribution with this set. Thus, it makes sense to consider also a stronger version of Π that prevents such scenarios:

Definition 4.6. Let Π be a restricted value set. Π is **reasonable** iff for every graph $\mathcal{G}$ s.t. $\mathsf{Nodes}(\mathcal{G}) \neq \emptyset$, $\mathsf{Dist}(\mathcal{G}, \Pi) \neq \emptyset$.

The following simple properties allow us to easily detect reasonable restricted sets:

Lemma 4.7. *The following hold:*

- *If Π is a nonempty restricted value set, then $0 \in \Pi$.*
- *If Π is a reasonable restricted value set, then $0 \in \Pi$.*
- *A restricted value set Π is reasonable iff $1 \in \Pi$.*

It can happen that the combination sets or value subsets of Π are empty. However, as we can see, this occurs only if particular conditions are met:

Proposition 4.8. *Let Π be a nonempty restricted value set, $x \in \Pi$ a value, $\# \in \{=, \neq, \geq, \leq, >, <\}$ an inequality, and $(*_1, \ldots, *_k)$ a sequence of operators where $*_i \in \{+, -\}$ and $k \geq 0$. Let $max(\Pi)$ denote the maximal value of Π. The following hold:*

- $\Pi^x_\# = \emptyset$ *if and only if:*
  1. $\Pi = \{0\}$ *and* $\# = \neq$, *or*
  2. $\#$ *is* $>$ *and* $x = max(\Pi)$, *or*
  3. $\#$ *is* $<$ *and* $x = 0$.
- $\Pi^{x,(*_1,\ldots,*_k)}_\# = \emptyset$ *if and only if:*
  1. $k = 0$ *and* $\Pi^x_\# = \emptyset$, *or*
  2. $k > 0$, $\#$ *is* $>$, $x = max(\Pi)$ *and for no* $*_i$, $*_i = +$, *or*
  3. $k > 0$, $\#$ *is* $>$ *and* $\Pi = \{0\}$, *or*
  4. $k > 0$, $\#$ *is* $<$, $x = 0$ *and for no* $*_i$, $*_i = -$, *or*
  5. $k > 0$, $\#$ *is* $<$ *and* $\Pi = \{0\}$.
  6. $k > 0$, $\#$ *is* $\neq$, $\Pi = \{0\}$.

Let us now focus on reasoning in the restricted scenario. With the definition of restricted satisfying distributions, we can now define the restricted epistemic entailment relation in a way similar to the standard epistemic entailment:

Definition 4.9. Let Π be a restricted value set, $\{\phi_1, \ldots, \phi_n\} \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$ a set of epistemic formulae, and $\psi \in \mathsf{VFormulae}(\mathcal{G})$ an epistemic formula. The **restricted epistemic entailment relation** w.r.t. Π, denoted $\Vdash_\Pi$, is defined as follows.

$$\{\phi_1, \ldots, \phi_n\} \Vdash_\Pi \psi \text{ iff } \mathsf{Sat}(\{\phi_1, \ldots, \phi_n\}, \Pi) \subseteq \mathsf{Sat}(\psi, \Pi)$$

Example 31. Consider $\Pi = \{0, 0.25, 0.5, 0.75, 1\}$ and restricted valued formulae $p(\mathtt{A}) + p(\neg\mathtt{B}) \leq 1$ and $p(\mathtt{A}) + p(\neg\mathtt{B}) \leq 0.75$. It holds that

$$\{p(\mathtt{A}) + p(\neg\mathtt{B}) \leq 0.75\} \Vdash p(\mathtt{A}) + p(\neg\mathtt{B}) \leq 1$$

Let us now discuss how the restricted satisfying distributions and the restricted entailment are related to the unrestricted versions we have introduced previously. First of all, by Definition 4.5, we can observe that every restricted satisfying distribution for an epistemic formula is also a satisfying distribution. Thus, we can easily show that epistemic entailment implies restricted entailment:

Proposition 4.10. *Let* Π *be a restricted value set,* $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$ *a set of epistemic formulae, and* $\psi \in \mathsf{VFormulae}(\mathcal{G})$ *an epistemic formula. If* $\Phi \Vdash \psi$ *then* $\Phi \Vdash_{\Pi} \psi$.

In principle, we can observe that if "less" restricted entailment implies a "more" restricted one:

Proposition 4.11. *Let* $\Pi_1 \subseteq \Pi_2$ *be restricted value sets,* $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi_1)$ *a set of epistemic formulae, and* $\psi \in \mathsf{VFormulae}(\mathcal{G})$ *an epistemic formula. If* $\Phi \Vdash_{\Pi_2} \psi$ *then* $\Phi \Vdash_{\Pi_1} \psi$.

Note, it does not necessarily hold that if one formula follows from another in a restricted manner, then it also follows in the unrestricted one as illustrated below:

Example 32. Consider two formulae $\varphi_1 : p(\mathtt{A}) \neq 0.5$ and $\varphi_2 : p(\mathtt{A}) = 0 \vee p(\mathtt{A}) = 1$ and a reasonable restricted set $\Pi = \{0, 0.5, 1\}$. We can observe that $\mathsf{Sat}(\varphi_1, \Pi) = \mathsf{Sat}(\varphi_2, \Pi)$ and therefore $\{\varphi_1\} \Vdash_{\Pi} \varphi_2$. However, in the unrestricted case we can consider a probability distribution P s.t. $P(\mathtt{A}) = 0.9$ in order to show that $\mathsf{Sat}(\varphi_1) \nsubseteq \mathsf{Sat}(\varphi_2)$. We can observe that this issue would have been bypassed if, instead of Π, we considered the set $\Pi_2 = \{0, 0.25, 0.5, 0.75, 1\}$, for which $\{\varphi_1\} \nVdash_{\Pi_2} \varphi_2$. Consequently, although restricted entailment does not in general imply unrestricted entailment, for a given set of formulae it is possible to find such a Π for which this property holds.

The reason that an inference from the restricted entailment relation is not necessarily an inference from the unrestricted entailment relation is that the restricted case contains more information. The set Π is extra information that restricts the possible assignments for the probability distribution. Indeed, it could equivalently be represented as a set of constraints that could be added to the left-hand side of the unrestricted entailment relation. This is analogous to the use of explicit constraints on the domain in order to formalise the closed world assumption in predicate logic [78].

4.1 Disjunctive, Conjunctive and Negation Normal Forms

In propositional logic, we often analyze formulae in various normal forms due to their useful properties. Traditional forms include the negation normal form NNF, conjunctive normal form CNF and disjunctive normal form DNF. Given that epistemic formulae extend propositional logic, they can also be transformed into various normal forms if we look at epistemic atoms as propositions. In principle, for every formula φ we can find at least one formula φ' that is in NNF, CNF or DNF and s.t. $\mathsf{Sat}(\varphi) = \mathsf{Sat}(\varphi')$. However, they can be usually obtained through additional post–processing and use of various logical equivalences. In this section we will recall approaches for constructing particular NNFs, CNFs and DNFs that we will use throughout the text.

Let us start with the negation normal form transformation, which brings a formula to a form in which only conjunction, disjunction and negation are allowed as connectives, and negation can appear only in front of atoms. Traditionally, this is done by rewriting other connectives and using De Morgan's and double negation laws to push the negations inward. We add two steps to this approach to simplify the formula, i.e. we remove unnecessary brackets and repetitions. Please note we do not allow the use of other logical equivalences, such as identity or absorption laws etc.

Definition 4.12. An epistemic formula can be brought into its **negation normal form** using the following rules, where $\Rightarrow$ is the rewriting operation:

1. Eliminate iff and implications

$$\alpha \leftrightarrow \beta \Rightarrow (\alpha \to \beta) \wedge (\beta \to \alpha)$$

$$\alpha \to \beta \Rightarrow \neg\alpha \vee \beta$$

2. Move negation inwards using the following rules until negations appear only in front of atoms

$$\neg\neg\alpha \Rightarrow \alpha$$

$$\neg(\alpha \vee \beta) \Rightarrow \neg\alpha \wedge \neg\beta$$
$$\neg(\alpha \wedge \beta) \Rightarrow \neg\alpha \vee \neg\beta$$

3. Flatten conjunctions and disjunctions

$$(\alpha \wedge (\beta \wedge \gamma)) \Rightarrow (\alpha \wedge \beta \wedge \gamma)$$

$$(\alpha \vee (\beta \vee \gamma)) \Rightarrow (\alpha \vee \beta \vee \gamma)$$
$$((\alpha \wedge \beta) \wedge \gamma) \Rightarrow (\alpha \wedge \beta \wedge \gamma)$$
$$((\alpha \vee \beta) \vee \gamma) \Rightarrow (\alpha \vee \beta \vee \gamma)$$

4. Remove double occurrences of formulae

$$\alpha \vee \beta_1 \vee \ldots \vee \beta_n \vee \alpha \Rightarrow \alpha \vee \beta_1 \vee \ldots \vee \beta_n$$

$$\alpha \wedge \beta_1 \wedge \ldots \wedge \beta_n \wedge \alpha \Rightarrow \alpha \wedge \beta_1 \wedge \ldots \wedge \beta_n$$

To the NNF of a given formula obtained using this approach we will refer as d-NNF to distinguish it from those obtained through other methods.

The conjunctive and disjunctive normal forms can be built on top of the negation normal form through the use of distributive laws. Below we present an approach for CNF, and by replacing the second step with rules for distributing $\wedge$ over $\vee$, we would obtain a DNF approach.

Definition 4.13. An epistemic formula can be brought into its **conjunctive normal form** using the following rules:

1. Bring the formula into its d-NNF form

2. Distribute $\vee$ over $\wedge$ using the following rules

$$\alpha \vee (\beta \wedge \gamma) \Rightarrow (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$$

$$(\beta \wedge \gamma) \vee \alpha \Rightarrow (\beta \vee \alpha) \wedge (\gamma \vee \alpha)$$

3. Flatten conjunctions and disjunctions

$$(\alpha \wedge (\beta \wedge \gamma)) \Rightarrow (\alpha \wedge \beta \wedge \gamma)$$

$$(\alpha \vee (\beta \vee \gamma)) \Rightarrow (\alpha \vee \beta \vee \gamma)$$
$$((\alpha \wedge \beta) \wedge \gamma) \Rightarrow (\alpha \wedge \beta \wedge \gamma)$$
$$((\alpha \vee \beta) \vee \gamma) \Rightarrow (\alpha \vee \beta \vee \gamma)$$

4. Remove double occurrences of formulae

$$\alpha \vee \beta_1 \vee \ldots \vee \beta_n \vee \alpha \Rightarrow \alpha \vee \beta_1 \vee \ldots \vee \beta_n$$

$$\alpha \wedge \beta_1 \wedge \ldots \wedge \beta_n \wedge \alpha \Rightarrow \alpha \wedge \beta_1 \wedge \ldots \wedge \beta_n$$

To the CNFs and DNFs of a given formula obtained using this approach we will refer as d-CNF and d-DNF to distinguish them from those obtained by other methods.

Example 33. Consider the formula

$$\varphi : \neg p(\mathtt{A}) > 0.5 \to (((p(\mathtt{A}) > 0.5 \vee p(\mathtt{B}) > 0.5) \to p(\mathtt{C}) < 0.5) \wedge p(\mathtt{D}) + p(\mathtt{C}) \leq 1)$$

Let us first bring to d-NNF. We start by rewriting the implications:

$$\neg\neg p(\mathtt{A}) > 0.5 \vee ((\neg(p(\mathtt{A}) > 0.5 \vee p(\mathtt{B}) > 0.5) \vee p(\mathtt{C}) < 0.5) \wedge p(\mathtt{D}) + p(\mathtt{C}) \leq 1)$$

We now push the negations inwards:

$$p(\mathtt{A}) > 0.5 \vee (((\neg p(\mathtt{A}) > 0.5 \wedge \neg p(\mathtt{B}) > 0.5) \vee p(\mathtt{C}) < 0.5) \wedge p(\mathtt{D}) + p(\mathtt{C}) \leq 1)$$

At this point nothing more can be done and we obtain our d-NNF. We can now transform it into d-CNF. Through repeated application of distribution rules and removal of brackets we obtain the formula:

$$\begin{aligned}&(p(\mathtt{A}) > 0.5 \vee \neg p(\mathtt{A}) > 0.5 \vee p(\mathtt{C}) < 0.5) \wedge \\ &\qquad (p(\mathtt{A}) > 0.5 \vee \neg p(\mathtt{B}) > 0.5 \vee p(\mathtt{C}) < 0.5) \wedge \\ &\qquad\qquad (p(\mathtt{A}) > 0.5 \vee p(\mathtt{D}) + p(\mathtt{C}) \leq 1)\end{aligned}$$

We observe that if further logical equivalences were employed, then the first clause could be removed from the formula. However, we keep the d-CNF they way it is. The d-DNF of our formula would be:

$$p(\mathtt{A}) > 0.5 \vee (\neg p(\mathtt{A}) > 0.5 \wedge \neg p(\mathtt{B}) > 0.5 \wedge p(\mathtt{D}) + p(\mathtt{C}) \leq 1) \vee (p(\mathtt{C}) < 0.5 \wedge p(\mathtt{D}) + p(\mathtt{C}) \leq 1)$$

4.2 Distribution Disjunctive Normal Form

In the previous section we have discussed the disjunctive, conjunctive and negation normal forms of epistemic formulae, in which epistemic atoms were viewed simply as propositions. However, further notions can be introduced once we take the meaning of the atoms into account. In this section we introduce a normal form for epistemic formulae from which it is easy to read if and how this formula can be satisfied. Let us start by observing that for every probability distribution, we can create a basic epistemic formula describing precisely that distribution. As we may remember, a probability distribution maps sets of arguments to probabilities. For every such set, we can create a term (i.e. a propositional formula over arguments) describing it, where arguments contained in the set appear as positive literals and those not in the set appear as negative literals. This brings us to the notion of argument complete terms:

Definition 4.14. Let $\langle \mathtt{A}_1, \ldots, \mathtt{A}_n \rangle$ be the order of arguments in $\mathcal{G}$ and $\varphi \in \mathsf{Terms}(\mathcal{G})$ a term. Then φ is **argument complete** iff it is of the form $\alpha_1 \wedge \ldots \wedge \alpha_n$, where $\alpha_i = \mathtt{A}_i$ or $\alpha_i = \neg\mathtt{A}_i$. With $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \ldots, c_j\}$ we denote the set of all complete formulae on $\mathcal{G}$.

Example 34. Let us consider a graph with arguments $\mathtt{A}$, $\mathtt{B}$ and $\mathtt{C}$ and ordering $\langle \mathtt{A}, \mathtt{B}, \mathtt{C} \rangle$. We can create the following argument complete terms: $\neg\mathtt{A} \wedge \neg\mathtt{B} \wedge \neg\mathtt{C}$, $\mathtt{A} \wedge \neg\mathtt{B} \wedge \neg\mathtt{C}$, $\neg\mathtt{A} \wedge \mathtt{B} \wedge \neg\mathtt{C}$, $\neg\mathtt{A} \wedge \neg\mathtt{B} \wedge \mathtt{C}$, $\mathtt{A} \wedge \mathtt{B} \wedge \neg\mathtt{C}$, $\neg\mathtt{A} \wedge \mathtt{B} \wedge \mathtt{C}$, $\mathtt{A} \wedge \neg\mathtt{B} \wedge \mathtt{C}$, and $\mathtt{A} \wedge \mathtt{B} \wedge \mathtt{C}$.

By building epistemic atoms from using such terms, we can create a formula that associated with a given probability distribution and such that this distribution is the only one satisfying it:

Definition 4.15. Let $P \in \mathsf{Dist}(\mathcal{G})$ be a probability distribution and $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \ldots, c_j\}$ the collection of all argument complete terms for $\mathcal{G}$. The **epistemic formula associated with** P is $\varphi^P = p(c_1) = x_1 \wedge p(c_2) = x_2 \wedge \ldots \wedge p(c_j) = x_j$, where $x_i = P(c_i)$.

Proposition 4.16. *Let $P \in \mathsf{Dist}(\mathcal{G})$ be a probability distribution and φ^P its associated epistemic formula. Then $\{P\} = \mathsf{Sat}(\varphi^P)$.*

Example 35. Let us consider the probability distribution P over the set of arguments $\{\mathtt{A}, \mathtt{B}, \mathtt{C}\}$ s.t. $P(\varnothing) = 0.1$, $P(\{\mathtt{A}\}) = 0.05$, $P(\{\mathtt{B}\}) = 0.2$, $P(\{\mathtt{C}\}) = 0.15$, $P(\{\mathtt{A}, \mathtt{B}\}) = 0.1$, $P(\{\mathtt{B}, \mathtt{C}\}) = 0.1$, $P\{\mathtt{A}, \mathtt{C}\}) = 0$, and $P(\{\mathtt{A}, \mathtt{B}, \mathtt{C}\}) = 0.3$. The epistemic formula associated with P is $p(\neg\mathtt{A} \wedge \neg\mathtt{B} \wedge \neg\mathtt{C}) = 0.1 \wedge p(\mathtt{A} \wedge \neg\mathtt{B} \wedge \neg\mathtt{C}) = 0.05 \wedge p(\neg\mathtt{A} \wedge \mathtt{B} \wedge \neg\mathtt{C}) = 0.2 \wedge p(\neg\mathtt{A} \wedge \neg\mathtt{B} \wedge \mathtt{C}) = 0.15 \wedge p(\mathtt{A} \wedge \mathtt{B} \wedge \neg\mathtt{C}) = 0.1 \wedge p(\neg\mathtt{A} \wedge \mathtt{B} \wedge \mathtt{C}) = 0.1 \wedge p(\mathtt{A} \wedge \neg\mathtt{B} \wedge \mathtt{C}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B} \wedge \mathtt{C}) = 0.3$.

Consequently, for every epistemic formula φ, we can create create a semantically equivalent formula φ' that is built from the formulae associated with the distributions satisfying φ. We refer to this new formula as the distribution disjunctive normal form. Given the fact that an epistemic formula can, in principle, be satisfied by infinitely many distributions, we only consider this form in the context of restricted reasoning.

Definition 4.17. Let Π be a reasonable restricted value set, $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$ be a restricted epistemic formula and $\{P_1, \ldots, P_n\} = \mathsf{Sat}(\psi, \Pi)$ the set of distributions satisfying ψ under Π. The **distribution disjunctive normal form** (abbreviated DDNF) of ψ is $\perp$ iff $\mathsf{Sat}(\psi, \Pi) = \emptyset$, and $\varphi^{P_1} \vee \varphi^{P_2} \ldots \vee \varphi^{P_n}$ otherwise, where φ^{P_i} is the epistemic formula associated with P_i.

Proposition 4.18. *Let Π be a reasonable restricted value set, $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$ be a restricted epistemic formula and φ its distribution disjunctive normal form. Then* $\mathsf{Sat}(\psi, \Pi) = \mathsf{Sat}(\varphi, \Pi)$.

Example 36. Let us consider a set of arguments $\{\mathtt{A}, \mathtt{B}\}$ and let $\Pi = \{0, 0.5, 1\}$. Assume we have a term $\mathtt{A} \vee \mathtt{B}$ and an epistemic atom $p(\mathtt{A} \vee \mathtt{B}) > 0.5$. The satisfying distributions and their associated epistemic formulae are tabulated below. Therefore, the DDNF associated with $p(\mathtt{A} \vee \mathtt{B}) > 0.5$ is $\varphi^{P_1} \vee \varphi^{P_2} \vee \varphi^{P_3} \vee \varphi^{P_4} \vee \varphi^{P_5} \vee \varphi^{P_6}$.

	$\emptyset$	$\{\mathtt{A}\}$	$\{\mathtt{B}\}$	$\{\mathtt{A}, \mathtt{B}\}$	φ^{P_i}
P_1	0	1	0	0	$p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 1 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0$
P_2	0	0	1	0	$p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 1 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0$
P_3	0	0	0	1	$p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 1$
P_4	0	0.5	0	0.5	$p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0.5 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0.5$
P_5	0	0	0.5	0.5	$p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0.5 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0.5$
P_6	0	0.5	0.5	0	$p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0.5 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0.5 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0$

4.3 Reasoning with Basic Constraints: Restricted Case

We now provide a proof theory for reasoning with basic constraints that extends propositional logic as follows. This allows us some ability to rewrite atoms of the constraint language, and to use the proof theory of classical propositional logic.

Definition 4.19. Let $\# \in \{=, \neq, \geq, \leq, >, <\}$ be the set of inequality relationships, let Π be a reasonable restricted value set, and let $\Pi^x_\# = \{y \in \Pi \mid x\#y\}$ be the subset obtained according to the value x and relationship $\#$. Let $\Pi^{x,(+,-)}_=$ be the set of triples of values from Π s.t. the sum of the first two values minus the last value equals to a value $x \in \Pi$. The **restricted epistemic consequence relation**, denoted $\Vdash_\Pi$, is defined as follows, where $\vdash$ is propositional consequence relation, $\alpha, \beta \in \mathsf{Terms}(\mathcal{G})$, $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$, $\phi, \psi, \phi_1, \ldots, \phi_n \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$ and $x \in \Pi$.

The following proof rules are the **basic rules**:

$(B1)$ $\Phi \Vdash_\Pi p(\alpha) \geq 0$ iff $\Phi \Vdash_\Pi \top$ $\quad$ $(B2)$ $\Phi \Vdash_\Pi p(\alpha) \leq 1$ iff $\Phi \Vdash_\Pi \top$

$(B3)$ $\Phi \Vdash_\Pi p(\top) = 1$ iff $\Phi \Vdash_\Pi \top$ $\quad$ $(B4)$ $\Phi \Vdash_\Pi p(\perp) = 0$ iff $\Phi \Vdash_\Pi \top$

The following proof rule is the **probabilistic rule**:

$$(PR1) \quad \Phi \Vdash_\Pi \bigvee_{(x,y,z,v) \in \Pi^{0,(-,-,+)}_=} (p(\alpha \vee \beta) = x \wedge p(\alpha) = y \wedge p(\beta) = z \wedge p(\alpha \wedge \beta) = v)$$

The following proof rules are the **subject rules**.

$(S1)$ $\Phi \Vdash_\Pi p(\alpha) > x$ and $\{\alpha\} \vdash \beta$ implies $\Phi \Vdash_\Pi p(\beta) > x$
$(S2)$ $\Phi \Vdash_\Pi p(\alpha) \geq x$ and $\{\alpha\} \vdash \beta$ implies $\Phi \Vdash_\Pi p(\beta) \geq x$
$(S3)$ $\Phi \Vdash_\Pi p(\alpha) \leq x$ and $\{\beta\} \vdash \alpha$ implies $\Phi \Vdash_\Pi p(\beta) \leq x$
$(S4)$ $\Phi \Vdash_\Pi p(\alpha) < x$ and $\{\beta\} \vdash \alpha$ implies $\Phi \Vdash_\Pi p(\beta) < x$

The following proof rules are the **enumeration rules**.

$(E1)$ $\Phi \Vdash_\Pi p(\alpha)\#x$ iff ($\Phi \Vdash_\Pi \bigvee_{v\in\Pi^x_\#} p(\alpha) = v$ if $\Pi^x_\# \neq \varnothing$ and $\Phi \Vdash_\Pi \bot$ otherwise)

$(E2)$ $\Phi \Vdash_\Pi p(\alpha) > x$ iff $\Phi \Vdash_\Pi \neg(\bigvee_{v\in\Pi^x_\leq} p(\alpha) = v)$

$(E3)$ $\Phi \Vdash_\Pi p(\alpha) \geq x$ iff ($\Phi \Vdash_\Pi \neg(\bigvee_{v\in\Pi^x_<} p(\alpha) = v)$ if $\Pi^x_< \neq \varnothing$ and $\Phi \Vdash_\Pi \neg(\bot)$ otherwise)

$(E4)$ $\Phi \Vdash_\Pi p(\alpha) < x$ iff $\Phi \Vdash_\Pi \neg(\bigvee_{v\in\Pi^x_\geq} p(\alpha) = v)$

$(E5)$ $\Phi \Vdash_\Pi p(\alpha) \leq x$ iff ($\Phi \Vdash_\Pi \neg(\bigvee_{v\in\Pi^x_>} p(\alpha) = v)$ if $\Pi^x_> \neq \varnothing$ and $\Phi \Vdash_\Pi \neg(\bot)$ otherwise)

The following proof rules are the **propositional rules**.

$(P1)$ $\Phi \Vdash_\Pi \phi_1$ and ... and $\Phi \Vdash_\Pi \phi_n$ and $n \geq 1$ and $\{\phi_1, \ldots, \phi_n\} \vdash \psi$ implies $\Phi \Vdash_\Pi \psi$
$(P2)$ if $\Phi \vdash \varphi$ then $\Phi \Vdash_\Pi \varphi$

The basic rules grasp the primitive properties of probabilities, i.e. that any probability is in the unit interval, and that probabilities of $\top$ and $\bot$ are respectively 1 and 0. The probabilistic rule states the relation between the probabilities of related formulae. The enumeration rules allow us to transform any inequality into a formula using only equality under the given restricted set Π. However, given the results of Proposition 4.8, in some cases it can happen that the appropriate subsets of Π are empty. Thus, wherever applicable, we make it clear that the resulting formula should be seen as falsity. The subject rule captures the fact that if a probabilistic inequality holds for a subject, it holds for weaker subjects. Finally, the propositional rules capture how the reasoning extends classical propositional logic.

Example 37. Let $\Pi = \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$. Consider $\Phi = \{p(\mathtt{A}) > 0.5, p(\mathtt{A}) < 0.7\}$. From $p(\mathtt{A}) > 0.5$, we get $\bigvee p(\mathtt{A}) \in \{0.6, 0.7, 0.8, 0.9, 1.0\}$, and from $p(\mathtt{A}) < 0.7$, we get $\neg(\bigvee p(\mathtt{A}) \in \{0.7, 0.8, 0.9, 1.0\})$. By the resolution proof rule for propositional logic applied to these two derived basic formulae, we get $\bigvee p(\mathtt{A}) \in \{0.5, 0.6, 0.7\}$.

From these rules we can derive a number of additional properties, such as:

Proposition 4.20. *Let $\# \in \{=, \neq, \geq, \leq, >, <\}$ be the set of inequality relationships, let Π be a reasonable restricted value set, and let $\Pi^x_\# = \{y \in \Pi \mid x\#y\}$ be the subset obtained according to the value x and relationship $\#$. The following hold, where $\vdash$ is propositional consequence relation, $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$, $\phi, \psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$ and $x \in \Pi$.*

1. $\Phi \Vdash_\Pi p(\alpha) \geq 0$
2. $\Phi \Vdash_\Pi p(\alpha) \leq 1$
3. $\Phi \Vdash_\Pi p(\top) = 1$
4. $\Phi \Vdash_\Pi p(\bot) = 0$
5. $\Phi \Vdash_\Pi p(\alpha) > x$ *implies* $\Phi \Vdash_\Pi p(\alpha) \geq x$
6. $\Phi \Vdash_\Pi p(\alpha) < x$ *implies* $\Phi \Vdash_\Pi p(\alpha) \leq x$
7. $\Phi \Vdash_\Pi p(\alpha) = x$ *implies* $\Phi \Vdash_\Pi p(\alpha) \geq x$
8. $\Phi \Vdash_\Pi p(\alpha) = x$ *implies* $\Phi \Vdash_\Pi p(\alpha) \leq x$
9. $\Phi \Vdash_\Pi p(\alpha) \neq x$ *iff* $\Phi \Vdash_\Pi p(\alpha) > x \vee p(\alpha) < x$
10. $\Phi \Vdash_\Pi p(\alpha) \geq x$ *iff* $\Phi \Vdash_\Pi p(\alpha) > x \vee p(\alpha) = x$

11. $\Phi \Vdash_\Pi p(\alpha) \leq x$ *iff* $\Phi \Vdash_\Pi p(\alpha) < x \vee p(\alpha) = x$

12. $\Phi \Vdash_\Pi p(\alpha) > x$ *iff* $\Phi \Vdash_\Pi p(\alpha) \geq x \wedge p(\alpha) \neq x$

13. $\Phi \Vdash_\Pi p(\alpha) < x$ *iff* $\Phi \Vdash_\Pi p(\alpha) \leq x \wedge p(\alpha) \neq x$

14. $\Phi \Vdash_\Pi p(\alpha) = x$ *iff* $\Phi \Vdash_\Pi p(\alpha) \geq x \wedge p(\alpha) \leq x$

15. $\Phi \Vdash_\Pi p(\alpha) > x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) \leq x)$

16. $\Phi \Vdash_\Pi p(\alpha) < x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) \geq x)$

17. $\Phi \Vdash_\Pi p(\alpha) \leq x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) > x)$

18. $\Phi \Vdash_\Pi p(\alpha) \geq x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) < x)$

19. $\Phi \Vdash_\Pi p(\alpha) = x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) \neq x)$

20. $\Phi \Vdash_\Pi p(\alpha) \neq x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) = x)$

21. $\Phi \Vdash_\Pi p(\alpha) = x$ *and* $\{\alpha\} \vdash \beta$ *implies* $\Phi \Vdash_\Pi p(\beta) \geq x$

22. $\Phi \Vdash_\Pi p(\alpha) = x$ *and* $\{\beta\} \vdash \alpha$ *implies* $\Phi \Vdash_\Pi p(\beta) \leq x$

23. *if* $\{\alpha\} \vdash \beta$ *and* $\{\beta\} \vdash \alpha$ *then* $\Phi \Vdash_\Pi p(\alpha) = x$ *iff* $\Phi \Vdash_\Pi p(\beta) = x$

24. $\Phi \Vdash_\Pi p(\alpha) > x$ *and* $x \geq y$ *implies* $\Phi \Vdash_\Pi p(\alpha) > y$

25. $\Phi \Vdash_\Pi p(\alpha) > x$ *and* $x > y$ *implies* $\Phi \Vdash_\Pi p(\alpha) > y$

26. $\Phi \Vdash_\Pi p(\alpha) \geq x$ *and* $x \geq y$ *implies* $\Phi \Vdash_\Pi p(\alpha) \geq y$

27. $\Phi \Vdash_\Pi p(\alpha) \geq x$ *and* $x > y$ *implies* $\Phi \Vdash_\Pi p(\alpha) > y$

28. $\Phi \Vdash_\Pi p(\alpha) < x$ *and* $x \leq y$ *implies* $\Phi \Vdash_\Pi p(\alpha) < y$

29. $\Phi \Vdash_\Pi p(\alpha) < x$ *and* $x < y$ *implies* $\Phi \Vdash_\Pi p(\alpha) < y$

30. $\Phi \Vdash_\Pi p(\alpha) \leq x$ *and* $x \leq y$ *implies* $\Phi \Vdash_\Pi p(\alpha) \leq y$

31. $\Phi \Vdash_\Pi p(\alpha) \leq x$ *and* $x < y$ *implies* $\Phi \Vdash_\Pi p(\alpha) < y$

32. $\Phi \Vdash_\Pi p(\alpha) = x \wedge p(\alpha) = y$ *where* $x \neq y$ *iff* $\Phi \Vdash_\Pi \perp$

33. $\Phi \Vdash_\Pi p(\alpha \vee \beta) = v$ *iff* $\Phi \Vdash_\Pi \bigvee_{(x,y,z) \in \Pi_{=}^{v,(+,-)}} (p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z)$

34. $\Phi \Vdash_\Pi p(\alpha \wedge \beta) = v$ *iff* $\Phi \Vdash_\Pi \bigvee_{(x,y,z) \in \Pi_{=}^{v,(+,-)}} (p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \vee \beta) = z)$

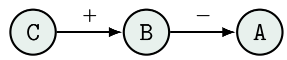
35. $\Phi \Vdash_\Pi p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z$ *and* $(0 > x + y - z$ *or* $x + y - z > 1)$ *implies* $\Phi \Vdash_\Pi \perp$

36. $\Phi \Vdash_\Pi p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \vee \beta) = z$ *and* $(0 > x + y - z$ *or* $x + y - z > 1)$ *implies* $\Phi \Vdash_\Pi \perp$

Example 38. Let $\Pi = \{0, 0.25, 0.5, 0.75, 1.0\}$. Consider $\Phi = \{p(\texttt{A}) > 0.5\}$. From $p(\texttt{A}) > 0.5$, we get $p(\texttt{A}) = 0.75 \vee p(\texttt{A}) = 1$. We can observe that $\{p(\texttt{A}) = 0.75 \vee p(\texttt{A}) = 1\} \vdash p(\texttt{A}) = 0.5 \vee p(\texttt{A}) = 0.75 \vee p(\texttt{A}) = 1$. Hence, through the use of propositional rule, $\Phi \Vdash_\Pi p(\texttt{A}) = 0.5 \vee p(\texttt{A}) = 0.75 \vee p(\texttt{A}) = 1$, which by the enumeration rule is the same as $\Phi \Vdash_\Pi p(\texttt{A}) \geq 0.5$. This could have been equivalently derived with derivable rule 5.

We can use the epistemic consequence relation to infer relationships between unconnected nodes as illustrated next.

Example 39. For the following graph, consider the constraints $\mathcal{C} = \{p(\texttt{C}) > 0.5 \rightarrow p(\texttt{B}) > 0.5, p(\texttt{B}) > 0.5 \rightarrow p(\texttt{A}) \leq 0.5\}$. From $\mathcal{C}$, we can infer $p(\texttt{C}) > 0.5 \rightarrow p(\texttt{A}) \leq 0.5$.



We now focus on the properties of the restricted epistemic consequence relation. First of all, it is easy to show that every rule in the system, and hence the system itself, is sound:

Proposition 4.21. *Let Π be a restricted value set. For $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$, and $\psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$, if $\Phi \Vdash_{\Pi} \psi$ then $\Phi \Vvdash_{\Pi} \psi$.*

However, as it is often the case, the completeness is somewhat more difficult to show. We may recall that for every probability distribution, we can create a basic epistemic formula describing precisely that distribution. From the disjunction of such formulae, we have created the distribution disjunctive normal form (DDNF) of every formula, the models of which were identical with the original formula. The challenge of the completeness proof is therefore to show that the DDNF of a given formula is equivalent to it not only semantically, but also syntactically. This is done by first transforming every term into a disjunction of argument complete terms, then separating this epistemic atom into further atoms s.t. every one of them contains precisely one complete term through the use of probabilistic rules. In order to do so, we first need an intermediate result stated in the first point of the proposition below. Since the obtained collection of atoms does not necessarily use all of the possible argument complete terms, the formulae need to be extended in a reasonable way. In order to do so, in the second point of the proposition below we show that from any set of formulae Φ we can obtain an epistemic formula describing all possible probabilistic distributions for a given graph. By combining these results we are therefore brought to the final result, showing the syntactical equivalence of the epistemic formula and its DDNF:

Proposition 4.22. *Let $\# \in \{=, \neq, \geq, \leq, >, <\}$ be the set of inequality relationships, let Π be a reasonable restricted value set, and let $\Pi^x_{\#} = \{y \in \Pi \mid x \# y\}$ be the subset obtained according to the value x and relationship $\#$. Let $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \ldots, c_j\}$ be the collection of all argument complete propositional formulae for $\mathcal{G}$ and $T^{\Pi}_{v,k} = \Pi^{v,(+,\ldots,+)}_{=}$ s.t. the length of $(+, \ldots, +)$ is $k-1$ be the collection of k–tuples of values from Π that sum up to $v \in \Pi$. The following hold, where $\vdash$ is propositional consequence relation, $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$, $\phi, \psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$ and $x \in \Pi$. The following hold:*

1. *for consistent formulae $\alpha_1, \ldots \alpha_m \in \mathsf{Terms}(\mathcal{G})$, if for all $1 \leq i, j \leq m$ s.t. $i \neq j$ it holds that $\alpha_i \wedge \alpha_j \vdash \bot$, then $\Phi \Vdash_{\Pi} p(\alpha_1 \vee \ldots \vee \alpha_m) = x$ iff $\Phi \Vdash_{\Pi} \bigvee_{(\tau_1, \ldots, \tau_m) \in T^{\Pi}_{x,m}} (p(\alpha_1) = \tau_1 \wedge \ldots \wedge p(\alpha_m) = \tau_m)$*
2. *$\Phi \Vdash_{\Pi} \bigvee_{(\tau_1, \ldots, \tau_j) \in T^{\Pi}_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$*
3. *$\Phi \Vdash_{\Pi} \psi$ iff $\Phi \Vdash_{\Pi} \varphi$, where φ is the distribution disjunctive normal form of ψ*

Example 40. Let us come back to the Example 36, i.e. we consider the set of arguments $\{\mathtt{A}, \mathtt{B}\}$, reasonable restricted value set $\Pi = \{0, 0.5, 1\}$ and an epistemic atom $p(\mathtt{A} \vee \mathtt{B}) > 0.5$. Based on the enumeration rule, $p(\mathtt{A} \vee \mathtt{B}) > 0.5$ is equivalent to $p(\mathtt{A} \vee \mathtt{B}) = 1$. The (full) disjunctive normal form of $\mathtt{A} \vee \mathtt{B}$ is $(\neg\mathtt{A} \wedge \mathtt{B}) \vee (\mathtt{A} \wedge \neg\mathtt{B}) \vee (\mathtt{A} \wedge \mathtt{B})$. We can therefore use the subject and derivable rules to show that $p(\mathtt{A} \vee \mathtt{B}) = 1$ is equivalent to $p((\neg\mathtt{A} \wedge \mathtt{B}) \vee (\mathtt{A} \wedge \neg\mathtt{B}) \vee (\mathtt{A} \wedge \mathtt{B})) = 1$. This, by the first point of Proposition 4.22, is equivalent to $(p(\mathtt{A} \wedge \neg\mathtt{B}) = 1 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0) \vee (p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 1 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0) \vee (p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 1) \vee (p(\mathtt{A} \wedge \neg\mathtt{B}) = 0.5 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0.5) \vee (p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0.5 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0.5) \vee (p(\mathtt{A} \wedge \neg\mathtt{B}) = 0.5 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0.5 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0)$. We then use the second point of the aforementioned proposition in order to insert the epistemic atoms associated with the term $\neg\mathtt{A} \wedge \neg\mathtt{B}$ into our formula. This brings us to the epistemic formula $(p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 1 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0) \vee (p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 1 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0) \vee (p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 1) \vee (p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0.5 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0.5) \vee (p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0.5 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0.5) \vee (p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0.5 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0.5 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0)$. It is easy to see that our result coincides with the formula from Example 36.

With the use of DDNFs and their properties, we can therefore show that any formula entailed by a given set of formulae also follows from this set through the use of the consequence relation. This, paired with Proposition 4.21, brings us to the following correctness result showing that the epistemic consequence relation is sound and complete with respect to the epistemic entailment relation.

Proposition 4.23. *Let Π be a restricted value set. For $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$, and $\psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$, $\Phi \Vdash_{\Pi} \psi$ iff $\Phi \Vdash_{\Pi} \psi$.*

In addition, the following property can be shown:

Proposition 4.24. *Let Π be a reasonable restricted value set. For $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$ and $\psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$, $\Phi \Vdash_{\Pi} \psi$ iff $\Phi \cup \{\neg\psi\} \Vdash_{\Pi} \bot$.*

For a finite set of rational numbers from the unit interval Π, representing and reasoning with the restricted constraint language w.r.t. Π is equivalent to propositional logic. We show this via the next two lemmas.

Lemma 4.25. *Let Π be a reasonable restricted value set, and let the restricted constraint language w.r.t. Π be $\mathsf{BFormulae}(\mathcal{G}, \Pi)$. There is a set of propositional formulae Ω with $\Lambda \subseteq \Omega$, and there is a function $f : \mathsf{BFormulae}(\mathcal{G}, \Pi) \to \Omega$ s.t. for each $\{\phi_1, \ldots, \phi_n\} \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$, and for each $\psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$,*

$$\{\phi_1, \ldots, \phi_n\} \Vdash_{\Pi} \psi \textit{ iff } \{f(\phi_1), \ldots, f(\phi_n)\} \cup \Lambda \vdash f(\psi)$$

Lemma 4.26. *Let Ω be a propositional language composed from a set of atoms and the usual definitions for the Boolean connectives. There is a restricted constraint language $\mathsf{BFormulae}(\mathcal{G}, \Pi)$ where $\Pi = \{0, 1\}$ and there is a function $g : \Omega \to \mathsf{BFormulae}(\mathcal{G}, \Pi)$ s.t. for each set of propositional formulae $\{\alpha_1, \ldots, \alpha_n\} \subseteq \Omega$ and for each propositional formula $\beta \in \Omega$,*

$$\{\alpha_1, \ldots, \alpha_n\} \vdash \beta \textit{ iff } \{g(\alpha_1), \ldots, g(\alpha_n)\} \Vdash_{\Pi} g(\beta)$$

From Lemmas 4.25 and 4.26, we obtain the following proposition showing the equivalence. This means that whatever can be represented or inferred in the restricted constraint language can be represented or inferred in the classical propositional language and vice versa.

Proposition 4.27. *The restricted constraint language with the epistemic consequence relation is equivalent to the classical propositional language with the classical propositional consequence relation.*

The restricted language (where the values for the inequalities are restricted to a finite set of values from the unit interval) allows for inequalities to be rewritten as a disjunction of equalities. This then allows for an epistemic consequence relation to be defined as a conservative extension of the classical propositional consequence relation. The advantage of this restricted version is that the reasoning is decidable, and it would be straightforward to implement automated reasoning machinery based, for instance, on SAT solvers. For some applications, such as user modelling in persuasion dialogues, having a restricted set of values (such as corresponding to a Likert scale) would offer a sufficiently rich framework.

4.4 Reasoning with Valued Constraints: Restricted Case

We now generalize the consequence relation from the previous subsection so that we can reason with valued constraints in the restricted language. We start with some subsidiary definitions associated with the arithmetic nature of operational formulae. Although they are not limited to restricted formulae only, we prefer to have them at hand due to the fact that we will be using them in our valued proof system:

Definition 4.28. Let $f_1 : p(\alpha_1) *_1 p(\alpha_2) *_2 \ldots *_{m-1} p(\alpha_m)$, and $f_2 : p(\beta_1) \star_1 p(\beta_2) \star_2 \ldots \star_{l-1} p(\beta_l)$, where $\alpha_i, \beta_i \in \mathsf{Terms}(\mathcal{G})$ and $*_i, \star_i \in \{+, -\}$, be operational formulae.

- $f_1 \simeq_{re} f_2$ denotes the **rearrangement equivalence relation** that holds when f_1 is a valid arithmetical rearrangement of f_2 obtained by appropriate commutative and associative laws.
- $f_1 \succeq_{su} f_2$ denotes the **subject inequality relation** that holds when f_2 is obtained from f_1 by logical weakening of an element $p(\alpha_i)$ of f_1 to $p(\alpha'_i)$ where $\{\alpha_i\} \vdash \alpha'_i$, and all other elements are the same in f_1 and f_2.

- with $f_1 \succeq_{su}^{+} f_2$ we denote the case where $f_1 \succeq_{su} f_2$ and either $i = 1$ or $*_{i-1} = +$
- with $f_1 \succeq_{su}^{-} f_2$ we denote the case where $f_1 \succeq_{su} f_2$, $i > 1$ and $*_{i-1} = -$

Let $\varphi_1 = f_1 \# x$ and $\varphi_2 = f_2 \# x$, where $\# \in \{=, \neq, \geq, \leq, >, <\}$ and $x \in [0,1]$, be valued epistemic atoms.

- we say that $\varphi_1 \simeq_{re} \varphi_2$ iff $f_1 \simeq_{re} f_2$
- we say that $\varphi_1 \succeq_{su} \varphi_2$ iff $f_1 \succeq_{su} f_2$
  - with $\varphi_1 \succeq_{su}^{+} \varphi_2$ we denote the case where $f_1 \succeq_{su}^{+} f_2$
  - with $\varphi_1 \succeq_{su}^{-} \varphi_2$ we denote the case where $f_1 \succeq_{su}^{-} f_2$

Example 41. The following illustrate the rearrangement equivalence relation.

$$p(\mathtt{A}) + p(\mathtt{B}) - p(\mathtt{D}) > 0 \simeq_{re} p(\mathtt{B}) - p(\mathtt{D}) + p(\mathtt{A}) > 0$$
$$p(\mathtt{A}) - p(\mathtt{C}) + p(\mathtt{B}) > 0.5 \simeq_{re} p(\mathtt{A}) + p(\mathtt{B}) - p(\mathtt{C}) > 0.5$$

Example 42. The following illustrate the subject inequality relation.

$$p(\mathtt{B}) - p(\mathtt{A} \wedge \mathtt{C}) > x \succeq_{su} p(\mathtt{B} \vee \mathtt{D}) - p(\mathtt{A} \wedge \mathtt{C}) > x$$
$$p(\mathtt{B}) - p(\mathtt{A} \wedge \mathtt{C} \wedge \mathtt{E}) < x \succeq_{su} p(\mathtt{B}) - p(\mathtt{A} \wedge \mathtt{C}) < x$$

Proposition 4.29. *For valued epistemic atoms $\varphi_1 = f_1 \# x$ and $\varphi_2 = f_2 \# x$ in $\mathsf{VFormulae}(\mathcal{G}, \Pi)$, the following hold:*

- *if $\varphi_1 \simeq_{re} \varphi_2$ then $\mathsf{Sat}(\varphi_1) = \mathsf{Sat}(\varphi_2)$*
- *if $\varphi_1 \succeq_{su}^{+} \varphi_2$ and $\# \in \{>, \geq\}$ then $\mathsf{Sat}(\varphi_1) \subseteq \mathsf{Sat}(\varphi_2)$, and if $\# \in \{<, \leq\}$, then $\mathsf{Sat}(\varphi_2) \subseteq \mathsf{Sat}(\varphi_1)$*
- *if $\varphi_1 \succeq_{su}^{-} \varphi_2$ and $\# \in \{<, \leq\}$ then $\mathsf{Sat}(\varphi_1) \subseteq \mathsf{Sat}(\varphi_2)$, and if $\# \in \{>, \geq\}$, then $\mathsf{Sat}(\varphi_2) \subseteq \mathsf{Sat}(\varphi_1)$*

We can now introduce the proof rule system for the valued epistemic formulae. We can observe that in many ways, these rules are natural generalizations of the basic system. We start by introducing the basic rules, which state some simple properties of probabilities, such as belonging to the unit interval. This is followed by the probabilistic rule that will later allow us to break epistemic atoms apart. The subject rules capture the behaviour of epistemic formulae that are connected through the subject inequality relation. The enumeration rules allow us to list all the possible values the epistemic atoms involved in a given formula may take in order to satisfy the inequality expressed in the rule. Similarly as in the case of the basic systems, in some cases it can happen that the appropriate combination sets are empty (see also Proposition 4.8). Thus, wherever applicable, we make it clear that the resulting formula should be seen as falsity. Finally, the propositional rule captures how the reasoning extends classical propositional logic.

Definition 4.30. Let $\# \in \{=, \neq, \geq, \leq, >, <\}$ be the set of inequality relationships, let $x \in [0,1]$, let Π be a restricted value set, and let $\Pi_{\#}^{x,(*_1,\ldots,*_{m-1})}$ be the combination set of Π obtained according to the value x, relationship $\#$ and the sequence arithmetic operations of arithmetic operations $(*_1, \ldots, *_{m-1})$. Also let $f_1 : p(\alpha_1) *_1 p(\alpha_2) *_2 \ldots *_{k-1} p(\alpha_k)$ and $f_2 : p(\beta_1) \star_1 p(\beta_2) \star_2 \ldots \star_{l-1} p(\beta_l)$, where $k, l \geq 1$, $\alpha_i, \beta_i \in \mathsf{Terms}(\mathcal{G})$ and $\star_j, *_i \in \{+, -\}$ be operational formulae. The **restricted++ epistemic consequence relation**, denoted $\Vdash_{\Pi}^{+}$, is defined as follows, where $\vdash$ is propositional consequence relation, $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$, and $\phi, \psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$.

The following proof rules are the **basic rules**:

$$(B1)\ \Phi \Vdash_{\Pi}^{+} p(\alpha) \geq 0 \text{ iff } \Phi \Vdash_{\Pi}^{+} \top \qquad (B2)\ \Phi \Vdash_{\Pi}^{+} p(\alpha) \leq 1 \text{ iff } \Phi \Vdash_{\Pi}^{+} \top$$
$$(B3)\ \Phi \Vdash_{\Pi}^{+} p(\top) = 1 \text{ iff } \Phi \Vdash_{\Pi}^{+} \top \qquad (B4)\ \Phi \Vdash_{\Pi}^{+} p(\bot) = 0 \text{ iff } \Phi \Vdash_{\Pi}^{+} \top$$

The following rule is the **probabilistic rule**:

$$(PR1) \quad \Phi \Vdash_{\Pi}^{+} p(\alpha \vee \beta) - p(\alpha) - p(\beta) + p(\alpha \wedge \beta) = 0$$

The following proof rules are the **subject rules**.

$$
\begin{array}{ll}
(S1) & \Phi \Vdash^{+}_{\Pi} f_1 > x \text{ and } f_1 \succeq^{+}_{su} f_2 \text{ implies } \Phi \Vdash^{+}_{\Pi} f_2 > x \\
(S2) & \Phi \Vdash^{+}_{\Pi} f_1 \geq x \text{ and } f_1 \succeq^{+}_{su} f_2 \text{ implies } \Phi \Vdash^{+}_{\Pi} f_2 \geq x \\
(S3) & \Phi \Vdash^{+}_{\Pi} f_1 < x \text{ and } f_1 \succeq^{-}_{su} f_2 \text{ implies } \Phi \Vdash^{+}_{\Pi} f_2 < x \\
(S4) & \Phi \Vdash^{+}_{\Pi} f_1 \leq x \text{ and } f_1 \succeq^{-}_{su} f_2 \text{ implies } \Phi \Vdash^{+}_{\Pi} f_2 \leq x \\
(S5) & \Phi \Vdash^{+}_{\Pi} f_2 < x \text{ and } f_1 \succeq^{+}_{su} f_2 \text{ implies } \Phi \Vdash^{+}_{\Pi} f_1 < x \\
(S6) & \Phi \Vdash^{+}_{\Pi} f_2 \leq x \text{ and } f_1 \succeq^{+}_{su} f_2 \text{ implies } \Phi \Vdash^{+}_{\Pi} f_1 \leq x \\
(S7) & \Phi \Vdash^{+}_{\Pi} f_2 > x \text{ and } f_1 \succeq^{-}_{su} f_2 \text{ implies } \Phi \Vdash^{+}_{\Pi} f_1 > x \\
(S8) & \Phi \Vdash^{+}_{\Pi} f_2 \geq x \text{ and } f_1 \succeq^{-}_{su} f_2 \text{ implies } \Phi \Vdash^{+}_{\Pi} f_1 \geq x
\end{array}
$$

The next rules are the **enumeration rules**.

$$
\begin{aligned}
(E1)\ & \Phi \Vdash^{+}_{\Pi} f_1 \# x \text{ iff } (\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\dots,v_k)\in \Pi^{x,\mathsf{AOp}(f_1)}_{\#}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k) \\
& \qquad \text{if } \Pi^{x,\mathsf{AOp}(f_1)}_{\#} \neq \varnothing \text{ and } \Phi \Vdash^{+}_{\Pi} \bot \text{ otherwise}) \\
(E2)\ & \Phi \Vdash^{+}_{\Pi} f_1 > x \text{ iff } \Phi \Vdash^{+}_{\Pi} \neg(\bigvee_{(v_1,\dots,v_k)\in \Pi^{x,\mathsf{AOp}(f_1)}_{\leq}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)) \\
(E3)\ & \Phi \Vdash^{+}_{\Pi} f_1 \geq x \text{ iff } (\Phi \Vdash^{+}_{\Pi} \neg(\bigvee_{(v_1,\dots,v_k)\in \Pi^{x,\mathsf{AOp}(f_1)}_{<}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)) \\
& \qquad \text{if } \Pi^{x,\mathsf{AOp}(f_1)}_{<} \neq \varnothing \text{ and } \Phi \Vdash^{+}_{\Pi} \neg(\bot) \text{ otherwise}) \\
(E4)\ & \Phi \Vdash^{+}_{\Pi} f_1 < x \text{ iff } \Phi \Vdash^{+}_{\Pi} \neg(\bigvee_{(v_1,\dots,v_k)\in \Pi^{x,\mathsf{AOp}(f_1)}_{\geq}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)) \\
(E5)\ & \Phi \Vdash^{+}_{\Pi} f_1 \leq x \text{ iff } (\Phi \Vdash^{+}_{\Pi} \neg(\bigvee_{(v_1,\dots,v_k)\in \Pi^{x,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)) \\
& \qquad \text{if } \Pi^{x,\mathsf{AOp}(f_1)}_{>} \neq \varnothing \text{ and } \Phi \Vdash^{+}_{\Pi} \neg(\bot) \text{ otherwise})
\end{aligned}
$$

The following proof rules are the **propositional rules**.

$$
\begin{array}{ll}
(P1) & \Phi \Vdash^{+}_{\Pi} \phi_1 \text{ and } \ldots \text{ and } \Phi \Vdash^{+}_{\Pi} \phi_n \text{ and } n \geq 1 \text{ and } \{\phi_1, \ldots, \phi_n\} \vdash \psi \text{ implies } \Phi \Vdash^{+}_{\Pi} \psi \\
(P2) & \text{if } \Phi \vdash \varphi \text{ then } \Phi \Vdash^{+}_{\Pi} \varphi
\end{array}
$$

Example 43. For $\Pi = \{0, 0.2, 0.4, 0.6, 0.8, 1.0\}$, the following illustrate the restricted++ epistemic consequence relation.

- $\{p(\mathtt{A}) + p(\mathtt{B}) \leq 1, p(\mathtt{A}) - p(\mathtt{B}) \geq 1\} \Vdash^{+}_{\Pi} p(\mathtt{A}) + p(\mathtt{B}) = 1$
- $\{p(\mathtt{A}) > 0.8, p(\mathtt{A}) > 0.5 \to p(\mathtt{B}) > 0.5\} \Vdash^{+}_{\Pi} p(\mathtt{B}) > 0.5$
- $\{p(\mathtt{C}) > 0.5 \to p(\mathtt{B}) > 0.5 \wedge p(\mathtt{A}) > 0.5, p(\mathtt{B}) > 0.5 \to p(\mathtt{A}) \leq 0.5\} \Vdash^{+}_{\Pi} \bot$
- $\{p(\mathtt{A}) > 0.6\} \Vdash^{+}_{\Pi} p(\mathtt{A}) = 0.8 \vee p(\mathtt{A}) = 1.0$

A number of valuable additional rules can be derived from our system. We can observe that many of them are straightforward generalizations of the properties from Proposition 4.20:

Proposition 4.31. *Let* $\# \in \{=, \neq, \geq, \leq, >, <\}$ *be the set of inequality relationships, let* Π *be a reasonable restricted value set, and let* $\Pi^{x}_{\#} = \{y \in \Pi \mid x \# y\}$ *be the subset obtained according to the value* x *and relationship* $\#$*. Let* $f_1 : p(\alpha_1) *_1 p(\alpha_2) *_2 \ldots *_{k-1} p(\alpha_k)$ *and* $f_2 : p(\beta_1) \star_1 p(\beta_2) \star_2 \ldots \star_{l-1} p(\beta_l)$*, where* $k, l \geq 1$*,* $\alpha_i, \beta_i \in \mathsf{Terms}(\mathcal{G})$ *and* $\star_j, *_i \in \{+, -\}$ *be operational formulae. The following hold, where* $\vdash$ *is propositional consequence relation,* $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$*,* $\phi, \psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$ *and* $x \in \Pi$*.*

1. $\Phi \Vdash^{+}_{\Pi} p(\alpha) \geq 0$

2. $\Phi \Vdash^{+}_{\Pi} p(\alpha) \leq 1$

3. $\Phi \Vdash^{+}_{\Pi} p(\top) = 1$

4. $\Phi \Vdash^{+}_{\Pi} p(\bot) = 0$

5. $\Phi \Vdash^{+}_{\Pi} f_1 > x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$

6. $\Phi \Vdash^{+}_{\Pi} f_1 < x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$

7. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$

8. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$

9. $\Phi \Vdash^{+}_{\Pi} f_1 \neq x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_1 > x \vee f_1 < x$

10. $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_1 > x \vee f_1 = x$

11. $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_1 < x \vee f_1 = x$

12. $\Phi \Vdash^{+}_{\Pi} f_1 > x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_1 \geq x \wedge f_1 \neq x$

13. $\Phi \Vdash^{+}_{\Pi} f_1 < x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_1 \leq x \wedge f_1 \neq x$

14. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_1 \geq x \wedge f_1 \leq x$

15. $\Phi \Vdash^{+}_{\Pi} f_1 > x$ *iff* $\Phi \Vdash^{+}_{\Pi} \neg(f_1 \leq x)$

16. $\Phi \Vdash^{+}_{\Pi} f_1 < x$ *iff* $\Phi \Vdash^{+}_{\Pi} \neg(f_1 \geq x)$

17. $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ *iff* $\Phi \Vdash^{+}_{\Pi} \neg(f_1 > x)$

18. $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ *iff* $\Phi \Vdash^{+}_{\Pi} \neg(f_1 < x)$

19. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *iff* $\Phi \Vdash^{+}_{\Pi} \neg(f_1 \neq x)$

20. $\Phi \Vdash^{+}_{\Pi} f_1 \neq x$ *iff* $\Phi \Vdash^{+}_{\Pi} \neg(f_1 = x)$

21. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *and* $f_1 \succeq^{+}_{su} f_2$ *implies* $\Phi \Vdash^{+}_{\Pi} f_2 \geq x$

22. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *and* $f_1 \succeq^{-}_{su} f_2$ *implies* $\Phi \Vdash^{+}_{\Pi} f_2 \leq x$

23. $\Phi \Vdash^{+}_{\Pi} f_2 = x$ *and* $f_1 \succeq^{+}_{su} f_2$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$

24. $\Phi \Vdash^{+}_{\Pi} f_2 = x$ *and* $f_1 \succeq^{-}_{su} f_2$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$

25. *if* $f_1 \succeq^{+}_{su} f_2$ *and* $f_2 \succeq^{+}_{su} f_1$, *then* $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_2 = x$

26. *if* $f_1 \succeq^{-}_{su} f_2$ *and* $f_2 \succeq^{-}_{su} f_1$, *then* $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_2 = x$

27. $\Phi \Vdash^{+}_{\Pi} f_1 > x$ *and* $x \geq y$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 > y$

28. $\Phi \Vdash^{+}_{\Pi} f_1 > x$ *and* $x > y$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 > y$

29. $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ *and* $x \geq y$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 \geq y$

30. $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ *and* $x > y$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 > y$

31. $\Phi \Vdash^{+}_{\Pi} f_1 < x$ *and* $x \leq y$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 < y$

32. $\Phi \Vdash^{+}_{\Pi} f_1 < x$ *and* $x < y$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 < y$

33. $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ *and* $x \leq y$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 \leq y$

34. $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ *and* $x < y$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 < y$

35. $\Phi \Vdash^{+}_{\Pi} f_1 = x \wedge f_1 = y$ *where* $x \neq y$ *iff* $\Phi \Vdash^{+}_{\Pi} \bot$

36. $\Phi \Vdash^{+}_{\Pi} f_1 \# x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_2 \# x$ *where* $f_1 \simeq_{re} f_2$

37. $\Phi \Vdash^{+}_{\Pi} f_1 \# x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_1 \star p(\bot) \# x$*, where* $\star \in \{+, -\}$

38. $\Phi \Vdash^{+}_{\Pi} f_1 > x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 + p(\gamma) > x$

39. $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 + p(\gamma) \geq x$

40. $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 - p(\gamma) \leq x$

41. $\Phi \Vdash^{+}_{\Pi} f_1 < x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 - p(\gamma) < x$

42. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *and* $\Phi \Vdash^{+}_{\Pi} f_2 = y$ *and* $z = x + y \in \Pi$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 + f_2 = z$

43. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *and* $\Phi \Vdash^{+}_{\Pi} f_2 = y$ *and* $z = x - y \in \Pi$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 - p(\beta_1) \smallstar_1 \dots \smallstar_{l-1} p(\beta_l) = z$*, where* $\smallstar_i = +$ *if* $\star_i = -$ *and* $\smallstar_i = -$ *if* $\star_i = +$

44. $\Phi \Vdash^{+}_{\Pi} p(\alpha) - p(\alpha) = 0$

45. $\Phi \Vdash^{+}_{\Pi} p(\alpha \vee \beta) = x$ *iff* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) = x$

46. $\Phi \Vdash^{+}_{\Pi} p(\alpha \wedge \beta) = x$ *iff* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \vee \beta) = x$

47. *if* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) < 0$ *or* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) > 1$ *then* $\Phi \Vdash^{+}_{\Pi} \bot$

48. *if* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \vee \beta) < 0$ *or* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \vee \beta) > 1$ *then* $\Phi \Vdash^{+}_{\Pi} \bot$

The following is a correctness result showing that the restricted++ epistemic consequence relation is sound with respect to the restricted epistemic entailment relation.

Proposition 4.32. *Let* Π *be a restricted value set. For* $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$*, and* $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$*, if* $\Phi \Vdash^{+}_{\Pi} \psi$ *then* $\Phi \Vdash_{\Pi} \psi$.

In order to prove completeness, we start by proving the valued counterpart of Proposition 4.22, which was shown to be true for basic formulae. In particular, we show that every valued formula can be syntactically brought to its DDNF:

Proposition 4.33. *Let* $\# \in \{=, \neq, \geq, \leq, >, <\}$ *be the set of inequality relationships, let* Π *be a reasonable restricted value set, and let* $\Pi^{x}_{\#} = \{y \in \Pi \mid x \# y\}$ *be the subset obtained according to the value* x *and relationship* $\#$*. Let* $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \dots, c_j\}$ *be the collection of all argument complete propositional formulae for* $\mathcal{G}$ *and* $T^{\Pi}_{v,k} = \Pi^{v,(+,\dots,+)}_{=}$ *s.t. the length of* $(+, \dots, +)$ *is* $k - 1$ *be the collection of k–tuples of values from* Π *that sum up to* $v \in \Pi$*. The following hold, where* $\vdash$ *is propositional consequence relation,* $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$*,* $\phi, \psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$ *and* $x \in \Pi$*. The following hold:*

1. *for consistent formulae* $\alpha_1, \dots \alpha_m \in \mathsf{Terms}(\mathcal{G})$*, if for all* $1 \leq i, j \leq m$ *s.t.* $i \neq j$ *it holds that* $\alpha_i \wedge \alpha_j \vdash \bot$*, then* $\Phi \Vdash^{+}_{\Pi} p(\alpha_1 \vee \dots \vee \alpha_m) = x$ *iff* $\Phi \Vdash^{+}_{\Pi} \bigvee_{(\tau_1, \dots, \tau_m) \in T^{\Pi}_{x,m}} (p(\alpha_1) = \tau_1 \wedge \dots \wedge p(\alpha_m) = \tau_m)$

2. $\Phi \Vdash^{+}_{\Pi} \bigvee_{(\tau_1, \dots, \tau_j) \in T^{\Pi}_{1,j}} (p(c_1) = \tau_1 \wedge \dots \wedge p(c_j) = \tau_j)$

3. $\Phi \Vdash^{+}_{\Pi} \psi$ *iff* $\Phi \Vdash^{+}_{\Pi} \varphi$*, where* φ *is the distribution disjunctive normal form of* ψ

Example 44. Let us consider a set of arguments $\{\mathtt{A}, \mathtt{B}\}$ and let $\Pi = \{0, 0.5, 1\}$. Assume we have a term $\mathtt{A} \vee \mathtt{B}$ and an epistemic atom $p(\mathtt{A}) + p(\mathtt{B}) > 1$. The satisfying distributions and their associated epistemic formulae are tabulated below. Therefore, the DDNF associated with $p(\mathtt{A}) + p(\mathtt{B}) > 1$ is $\varphi^{P_1} \vee \varphi^{P_2} \vee \varphi^{P_3}$.

	$\varnothing$	$\{\mathtt{A}\}$	$\{\mathtt{B}\}$	$\{\mathtt{A}, \mathtt{B}\}$	φ^{P_i}
P_1	0	0	0	1	$p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 1$
P_2	0	0	0.5	0.5	$p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0.5 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0.5$
P_3	0	0.5	0	0.5	$p(\neg\mathtt{A} \wedge \neg\mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \neg\mathtt{B}) = 0.5 \wedge p(\neg\mathtt{A} \wedge \mathtt{B}) = 0 \wedge p(\mathtt{A} \wedge \mathtt{B}) = 0.5$

Let us now consider constructing DDNF using the proof rules. Based on the enumeration rule E1, $p(\mathtt{A})+p(\mathtt{B})>1$ is equivalent to $(p(\mathtt{A})=1\wedge p(\mathtt{B})=1)\vee(p(\mathtt{A})=0.5\wedge p(\mathtt{B})=1)\vee(p(\mathtt{A})=1\wedge p(\mathtt{B})=0.5)$. The (full) disjunctive normal form for A is $(\mathtt{A}\wedge\neg\mathtt{B})\vee(\mathtt{A}\wedge\mathtt{B})$, and the full DNF for B is $(\neg\mathtt{A}\wedge\mathtt{B})\vee(\mathtt{A}\wedge\mathtt{B})$ We can use the propositional, subject and derivable rules to show that $(p(\mathtt{A})=1\wedge p(\mathtt{B})=1)\vee(p(\mathtt{A})=0.5\wedge p(\mathtt{B})=1)\vee(p(\mathtt{A})=1\wedge p(\mathtt{B})=0.5)$ is equivalent to $(p((\mathtt{A}\wedge\neg\mathtt{B})\vee(\mathtt{A}\wedge\mathtt{B}))=1\wedge p((\neg\mathtt{A}\wedge\mathtt{B})\vee(\mathtt{A}\wedge\mathtt{B}))=1)\vee(p((\mathtt{A}\wedge\neg\mathtt{B})\vee(\mathtt{A}\wedge\mathtt{B}))=0.5\wedge p((\neg\mathtt{A}\wedge\mathtt{B})\vee(\mathtt{A}\wedge\mathtt{B}))=1)\vee(p((\mathtt{A}\wedge\neg\mathtt{B})\vee(\mathtt{A}\wedge\mathtt{B}))=1\wedge p((\neg\mathtt{A}\wedge\mathtt{B})\vee(\mathtt{A}\wedge\mathtt{B}))=0.5)$ This, by the first point of Proposition 4.33, propositional and derivable rules (in particular, rule 33) is equivalent to $(p(\neg\mathtt{A}\wedge\mathtt{B})=1\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=1\wedge p(\mathtt{A}\wedge\mathtt{B})=0)\vee(p(\neg\mathtt{A}\wedge\mathtt{B})=0\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=0\wedge p(\mathtt{A}\wedge\mathtt{B})=1)\vee(p(\neg\mathtt{A}\wedge\mathtt{B})=0.5\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=0.5\wedge p(\mathtt{A}\wedge\mathtt{B})=0.5)\vee(p(\neg\mathtt{A}\wedge\mathtt{B})=1\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=0.5\wedge p(\mathtt{A}\wedge\mathtt{B})=0)\vee(p(\neg\mathtt{A}\wedge\mathtt{B})=0.5\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=0\wedge p(\mathtt{A}\wedge\mathtt{B})=0.5)\vee(p(\neg\mathtt{A}\wedge\mathtt{B})=0.5\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=1\wedge p(\mathtt{A}\wedge\mathtt{B})=0)\vee(p(\neg\mathtt{A}\wedge\mathtt{B})=0\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=0.5\wedge p(\mathtt{A}\wedge\mathtt{B})=0.5)$. We then use the second point of the aforementioned proposition along with suitable propositional and derivable rules in order to insert the epistemic atoms associated with the term $\neg\mathtt{A}\wedge\neg\mathtt{B}$ into our formula and to remove those conjunctive clauses that are in fact unsatisfiable (i.e. the first, third, fourth and 6th clause). This brings us to the epistemic formula $(p(\neg\mathtt{A}\wedge\neg\mathtt{B})=0\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=0\wedge p(\neg\mathtt{A}\wedge\mathtt{B})=0\wedge p(\mathtt{A}\wedge\mathtt{B})=1)\vee\vee(p(\neg\mathtt{A}\wedge\neg\mathtt{B})=0\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=0\wedge p(\neg\mathtt{A}\wedge\mathtt{B})=0.5\wedge p(\mathtt{A}\wedge\mathtt{B})=0.5)\vee(p(\neg\mathtt{A}\wedge\neg\mathtt{B})=0\wedge p(\mathtt{A}\wedge\neg\mathtt{B})=0.5\wedge p(\neg\mathtt{A}\wedge\mathtt{B})=0\wedge p(\mathtt{A}\wedge\mathtt{B})=0.5)$. It is easy to see that our result coincides with the formula we have obtained through semantical means.

The ability to transform any formula into its distribution disjunctive normal form both semantically and syntactically, along with the previous soundness results, brings us to the final correctness result for our system:

Proposition 4.34. *Let Π be a restricted value set. For $\Phi\subseteq\mathsf{VFormulae}(\mathcal{G},\Pi)$, and $\psi\in\mathsf{VFormulae}(\mathcal{G},\Pi)$, $\Phi\Vdash^{+}_{\Pi}\psi$ iff $\Phi\Vvdash_{\Pi}\psi$.*

In addition, similarly as in the basic system, the following property can be shown:

Proposition 4.35. *Let Π be a reasonable restricted value set. For $\Phi\subseteq\mathsf{VFormulae}(\mathcal{G},\Pi)$ and $\psi\in\mathsf{VFormulae}(\mathcal{G},\Pi)$, $\Phi\Vdash^{+}_{\Pi}\psi$ iff $\Phi\cup\{\neg\psi\}\Vdash^{+}_{\Pi}\bot$.*

Given that both the basic and the valued systems are sound and complete and that DDNF of any epistemic formula is a basic one as well, the following can be proved:

Proposition 4.36. *Let Π be a reasonable restricted value set, $\Phi\subseteq\mathsf{BFormulae}(\mathcal{G},\Pi)$ be a set of basic restricted formulae and $\varphi\in\mathsf{BFormulae}(\mathcal{G},\Pi)$ a basic restricted formulae. If $\Phi\Vdash_{\Pi}\varphi$ then $\Phi\Vdash^{+}_{\Pi}\varphi$.*

Proposition 4.37. *Let Π be a reasonable restricted value set, and let the restricted valued language w.r.t. Π be $\mathsf{VFormulae}(\mathcal{G},\Pi)$. There is a function $f:\mathsf{VFormulae}(\mathcal{G},\Pi)\to\mathsf{BFormulae}(\mathcal{G},\Pi)$ s.t. for each $\{\phi_1,\ldots,\phi_n\}\subseteq\mathsf{VFormulae}(\mathcal{G},\Pi)$, and for each $\psi\in\mathsf{VFormulae}(\mathcal{G},\Pi)$, $\{\phi_1,\ldots,\phi_n\}\Vdash^{+}_{\Pi}\psi$ iff $\{f(\phi_1),\ldots,f(\phi_n)\}\Vdash_{\Pi}f(\psi)$.*

Proposition 4.38. *The restricted basic language with the restricted epistemic consequence relation is equivalent to the restricted valued language with the restricted++ epistemic consequence relation.*

The above result, paired with Proposition 4.27, also means that for a finite set of rational numbers from the unit interval Π, representing and reasoning with the restricted constraint language w.r.t. Π is equivalent to propositional logic.

Lemma 4.39. *Let Π be a reasonable restricted value set, and let the restricted constraint language w.r.t. Π be $\mathsf{VFormulae}(\mathcal{G},\Pi)$. There is a set of propositional formulae Ω with $\Lambda\subseteq\Omega$, and there is a function $f:\mathsf{VFormulae}(\mathcal{G},\Pi)\to\Omega$ s.t. for each $\{\phi_1,\ldots,\phi_n\}\subseteq\mathsf{VFormulae}(\mathcal{G},\Pi)$, and for each $\psi\in\mathsf{VFormulae}(\mathcal{G},\Pi)$,*

$$\{\phi_1,\ldots,\phi_n\}\Vdash^{+}_{\Pi}\psi \textit{ iff } \{f(\phi_1),\ldots,f(\phi_n)\}\cup\Lambda\vdash f(\psi)$$

From Lemma 4.39 and Lemma 4.26, we obtain the following result.

Proposition 4.40. *The restricted constraint language with the restricted++ epistemic consequence relation is equivalent to the classical propositional language with the classical propositional consequence relation.*

4.5 Closure

Last, but not the least, we define the notion of an epistemic closure, which will become particularly useful in the analysis of relation coverage and labelings in Sections 5.2.2 and 5.3. To put it simply, closure produces the set of all formulae derivable from a given set:

Definition 4.41. Let $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G})$. The **epistemic closure function** is defined as follows.

$$\mathsf{Closure}(\Phi) = \{\psi \mid \Phi \Vdash \psi\}$$

We can observe that closure can produce infinitely many formulae that, depending on how we intend to use it, can be seen as redundant. For example, from a formula $p(\mathtt{A}) > 0.5$ we can derive $p(\mathtt{A}) > y$ for every real number $y \in [0, 0.5]$. Consequently, in many cases it makes sense to focus on closure w.r.t. a given reasonable restricted set of values Π:

Definition 4.42. Let Π be a reasonable restricted value set, and let $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$. The **restricted epistemic closure function** is defined as follows.

$$\mathsf{Closure}(\Phi, \Pi) = \{\psi \mid \Phi \Vdash_{\Pi} \psi\}$$

Given the soundness and completeness results for our proof systems, we can observe that closure can also be defined using $\Vdash_{\Pi}$ and $\Vdash^{+}_{\Pi}$. The closure function is monotonic on both of its arguments (i.e. if $\Phi \subseteq \Phi'$ and $\Pi \subseteq \Pi'$, then $\mathsf{Closure}(\Phi, \Pi) \subseteq \mathsf{Closure}(\Phi, \Pi)$).

Example 45. Let us consider the reasonable restricted value set $\Pi = \{0, 0.1, 0.2, \ldots, 0.9, 1\}$ and the set of formulae $\Phi = \{p(\mathtt{A}) > 0.5, (p(\mathtt{B}) > 0.5 \land p(\mathtt{A}) > 0.4) \rightarrow p(\mathtt{C}) > 0.6, p(\mathtt{C}) = 1 \rightarrow p(\mathtt{B}) = 0.9\}$. We can observe that $\Phi \Vdash_{\Pi} p(\mathtt{A}) \leq x$ for $x \in \{0, 0.1, 0.2, 0.3, 0.4\}$, thus these formulae belong to the (both restricted and standard) closure of Φ. On the other hand, the formula $p(\mathtt{A}) = 0.7$ does not. The formula $p(\mathtt{A}) > 0.5 \land (p(\mathtt{B}) \leq 0.5 \lor p(\mathtt{C}) > 0.6) \land (p(\mathtt{C}) < 1 \lor p(\mathtt{B}) = 0.9)$ also belongs to the closure. The formula $p(\mathtt{B}) = 0.8 \land p(\mathtt{C}) = 0.2$ does not.

5 Properties of Epistemic Graphs

In the previous sections we have introduced epistemic constraints, epistemic graphs and their semantics, and provided a sound and complete proof theory for reasoning with the constraints. In this section, we would like to analyze various properties that epistemic graphs have. We start by distinguishing various subclasses and forms of epistemic graphs and then analyze how the constraints affect arguments and relations present in the graph. Finally, we analyze how the relation labels present in the graph are reflected by the constraints.

5.1 Types of Epistemic Graphs

The way epistemic graphs are defined gives us a lot of freedom. However, for some applications, we may not require all the features of the graph. Furthermore, certain types of analyses might be easier to carry out when we consider graphs in which the form of the constraints is more restricted. Thus, in this section we will distinguish various types and forms of epistemic graphs. We start by differentiating between graphs that use the basic, target or source types of constraints, previously discussed in Section 3.1:

Definition 5.1. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph. Then X is:

- **basic** iff $\mathcal{C} \subseteq \mathsf{BCon}(\mathcal{G})$
- **target style** iff for every $\varphi \in \mathcal{C}$ there is an argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ s.t. φ is a target constraint for $\mathtt{A}$
- **full target style** iff for every $\varphi \in \mathcal{C}$ there is an argument$\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ s.t. φ is a full target constraint for $\mathtt{A}$
- **source style** iff for every $\varphi \in \mathcal{C}$ there is an argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ s.t. φ is a source constraint for $\mathtt{A}$

- **full source style** iff for every $\varphi \in \mathcal{C}$ there is an argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ s.t. φ is a full source constraint for $\mathtt{A}$

Proposition 5.2. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph s.t. $\mathcal{C}$ is finite and every $\varphi \in \mathcal{C}$ is of finite length. There might not exist a graph $X' = (\mathcal{G}, \mathcal{L}, \mathcal{C}')$ s.t. $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$, $\mathcal{C}'$ is finite, every $\varphi' \in \mathcal{C}'$ is of finite length, and X' is basic, target style, or source style.*

Example 46. Let us consider a simple graph $\mathcal{G} = (\{\mathtt{A}, \mathtt{B}\}, \varnothing)$ with an empty labeling function and the set of constraints $\mathcal{C} = \{p(\mathtt{A}) = 0 \lor p(\mathtt{B}) = 1\}$. An equivalent target and source style constraint set would have to be composed of constraints that either contain only A or only B since there are no relations in our graph. Hence, we would have to create a set of constraints $\mathcal{C}' = \{\varphi_1, \dots, \varphi_n\}$ s.t. for every φ_i, $\mathsf{FArgs}(\varphi_i) = \{\mathtt{A}\}$ or $\mathsf{FArgs}(\varphi_i) = \{\mathtt{B}\}$, and $\mathsf{Sat}(p(\mathtt{A}) = 0 \lor p(\mathtt{B}) = 1) = \mathsf{Sat}(\{\varphi_1, \dots, \varphi_n\})$, which based on properties of Sat is also equivalent to $\mathsf{Sat}(\varphi_1 \land \dots \land \varphi_n)$. Without the loss of generality we can also assume that every φ_i is a disjunction of probabilistic literals. Thus, in reality, constructing $\mathcal{C}'$ would be similar to constructing a conjunctive normal form of $p(\mathtt{A}) = 0 \lor p(\mathtt{B}) = 1$ s.t. every disjunctive clause contains only A or only B, and intuitively this is not possible.

In order to see that not every valued constraint can be equivalently expressed with a finite set of basic constraints, please consult Example 17.

Previous properties analyzed the constraints more in terms of the relations between the arguments contained in them. However, it also makes sense to impose certain structural restrictions on the constraints. We introduce the clausal form of the epistemic graph, i.e. one in which every constraint is a clause, and the single form, where we only deal with at most one constraint.

Definition 5.3. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph. Then X is:

- **single** iff $|\mathcal{C}| \leq 1$
- **clausal** iff every $\varphi \in \mathcal{C}$ is a disjunctive clause built from epistemic atoms or their negations

The constraints are added to the labelled graph with the purpose of creating distributions satisfying them. Consequently, there is an implicit conjunction between all of the constraints in the epistemic graph. By making it explicit, we can create the single form of the epistemic graph:

Definition 5.4. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph and $\mathcal{C} = \{\varphi_1, \dots, \varphi_m\}$ its set of constraints. The **single form** of X, denoted $\mathsf{Single}(X)$, is the epistemic graph $X' = (\mathcal{G}, \mathcal{L}, \mathcal{C}')$, where $\mathcal{C}' = \mathcal{C}$ if $m = 0$ and $\mathcal{C}' = \{\varphi_1 \land \dots \land \varphi_m\}$ otherwise.

The following property follows straightforwardly:

Proposition 5.5. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph and $X' = (\mathcal{G}, \mathcal{L}, \mathcal{C}')$ its single form. Then $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$.*

Example 47. Consider a simple graph $\mathcal{G} = (\{\mathtt{A}, \mathtt{B}, \mathtt{C}\}, \{(\mathtt{A}, \mathtt{B}), (\mathtt{B}, \mathtt{C})\})$ s.t. $\mathcal{L}((\mathtt{A}, \mathtt{B})) = \mathcal{L}((\mathtt{B}, \mathtt{C})) = \{-\}$ and the following set of constraints $\mathcal{C}$:

- $\varphi_1 : p(\mathtt{A}) > 0.5$
- $\varphi_2 : p(\mathtt{B}) + p(\mathtt{A}) \leq 1$
- $\varphi_2 : p(\mathtt{C}) + p(\mathtt{B}) \leq 1$

The set of satisfying distributions for $\mathcal{C}$ is, by definition, $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\varphi_1) \cap \mathsf{Sat}(\varphi_2) \cap \mathsf{Sat}(\varphi_3)$. Again, by the definition of a satisfying distribution, it also holds that $\mathsf{Sat}(\varphi_1) \cap \mathsf{Sat}(\varphi_2) \cap \mathsf{Sat}(\varphi_3) = \mathsf{Sat}(\varphi_1 \land \varphi_2 \land \varphi_3)$. Hence, replacing $\mathcal{C}$ by the set $\mathcal{C}' = \{\varphi_4 : p(\mathtt{A}) > 0.5 \land p(\mathtt{B}) + p(\mathtt{A}) \leq 1 \land p(\mathtt{C}) + p(\mathtt{B}) \leq 1$ does not affect the set of satisfying distributions at all.

This observation also allows us to turn a graph into its clausal equivalent. In order to turn an arbitrary graph into a clausal graph, it suffices to transform every constraint into its conjunctive normal form (see Section 4.1) and take every disjunctive clause as a new constraint. Thus, an explicit conjunction now becomes implicit.

Definition 5.6. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph and $\mathcal{C} = \{\varphi_1, \ldots, \varphi_m\}$ its set of constraints. For a constraint φ_i, let $\mathsf{d\text{-}CNF}(\varphi_i) = \psi^i_1 \wedge \ldots \wedge \psi^i_k$ be its d-CNF and let $\mathsf{sep}(\varphi_i) = \{\psi^i_1, \ldots, \psi^i_k\}$ be the set of the disjunctive clauses of the d-CNF of φ_i. The **clausal form** of X, denoted $\mathsf{Clausal}(X)$, is the graph $X' = (\mathcal{G}, \mathcal{L}, \mathcal{C}')$, where $\mathcal{C}' = \mathcal{C}$ if $m = 0$ and $\mathcal{C}' = \bigcup_{i=1^m} \mathsf{sep}(\mathsf{d\text{-}CNF}(\varphi_i))$.

Again, the following property is easy to show:

Proposition 5.7. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph and* $\mathsf{Clausal}(X) = (\mathcal{G}, \mathcal{L}, \mathcal{C}')$ *its clausal form. Then* $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$.

Example 48. Consider a simple graph $\mathcal{G} = (\{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}\}, \{(\mathtt{A}, \mathtt{C}), (\mathtt{B}, \mathtt{C}), (\mathtt{C}, \mathtt{D})\})$ such that $\mathcal{L}((\mathtt{A}, \mathtt{C})) = \mathcal{L}((\mathtt{B}, \mathtt{C})) = \mathcal{L}((\mathtt{C}, \mathtt{D})) = \{-\}$ and the following set of constraints $\mathcal{C}$:

- $\varphi_1 : p(\mathtt{A}) > 0.5$
- $\varphi_2 : (p(\mathtt{A}) > 0.5 \vee p(\mathtt{B}) > 0.5) \rightarrow p(\mathtt{C}) < 0.5$
- $\varphi_3 : p(\mathtt{D}) + p(\mathtt{C}) \leq 1$

Let us now bring $\mathcal{C}$ to clausal form. We can observe that φ_1 is does not need to undergo any changes. However, φ_2 has to be modified. By bringing φ_2 to d-CNF, we obtain the formula $(\neg p(\mathtt{A}) > 0.5 \vee p(\mathtt{C}) < 0.5) \wedge (\neg p(\mathtt{B}) > 0.5 \vee p(\mathtt{C}) < 0.5)$, which is further split into two new constraints $\varphi_2' : \neg p(\mathtt{A}) > 0.5 \vee p(\mathtt{C}) < 0.5$ and $\varphi_2'' : \neg p(\mathtt{B}) > 0.5 \vee p(\mathtt{C}) < 0.5$. Finally, φ_3 is also already in clause form and remains unaffected. The new set of constraints is therefore $\mathcal{C}' = \{\varphi_1, \varphi_2', \varphi_2'', \varphi_3\}$ and $(\mathcal{G}, \mathcal{L}, \mathcal{C}')$ is a clausal epistemic graph. It is easy to check that $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$.

Another interesting property concerns whether the set constraints contains any redundancies, by which we understand constraints whose removal does not affect the collection of satisfying distributions. Let us consider the following example:

Example 49. Let us consider a graph with nodes A, B and C and the set of constraints $\mathcal{C} = \{\varphi_1 = (p(\mathtt{A}) > 0.5 \wedge p(\mathtt{A}) > 0.5) \rightarrow p(\mathtt{C}) < 0.5,\ \varphi_2 = p(\mathtt{A}) \geq 0.5,\ \varphi_3 = p(\mathtt{B}) > 0.6,\ \varphi_4 = p(\mathtt{B}) > 0.5 \vee p(\mathtt{B}) \leq 0.5,\ \varphi_5 = p(\mathtt{B}) > 0.7\}$. We can observe that the constraint φ_4 is just a tautology and in no way contributes to the graph. Furthermore, given the constraint φ_5, φ_3 also becomes redundant.

Thus, we can consider imposing certain minimality restrictions on the set of constraints. By this we understand that removing any of the formulae from the set leads to admitting more distributions:

Definition 5.8. Let $\mathcal{C} = \{\phi_1, \ldots, \phi_m\}$ be a set of epistemic constraints. Then $\mathcal{C}$ is **minimal** iff $m = 0$ or for all $\phi \in \mathcal{C}$, $\mathsf{Sat}(\mathcal{C}) \subset \mathsf{Sat}(\mathcal{C} \setminus \{\phi\})$.

It is worth noting that if we tried to bring a set of constraints to a minimal form by removing unnecessary conditions, we could produce more than one equivalent version of $\mathcal{C}$:

Example 50. Let us consider a graph with nodes A, B and C and the following set of constraints $\mathcal{C}$:

- $\varphi_1 = p(\mathtt{A}) \geq 0.5$
- $\varphi_2 = p(\mathtt{B}) > 0.6$
- $\varphi_3 = p(\mathtt{A}) \geq 0.5 \wedge p(\mathtt{B}) > 0.6$

We can observe either φ_3, or jointly φ_1 and φ_2 could be removed from the graph in order to obtain a set of constraints without redundant formulae.

Another thing worth noting is that the aforementioned notion of minimality is not necessarily the strongest one that we could impose. While we would have no problems removing redundant formulae from the set $\{p(\mathtt{A}) > 0.5, p(\mathtt{A}) > 0.6\}$, the set $\{p(\mathtt{A}) > 0.5 \wedge p(\mathtt{A}) > 0.6\}$ would remain unchanged. This is because simple constraint removal is vulnerable to single form transformation. This issue could be addressed by analyzing not the constraint itself, but its conjunctive normal form, similarly as we had done in the case of graph clausal form. Hence, we can propose the following, stronger definition of minimality:

Definition 5.9. Let $\mathcal{C} = \{\varphi_1, \dots, \varphi_m\}$ be a set of constraints. For a constraint φ_i, let $\text{d-CNF}(\varphi_i) = \psi_1^i \wedge \dots \wedge \psi_k^i$ be its d-CNF and let $\mathsf{sep}(\varphi_i) = \{\psi_1^i, \dots, \psi_k^i\}$ be the set of the disjunctive clauses of the d-CNF of φ_i. A set of constraints $\mathcal{C}$ is **strongly minimal** iff $m = 0$ or for every $\varphi_i \in \mathcal{C}$ and every $\psi_j^i \in \mathsf{sep}(\varphi_i)$, $\mathsf{Sat}(\mathcal{C}) \subset \mathsf{Sat}((\mathcal{C} \setminus \{\phi\}) \cup (\mathsf{sep}(\varphi_i) \setminus \{\psi_j^i\})$.

Example 51. Let us check whether the set of constraints $\mathcal{C} = \{\varphi_1 : p(\mathtt{A}) > 0.5, \varphi_2 : (p(\mathtt{A}) > 0.5 \vee p(\mathtt{B}) > 0.5) \rightarrow p(\mathtt{C}) < 0.5, \varphi_3 : p(\mathtt{D}) + p(\mathtt{C}) \leq 1\}$ is strongly minimal.

We recall that $\mathcal{C}' = \{\varphi_1, \varphi_2', \varphi_2'', \varphi_3\}$, where $\varphi_2' : \neg p(\mathtt{A}) > 0.5 \vee p(\mathtt{C}) < 0.5$ and $\varphi_2'' : \neg p(\mathtt{B}) > 0.5 \vee p(\mathtt{C}) < 0.5$, is the clausal form of $\mathcal{C}$ and that $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$.

First of all, we can observe that $\mathsf{sep}(\varphi_1) = \{\varphi_1\}$. Thus, we want to verify whether $\mathsf{Sat}(\mathcal{C}) \subset \mathsf{Sat}(\mathcal{C} \setminus \{\varphi_1\})$, and in this case the answer is yes. For example, a probability distribution P s.t. $P(\mathtt{A}) = P(\mathtt{B}) = P(\mathtt{D}) = 0$ and $P(\mathtt{C}) = 1$ is a satisfying distribution for $\mathcal{C} \setminus \{\varphi_1\}$, but not for $\mathcal{C}$.

We can now analyze φ_2. We observe that $\mathsf{sep}(\text{d-CNF}(\varphi_2)) = \{\varphi_2', \varphi_2''\}$. We first check whether $\mathsf{Sat}(\mathcal{C}) \subset \mathsf{Sat}((\mathcal{C} \setminus \{\varphi_2\}) \cup \{\varphi_2''\})$. Given that $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$, this is equivalent to checking that $\mathsf{Sat}(\{\varphi_1, \varphi_2', \varphi_2'', \varphi_3\}) \subset \mathsf{Sat}(\{\varphi_1, \varphi_2'', \varphi_3\})$. Since $\mathsf{Sat}(\{\varphi_1, \varphi_2', \varphi_2'', \varphi_3\}) \subseteq \mathsf{Sat}(\{\varphi_1, \varphi_2'', \varphi_3\})$, we only need to show that there is a distribution satisfying the set $\{\varphi_1, \varphi_2'', \varphi_3\}$, but not $\mathcal{C}'$. An example of such a distribution is P s.t. $P(\mathtt{A}) = P(\mathtt{C}) = 1$ and $P(\mathtt{B}) = P(\mathtt{D}) = 0$. Thus, the subset relation is indeed strict. The next step is to test whether $\mathsf{Sat}(\mathcal{C}) \subset \mathsf{Sat}((\mathcal{C} \setminus \{\varphi_2\}) \cup \{\varphi_2'\})$, which is equivalent to checking if $\mathsf{Sat}(\{\varphi_1, \varphi_2', \varphi_2'', \varphi_3\}) \subset \mathsf{Sat}(\{\varphi_1, \varphi_2', \varphi_3\})$. We can observe that $\mathsf{Sat}(\{\varphi_1, \varphi_2', \varphi_2'', \varphi_3\}) = \mathsf{Sat}(p(\mathtt{A}) > 0.5 \wedge p(\mathtt{C}) < 0.5 \wedge p(\mathtt{D}) + p(\mathtt{C}) \leq 1)$. At the same time, also $\mathsf{Sat}(\{\varphi_1, \varphi_2', \varphi_3\}) = \mathsf{Sat}(p(\mathtt{A}) > 0.5 \wedge p(\mathtt{C}) < 0.5 \wedge p(\mathtt{D}) + p(\mathtt{C}) \leq 1)$. Consequently, removing the φ_2'' part of the φ_2 constraint does not change the satisfiability of the constraint set and $\mathcal{C}$ is not minimal.

Let us now consider a modified version of $\mathcal{C}$, namely $\mathcal{C}'' = \{\varphi_1' : p(\mathtt{A}) \geq 0.5, \varphi_2 : (p(\mathtt{A}) > 0.5 \vee p(\mathtt{B}) > 0.5) \rightarrow p(\mathtt{C}) < 0.5, \varphi_3 : p(\mathtt{D}) + p(\mathtt{C}) \leq 1\}$. We can repeat the previous analysis to show that φ_1' is not redundant. The same holds for the φ_2' component of φ_2. Let us therefore consider φ_2''. In this case, we can construct a probability distribution P s.t. $P(\mathtt{A}) = 0.5$, $P(\mathtt{B}) = P(\mathtt{C}) = 1$ and $P(\mathtt{D}) = 0$. This distribution does not satisfy $\{\varphi_1, \varphi_2', \varphi_2'', \varphi_3\}$, but it does satisfy $\{\varphi_1, \varphi_2', \varphi_3\}$. Since it is also easy to show that $\mathsf{Sat}(\{\varphi_1, \varphi_2', \varphi_2'', \varphi_3\}) \subseteq \mathsf{Sat}(\{\varphi_1, \varphi_2', \varphi_3\})$, we can conclude that the strict subset relation indeed holds and φ_2'' is not redundant.

Finally, we can verify φ_3. Since $\mathsf{sep}(\varphi_3) = \{\varphi_3\}$, we only need to check whether $\mathsf{Sat}(\mathcal{C}'') \subset \mathsf{Sat}(\mathcal{C}'' \setminus \{\varphi_3\})$. The answer is yes. In particular, we can construct a probability distribution P s.t. $P(\mathtt{A}) = P(\mathtt{C}) = P(\mathtt{D}) = 1$ and $P(\mathtt{B}) = 0$, which satisfies $\mathcal{C}'' \setminus \{\varphi_3\}$ but not $\mathcal{C}''$. Hence, our modified set of constraints is strongly minimal.

Another interesting property concerns how precise the constraints for a given graph are. For example, in the face of a believed attacker, one constraint can state that the attackee should be disbelieved, while another constraint can state that the attackee should be disbelieved up to a given degree. Given the various ways constraints can be defined for a single graph, it makes sense to be able to compare them. A possible way of doing this is by analyzing how specific they are. One set is more specific than another when all the constraints in the first set can be implied by the constraints in the second, but not vice versa:

Definition 5.10. Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be sets of constraints. $\mathcal{C}_1$ is **less specific than** $\mathcal{C}_2$, denoted $\mathcal{C}_1 \leq \mathcal{C}_2$, iff for each $\phi \in \mathcal{C}_1$, $\mathcal{C}_2 \vDash \phi$. $\mathcal{C}_1$ is **strictly less specific than** $\mathcal{C}_2$, denoted $\mathcal{C}_1 < \mathcal{C}_2$, iff $\mathcal{C}_1 \leq \mathcal{C}_2$ and it does not hold that $\mathcal{C}_2 \leq \mathcal{C}_1$. Equivalently, $\mathcal{C}_1 < \mathcal{C}_2$ iff for each $\phi \in \mathcal{C}_1$, $\mathcal{C}_2 \vDash \phi$, and there is a $\phi \in \mathcal{C}_2$, $\mathcal{C}_1 \nvDash \phi$.

The specificity relation of course also manifests itself in the sets of distributions satisfying the set of constraints. In particular, we can observe that a distribution satisfying the more specific constraints will also satisfy the less specific ones:

Proposition 5.11. *Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be sets of constraints: $\mathcal{C}_1 \leq \mathcal{C}_2$ iff* $\mathsf{Sat}(\mathcal{C}_2) \subseteq \mathsf{Sat}(\mathcal{C}_1)$

The more specific a set of constraints is, the less there are satisfying distributions. In this approach, an inconsistent set of constraints is in fact the most specific in the sense that it does not admit any distributions.

Example 52. Consider the following sets of constraints:

- $\mathcal{C}_1 = \{\varphi_1 : (p(\mathtt{A}) > 0.5 \wedge p(\mathtt{B}) > 0.5) \rightarrow p(\mathtt{C}) > 0.5, \varphi_2 : \top \rightarrow p(\mathtt{D}) > 0.5, \varphi_3 : \top \rightarrow p(\mathtt{E}) > 0.5\}$

- $\mathcal{C}_2 = \{\psi_1 : p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{C}) > 0.5, \psi_2 : \top \rightarrow p(\mathtt{D}) > 0.8, \psi_3 : \top \rightarrow p(\mathtt{E} \wedge \mathtt{F}) > 0.5\}$

We can observe that for every i, $\{\psi_i\} \Vdash \varphi_i$. Hence, $\mathcal{C}_1 \leq \mathcal{C}_2$, i.e. $\mathcal{C}_1$ is less specific than $\mathcal{C}_2$. At the same time, we can observe that e.g. $\mathcal{C}_1 \nVdash \psi_3$, thus the relation is in fact strict. Given the definition of $\Vdash$, we can observe that for every i, $\mathsf{Sat}(\psi_i) \subseteq \mathsf{Sat}(\varphi_i)$, and therefore $\mathsf{Sat}(\mathcal{C}_2) \subseteq \mathsf{Sat}(\mathcal{C}_1)$.

5.2 Coverage

Previously, we have stated that it is not necessary for the constraints to account for all arguments and all the relations between them. While the ability to operate a not fully defined framework is valuable from the practical point of view, for example when dealing with limited knowledge about an opponent during a dialogue, having a graph in which the constraints cover all possible scenarios has undeniable benefits. In this section we would like to discuss various levels of coverage that we can demand from an epistemic graph.

Let us start with explaining what we understand by argument coverage. On its own, an argument can be assigned any probability value from $[0,1]$. One of the purposes of the constraints is - as the name suggest - to constrain the range of values that an argument may take, for example by the values assigned to its parents. In other words, coverage means that there is at least one value for the degree of belief of an an argument cannot be assigned, be it straight from the constraints or under certain assumptions concerning the beliefs in other arguments. By relation coverage we will understand that the relation between two arguments has impact on the belief we have in its source or target, i.e. that we can find a situation in which imposing restrictions on the belief in one of the arguments should have an effect on the restrictions in the belief of the other argument.

Given the fact that the constraints can occur between unrelated arguments and that for certain types of relations the belief in an argument is more affected by the arguments it is targeting rather than by those that are its parents, one needs to be careful in deciding against what assumptions coverage should be tested. Consequently, one of the things the definition of coverage should allow is the choice of the arguments against which the checks are performed. We therefore create a definition of constraint combinations which allow us to decide on the degrees of beliefs in arguments that will later be used in coverage checking:

Definition 5.12. Let $F = \{\mathtt{A}_1, \ldots, \mathtt{A}_m\} \subseteq \mathsf{Nodes}(\mathcal{G})$ be a set of arguments. An **exact constraint combination** for F is a set $\mathcal{CC}^F = \{p(\mathtt{A}_1) = x_1, p(\mathtt{A}_2) = x_2, \ldots, p(\mathtt{A}_m) = x_m\}$, where $x_1, \ldots, x_m \in [0,1]$. A **soft constraint combination** for F is a set $\mathcal{CC}^F = \{p(\mathtt{A}_1)\#_1 x_1, p(\mathtt{A}_2)\#_2 \mathtt{A}_2, \ldots, p(f_m)\#_m x_m\}$, where $x_1, \ldots, x_m \in [0,1]$ and $\#_1, \ldots, \#_m \in \{=, \neq, \geq, \leq, >, <\}$. With $\mathcal{CC}^F|_G$ for $G \subseteq \mathsf{Nodes}(\mathcal{G})$ we denote the subset of $\mathcal{CC}^F$ that consists of all and only constraints of $\mathcal{CC}^F$ that are on arguments contained in $F \cap G$.

Verifying if and how the belief in an argument changes given the beliefs in other arguments can possess certain challenges depending on how the set of constraints is defined. Amending the set of constraints with the aforementioned combinations might lead to inconsistencies coming from the fact that the arguments in the combinations themselves are interrelated or because the set of constraints already affects the belief in one of the arguments in the combination by default. Furthermore, we need to take into account the fact that the set of constraints associated with the graph might not be consistent to start with. Consequently, in the following sections we will work under the assumption that we are dealing with a graph that possesses at least one satisfying distribution, and discuss inconsistent constraints in Section 5.2.3.

5.2.1 Argument Coverage

The most basic form of coverage is the default coverage, where we can find a degree of belief that an argument cannot take straightforwardly from the constraints and without imposing additional assumptions:

Definition 5.13. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph. We say that an argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ is **default covered in** X if there is a value $x \in [0,1]$ s.t. $\mathcal{C} \Vdash p(\mathtt{A}) \neq x$.

Example 53. Let us consider the graph depicted in Figure 17 and the associated set of constraints $\mathcal{C}$:

- $p(\mathtt{A}) > 0.5$

- $p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{B}) < 0.5$
- $(p(\mathtt{B}) < 0.5 \wedge p(\mathtt{C}) > 0.5) \rightarrow p(\mathtt{D}) \leq 0.5$
- $p(\mathtt{C}) \leq 0.5 \rightarrow p(\mathtt{D}) > 0.5$

In this case, we can observe that both A and B are covered by default. For example, $\mathcal{C} \Vdash p(\mathtt{A}) \neq 0.5$ and $\mathcal{C} \Vdash p(\mathtt{B}) \neq 0.5$. This comes from the fact that the belief in A is restricted from the very beginning and from it we can derive the restrictions for B. However, arguments C and D are not default covered. Although they are constrained and, for example, it cannot be the case that they are both believed or both disbelieved at the same time, for every belief value $x \in [0,1]$ we can still find a probability distribution P s.t. $p(\mathtt{C}) = x$ (resp. $p(\mathtt{D}) = x$).

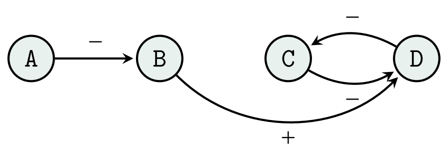


Figure 17: An argument graph

The above example also shows that in some cases, the default coverage may be too restrictive. Although neither C nor D are default covered, the belief we have in one restricts the belief we have in the other. Thus, our intuition is that some form of coverage should exist. In our case, every level of belief we had in C had constrained D and vice versa. However, even weaker forms may be considered:

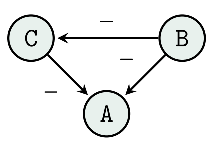


Figure 18: A conflict–based argument graph

Example 54. Let us consider the framework depicted in Figure 18 and the following set of constraints $\mathcal{C}$:

- $\varphi_1 : p(\mathtt{B}) > 0.5 \rightarrow p(\mathtt{C}) \leq 0.5$
- $\varphi_2 : (p(\mathtt{B}) > 0.5 \wedge p(\mathtt{C}) \geq 0.5) \rightarrow p(\mathtt{A}) < 0.5$

Let us analyze how the belief in A is constrained in the graph. Our intuition is that some coverage does exist. In particular, we can observe that if B is believed and C is not disbelieved, then A is disbelieved and thus there are some probabilities it cannot take in this context. However, if this condition is not satisfied, then A can take on any probability. Thus, the coverage is, in a sense, "partial".

We therefore introduce the additional notions of coverage below. We say that an argument is partially covered by a set of arguments F if we can find a belief assignment for F that respects the existing constraints and leads to our argument not being able to take on some values. Full coverage states that every appropriate belief assignment for F should lead to the argument not taking on some values. We can observe that since a consistent graph has at least one probability distribution P satisfying its constraints, then for every set of arguments F we can find a constraint combination $\mathcal{CC}^F$ for F s.t. $\mathcal{CC}^F \cup \mathcal{C}$ is consistent (see also Definition 5.12).

Definition 5.14. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph, $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ an argument and $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ a set of arguments. We say that A is:

- **partially covered by** F in X if there exists a constraint combination $\mathcal{CC}^F$ and a value $x \in [0,1]$ s.t. $\mathcal{CC}^F \cup \mathcal{C} \nVdash \perp$ and $\mathcal{CC}^F \cup \mathcal{C} \Vdash p(\mathtt{A}) \neq x$

- **fully covered by** F in X if for every constraint combination $\mathcal{CC}^F$ s.t. $\mathcal{CC}^F \cup \mathcal{C} \not\Vdash \bot$, there exists a value $x \in [0,1]$ s.t. $\mathcal{CC}^F \cup \mathcal{C} \Vdash p(\mathtt{A}) \neq x$

We can observe that in the above definition, we exclude the effect an argument may have on itself (i.e. the set F cannot contain the argument in question). Let us consider a simple graph with an argument A s.t A is a self–attacker, which can be represented with a constraint $p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{A}) < 0.5$ (i.e. if A is believed, then A is disbelieved). This is equivalent to $p(\mathtt{A}) \leq 0.5$ and leads to default coverage. If we also add the constraint $p(\mathtt{A}) < 0.5 \rightarrow p(\mathtt{A}) > 0.5$ (i.e. if A is disbelieved, then its attackee (and/or attacker) A is believed), which is equivalent to $p(\mathtt{A}) \geq 0.5$, then we can conclude that $p(\mathtt{A}) = 0.5$ which again provides default coverage. Performing a similar analysis for a self–supporter (i.e. if A is believed, then A is believed and if A is disbelieved, then A is disbelieved) leads to a tautology constraint and provides no coverage at all.

Example 55. Let us consider the graph from Example 53 and look at arguments C and D. We can start by analyzing whether arguments A and B provide any coverage for them. We can see that any constraint combination $\{p(\mathtt{A}) = x, p(\mathtt{B}) = y\}$ for these two arguments that is consistent with the existing formulae is such that $x \in (0.5, 1]$ and $y \in [0, 0.5)$. Nevertheless, there is no value $z \in [0,1]$ s.t. the union of our constraint combination and the original set of constraints entails $p(\mathtt{C}) \neq z$ or $p(\mathtt{D}) \neq z$. Consequently, these arguments provide no coverage (be it full or partial), which is in accordance with our intuition. Let us therefore consider constraint combinations on C and analyze the argument D. We can observe that any set $\{p(\mathtt{C}) = v\}$ for $v \in [0,1]$ is consistent with $\mathcal{C}$. For $v \in [0, 0.5]$, we can observe that $\mathcal{C} \cup \{p(\mathtt{C}) = v\} \Vdash p(\mathtt{D}) > 0.5$. Thus, for example, $\mathcal{C} \cup \{p(\mathtt{C}) = v\} \Vdash p(\mathtt{D}) \neq 0$. For $v \in (0.5, 1]$, we can observe that $\mathcal{C} \cup \{p(\mathtt{C}) = v\} \Vdash p(\mathtt{D}) \leq 0.5$. Therefore, for example, $\mathcal{C} \cup \{p(\mathtt{C}) = v\} \Vdash p(\mathtt{D}) \neq 1$. Hence, we can argue that D is both partially and fully covered by $\{\mathtt{C}\}$ (and, as a result, also by sets containing C). Similar arguments can be made for showing that C is partially and fully covered by $\{\mathtt{D}\}$.

Example 56. Let us come back to Example 54 and check whether argument A is covered by the set $\{\mathtt{B}, \mathtt{C}\}$. We can observe that all constraint combinations $\{p(\mathtt{B}) = x, p(\mathtt{C}) = y\}$ are consistent with $\mathcal{C}$ as long as either $x \leq 0.5$ or $y \leq 0.5$. We can observe that $\{p(\mathtt{B}) = 1, p(\mathtt{C}) = 0.5\} \cup \mathcal{C} \Vdash p(\mathtt{A}) < 0.5$. Thus, for example, $\{p(\mathtt{B}) = 1, p(\mathtt{C}) = 0.5\} \cup \mathcal{C} \Vdash p(\mathtt{A}) \neq 1$, and we have at least partial coverage. However, if we consider $\{p(\mathtt{B}) = 0.5, p(\mathtt{C}) = 0.5\}$, then A can be assigned any belief from $[0,1]$. In other words, there is no value $z \in [0,1]$ s.t. $\{p(\mathtt{B}) = 0.5, p(\mathtt{C}) = 0.5\} \cup \mathcal{C} \Vdash p(\mathtt{A}) \neq z$. Thus, the coverage is not full.

In the above partial and full versions of the coverage, we needed to select the arguments against which we wanted to check whether the belief in an argument is restricted or not. For some applications, this extra information might be unnecessary, and thus we can consider the arbitrary versions of partial and full coverage, i.e. ones in which the actual set F is not important as long as least one exists.

Definition 5.15. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph. An argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ has **arbitrary full/partial coverage** iff there exists a set of arguments $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ s.t. A is fully or partially covered w.r.t. F.

For all of the aforementioned versions of coverage, we can also create their restricted counterparts, in which a reasonable restricted value set Π is assumed and the use of $\Vdash$ is replaced with $\Vdash_\Pi$ or $\Vdash^+_\Pi$ whenever appropriate. However, we can also consider an additional form of coverage that is applicable primarily to restricted reasoning and which can be seen as a different development of default coverage. Rather than demanding that a single belief level is not permitted for an argument at all times, we create a number of restrictions s.t. at least one of them holds at a time:

Definition 5.16. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph, Π be a reasonable restricted value set s.t. $|\Pi| = n$ and $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ be an argument. Let $1 \leq k < n$. Then A is **k–covered** w.r.t. Π by $\mathcal{C}$ iff there exists a set of distinct values $\{x_1, \ldots, x_k\} \subseteq \Pi$ s.t. $\mathcal{C} \Vdash^+_\Pi p(\mathtt{A}) \neq x_1 \vee \ldots \vee p(\mathtt{A}) \neq x_k$. For $k \geq 2$, A is **strictly k–covered** w.r.t. Π by $\mathcal{C}$ iff it is k–covered but not (k-1)–covered.

We can observe that testing for 1–coverage would lead to default coverage. Additionally, we do not use n–coverage due to the fact that the resulting formula would be equivalent to a tautology in restricted reasoning, hence providing us with no information.

Example 57. Let us assume we have a reasonable restricted value set $\Pi = \{0, 0.25, 0.5, 0.75, 1\}$ and the following set of constraints:

- $\varphi_1 : p(\mathtt{A}) = 0.25$
- $\varphi_2 : p(\mathtt{A}) < 0.5 \rightarrow p(\mathtt{B}) \leq 0.75$
- $\varphi_3 : p(\mathtt{B}) < 1 \rightarrow (p(\mathtt{C}) \geq 0.5 \vee p(\mathtt{C}) < 0.75)$

We observe that $\mathcal{C} \Vdash_\Pi p(\mathtt{A}) \neq 0 \wedge p(\mathtt{A}) \neq 0.5 \wedge p(\mathtt{A}) \neq 0.75 \wedge p(\mathtt{A}) \neq 1$. Thus, we can show that A is 1–covered, though with more than one value. Additionally, $\mathcal{C} \Vdash_\Pi p(\mathtt{B}) \neq 1$, and B is also 1–covered. Finally, we observe that $\mathcal{C} \Vdash_\Pi (p(\mathtt{C}) \neq 0.25 \wedge p(\mathtt{C}) \neq 0) \vee (p(\mathtt{C}) \neq 1 \wedge p(\mathtt{C}) \neq 0.75)$, which, among other things, means that $\mathcal{C} \Vdash_\Pi p(\mathtt{C}) \neq 0.25 \vee p(\mathtt{C}) \neq 1$, and C is 2–covered in the graph.

The following relationships between the unrestricted forms of coverage can be shown straightforwardly:

Proposition 5.17. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph, $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ be an argument and $F = \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ be a set of arguments. The following hold:*

- *If A is default covered in X, then it is partially and fully covered w.r.t. any set of arguments $G \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$, but not necessarily vice versa*
- *If A is fully covered in X w.r.t. F, then it is partially covered in X w.r.t. F, but not necessarily vice versa*

Given the results of Proposition 4.10, the above proposition also holds for restricted versions of default, full and partial coverage. We can also connect these notions to k–coverage in the following way:

Proposition 5.18. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph, $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ be an argument and $F = \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ be a set of arguments. Let Π be a reasonable restricted value set and $1 \leq k < |\Pi|$. The following hold:*

- *A is 1–covered w.r.t. Π in X iff it is restricted default covered in X*
- *If A is k–covered w.r.t. Π in X, then it is restricted arbitrary fully covered in X, but not necessarily vice versa*
- *For $k < |\Pi| - 1$, if A is k–covered w.r.t. Π in X then it is (k+1)–covered w.r.t. Π in X*

Example 58. Let us come back to Examples 53 and 55 and assume we have a reasonable restricted value set $\Pi = \{0, 0.25, 0.5, 0.75, 1\}$. Our set of constraints $\mathcal{C}$ is:

- $p(\mathtt{A}) > 0.5$
- $p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{B}) < 0.5$
- $(p(\mathtt{B}) < 0.5 \wedge p(\mathtt{C}) > 0.5) \rightarrow p(\mathtt{D}) \leq 0.5$
- $p(\mathtt{C}) \leq 0.5 \rightarrow p(\mathtt{D}) > 0.5$

We can observe that $\mathcal{C} \Vdash_\Pi p(\mathtt{A}) \neq 0 \wedge p(\mathtt{A}) \neq 0.25 \wedge p(\mathtt{A}) \neq 0.5$ and that $\mathcal{C} \Vdash_\Pi p(\mathtt{B}) \neq 0.5 \wedge p(\mathtt{B}) \neq 0.75 \wedge p(\mathtt{B}) \neq 1$. Thus, these arguments are 1–covered, even though it is possible to find more than one value that they cannot take. They are also (restricted and unrestricted) default covered. Arguments C and D are not k–covered for any k, however, they are arbitrary partially and fully covered.

Finally, we can observe that for epistemic graphs that have the same satisfying distributions, the coverage analysis leads to the same results:

Proposition 5.19. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ and $X' = (\mathcal{G}', \mathcal{L}', \mathcal{C}')$ be consistent epistemic graphs s.t. $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$. An argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ is default (partially, fully) covered in X (and w.r.t. $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$) iff it is default (partially, fully) covered in X' (w.r.t. F).*

A similar result can be proved under restricted versions of coverage:

Proposition 5.20. *Let Π be a reasonable restricted value set. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ and $X' = (\mathcal{G}', \mathcal{L}', \mathcal{C}')$ be consistent epistemic graphs s.t. $\mathsf{Sat}(\mathcal{C}, \Pi) = \mathsf{Sat}(\mathcal{C}', \Pi)$. An argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ is restricted default (partially, fully,k–) covered in X iff it is restricted default (partially, fully,k–) covered in X'.*

5.2.2 Relation Coverage

In the previous section we have checked whether an argument is sufficiently covered by the constraints. However, it also makes sense to check whether every relation is covered by the constraints as well. For example, we can consider an argument A and its parents B and C. It is possible that the constraints are defined in a way that only B has an actual effect on A. Thus, the relation between C and A might have no real impact, despite the fact that A may be fully covered in the graph. Hence, we also test for the effectiveness of a given relation, which is understood as the ability of the source to change the belief restrictions on the target argument. We therefore introduce the following definition, which simply states that there is a point at which changing the belief of the source of a relation will lead to a change in the belief we have in the target.

Definition 5.21. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph. Let $(\mathtt{A}, \mathtt{B})$ be a relation in $\mathsf{Arcs}(\mathcal{G})$, $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ and $G = F \setminus \{\mathtt{A}\}$ sets of arguments. Then $(\mathtt{A}, \mathtt{B})$ is:

- **effective w.r.t.** F if there exists a constraint combination $\mathcal{CC}^F$ and values $x, y \in [0,1]$ s.t.
  - $\mathcal{C} \cup \mathcal{CC}^F \not\Vdash \bot$, and
  - $\mathcal{C} \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \not\Vdash \bot$, and
  - at least one of the following conditions holds:
    - $\mathcal{C} \cup \mathcal{CC}^F \not\Vdash p(\mathtt{B}) \neq x$ and $\mathcal{C} \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \Vdash p(\mathtt{B}) \neq x$, or
    - $\mathcal{C} \cup \mathcal{CC}^F \Vdash p(\mathtt{B}) \neq x$ and $\mathcal{C} \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \not\Vdash p(\mathtt{B}) \neq x$.
- **strongly effective w.r.t.** F if for every constraint combination $\mathcal{CC}^F$ s.t. $\mathcal{C} \cup \mathcal{CC}^F \not\Vdash \bot$, there exist values $x, y \in [0,1]$ s.t. $\mathcal{C} \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \not\Vdash \bot$, and at least one of the following conditions holds:
  - $\mathcal{C} \cup \mathcal{CC}^F \not\Vdash p(\mathtt{B}) \neq x$ and $\mathcal{C} \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \Vdash p(\mathtt{B}) \neq x$, or
  - $\mathcal{C} \cup \mathcal{CC}^F \Vdash p(\mathtt{B}) \neq x$ and $\mathcal{C} \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \not\Vdash p(\mathtt{B}) \neq x$.

Example 59. Let us consider a simple set of constraints $\mathcal{C} = \{p(\mathtt{A}) > 0.5 \to p(\mathtt{B}) \leq 0.5\}$ and assume that the $(\mathtt{A}, \mathtt{B})$ relation is present in our graph. We can observe that for every value $z \in [0,1]$, $\{p(\mathtt{A}) = z\} \cup \mathcal{C} \not\Vdash \bot$. Let $z = 1$. Then $\{p(\mathtt{A}) = 1\} \cup \mathcal{C} \Vdash p(\mathtt{B}) \leq 0.5$. Thus, for example, $\{p(\mathtt{A}) = 1\} \cup \mathcal{C} \Vdash p(\mathtt{B}) \neq 1$. However, if we set the probability of A to 0, then B is allowed to take on any probability. Thus, $\{p(\mathtt{A}) = 0\} \cup \mathcal{C} \not\Vdash p(\mathtt{B}) \neq 1$. Hence, the $(\mathtt{A}, \mathtt{B})$ relation is effective. In a similar fashion, we can show that it is strongly effective.

Let us now consider the following set of constraints $\mathcal{C}$ and assume that the $(\mathtt{A}, \mathtt{B})$ and $(\mathtt{C}, \mathtt{B})$ relations are present in the graph:

- $\varphi_1 : p(\mathtt{A}) > 0.5 \to p(\mathtt{B}) > 0.5$
- $\varphi_2 : p(\mathtt{C}) > 0.5 \to p(\mathtt{B}) > 0.9$.

Let us now consider the constraint combinations on $\{\mathtt{A}, \mathtt{C}\}$. We can observe that both $(\mathtt{A}, \mathtt{B})$ and $(\mathtt{C}, \mathtt{B})$ are effective w.r.t. $\{\mathtt{A}, \mathtt{C}\}$. For example, we can take the constraint combination $\{p(\mathtt{A}) = 0, p(\mathtt{C}) = 0\}$ in both cases. We can also observe that $(\mathtt{C}, \mathtt{B})$ is strongly effective w.r.t. $\{\mathtt{A}, \mathtt{C}\}$. For any combination $\{p(\mathtt{A}) = x, p(\mathtt{C}) = y\}$ we can observe changes in the restrictions on the probability of B. If $x \in [0,1]$ and $y \leq 0.5$, then we can replace the constraint for C with $p(\mathtt{C}) = 1$ and observe that $\{p(\mathtt{A}) = x, p(\mathtt{C}) = y\} \cup \mathcal{C} \not\Vdash p(\mathtt{B}) \neq 0.9$ and $\{p(\mathtt{A}) = x, p(\mathtt{C}) = 1\} \cup \mathcal{C} \Vdash p(\mathtt{B}) \neq 0.9$. If $x \in [0,1]$ and $y > 0.5$, then we can take $p(\mathtt{C}) = 0$ to show that $\{p(\mathtt{A}) = x, p(\mathtt{C}) = y\} \cup \mathcal{C} \Vdash p(\mathtt{B}) \neq 0.9$. and $\{p(\mathtt{A}) = x, p(\mathtt{C}) = 0\} \cup \mathcal{C} \not\Vdash p(\mathtt{B}) \neq 0.9$ Hence, in all cases, modifying the belief associated with C changes the restrictions on B. However, $(\mathtt{A}, \mathtt{B})$ is not strongly effective w.r.t. $\{\mathtt{A}, \mathtt{C}\}$. For example, we can consider the combination $\{p(\mathtt{A}) = 0.6, p(\mathtt{C}) = 0.6\}$ for A. We can observe that $\{p(\mathtt{C}) = 0.6\} \cup \mathcal{C} \Vdash p(\mathtt{B}) > 0.9$ and no matter the value of $x \in [0,1]$, adding $p(\mathtt{A}) = x$ to the set will not change the restrictions on B.

The above definition of effectiveness is in fact a rather demanding one in the sense that even though there might exist a constraint from which we can see how two arguments are connected, other constraints in the graph might make it impossible for it to ever become "active", so to speak. For example, coverage such as default, can interfere with detecting the effectiveness of a given relation. Let us consider the following scenario:

Example 60. Let us look at the following set of constraints $\mathcal{C}$ and assume that $(\mathtt{B},\mathtt{A})$ and $(\mathtt{C},\mathtt{A})$ are edges in our graph:

- $\varphi_1 : p(\mathtt{B}) > 0.5 \rightarrow p(\mathtt{C}) < 0.5$
- $\varphi_2 : (p(\mathtt{B}) > 0.5 \wedge p(\mathtt{C}) \geq 0.5) \rightarrow p(\mathtt{A}) < 0.5$

We can observe that even though B and C are not default covered, A is. In particular, $\mathcal{C} \Vdash p(\mathtt{A}) < 0.5$, given that the antecedent of φ_2 will never be satisfied due to the conditions of φ_1. In other words, no constraint combination on $\{\mathtt{B},\mathtt{C}\}$ that is consistent with $\mathcal{C}$ will affect the restrictions on the probability of A. Hence, the $(\mathtt{B},\mathtt{A})$ and $(\mathtt{C},\mathtt{A})$ relations will not be considered effective.

Given this, we can consider a weaker form of effectiveness, where the impact of other constraints may be disregarded. To achieve this, we test effectiveness not against the set of constraints $\mathcal{C}$, but against any consistent set of constraints derivable from it:

Definition 5.22. Let $X = (\mathcal{G},\mathcal{L},\mathcal{C})$ be a consistent epistemic graph. Let $(\mathtt{A},\mathtt{B})$ be a relation in $\mathsf{Arcs}(\mathcal{G})$, $Z \subseteq \mathsf{Closure}(\mathcal{C})$ a consistent set of epistemic constraints, $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ and $G = F \setminus \{\mathtt{A}\}$ sets of arguments. Then $(\mathtt{A},\mathtt{B})$ is:

- **semi–effective w.r.t.** (Z,F) if there exist a constraint combination $\mathcal{CC}^F$ and values $x,y \in [0,1]$ s.t.
  - $Z \cup \mathcal{CC}^F \nVdash \perp$, and
  - $Z \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \nVdash \perp$, and
  - at least one of the following conditions holds:
    - $Z \cup \mathcal{CC}^F \nVdash p(\mathtt{B}) \neq x$ and $Z \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \Vdash p(\mathtt{B}) \neq x$, or
    - $Z \cup \mathcal{CC}^F \Vdash p(\mathtt{B}) \neq x$ and $Z \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \nVdash p(\mathtt{B}) \neq x$.
- **strongly semi–effective w.r.t.** (Z,F) if for every constraint combination $\mathcal{CC}^F$ s.t. $Z \cup \mathcal{CC}^F \nVdash \perp$, there exist values $x,y \in [0,1]$ s.t. $Z \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \nVdash \perp$ and at least one of the following conditions holds:
  - $Z \cup \mathcal{CC}^F \nVdash p(\mathtt{B}) \neq x$ and $Z \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \Vdash p(\mathtt{B}) \neq x$, or
  - $Z \cup \mathcal{CC}^F \Vdash p(\mathtt{B}) \neq x$ and $Z \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) = y\} \nVdash p(\mathtt{B}) \neq x$.

Example 61. Let us come back to Example 60. We could have observed that the $(\mathtt{B},\mathtt{A})$ and $(\mathtt{C},\mathtt{A})$ relations were not effective. Let us take $Z = \{(p(\mathtt{B}) \leq 0.5 \wedge p(\mathtt{C}) < 0.5) \rightarrow p(\mathtt{A}) < 0.5\}$ and $F = \{\mathtt{B},\mathtt{C}\}$. It is easy to verify that $Z \subseteq \mathsf{Closure}(\mathcal{C})$. We observe that any constraint combination on the set $\{\mathtt{B},\mathtt{C}\}$ is consistent with Z, i.e. for any $x,y \in [0,1]$, $\{p(\mathtt{B}) = x, p(\mathtt{C}) = y\} \cup Z \nVdash \perp$. We observe that if $x \leq 0.5$ and $y < 0.5$, then $\{p(\mathtt{B}) = x, p(\mathtt{C}) = y\} \cup Z \Vdash p(\mathtt{A}) < 0.5$. Hence, for example, $\{p(\mathtt{B}) = x, p(\mathtt{C}) = y\} \cup Z \Vdash p(\mathtt{A}) \neq 1$. If we change either x or y in a way that $x > 0.5$ or $y \geq 0.5$, then A can take on any probability. Thus, for such new x' or y', $\{p(\mathtt{B}) = x', p(\mathtt{C}) = y\} \cup Z \nVdash p(\mathtt{A}) \neq 1$ and $\{p(\mathtt{B}) = x, p(\mathtt{C}) = y'\} \cup Z \nVdash p(\mathtt{A}) \neq 1$. Hence, the relations are semi–effective w.r.t (Z,F), even though they were not effective w.r.t. F. They are unfortunately not strongly semi–effective w.r.t. (Z,F). For example, if we took a constraint combination $\{p(\mathtt{B}) = 1, p(\mathtt{C}) = 1\}$, altering the assignment for B (C respectively) would not change the restrictions on A.

The following connections can be drawn between all of these forms of effectiveness:

Proposition 5.23. *Let $X = (\mathcal{G},\mathcal{L},\mathcal{C})$ be a consistent epistemic graph. Let $(\mathtt{A},\mathtt{B})$ be a relation in $\mathsf{Arcs}(\mathcal{G})$, $Z \subseteq \mathsf{Closure}(\mathcal{C})$ a consistent set of epistemic constraints, $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ and $G = F \setminus \{\mathtt{A}\}$ sets of arguments.*

- *If $(\mathtt{A},\mathtt{B})$ is strongly effective w.r.t. F, then it is effective w.r.t. F, but not necessarily vice versa*

- *If* $(\mathtt{A},\mathtt{B})$ *is strongly semi–effective w.r.t.* (Z, F)*, then it is semi–effective w.r.t.* (Z, F)*, but not necessarily and vice versa*
- *If* $(\mathtt{A},\mathtt{B})$ *is effective w.r.t.* F*, then it is semi–effective w.r.t.* $(\mathcal{C}, F)$ *and vice versa*
- *If* $(\mathtt{A},\mathtt{B})$ *is strongly effective w.r.t.* F*, then it is strongly semi–effective w.r.t.* $(\mathcal{C}, F)$ *and vice versa*
- *If* $Z \neq \mathcal{C}$ *and* $(\mathtt{A},\mathtt{B})$ *is semi–effective w.r.t.* (Z, F)*, then it is not necessarily effective w.r.t.* F
- *If* $Z \neq \mathcal{C}$ *and* $(\mathtt{A},\mathtt{B})$ *is strongly semi–effective w.r.t.* (Z, F)*, then it is not necessarily strongly effective w.r.t.* F

5.2.3 Coverage in Inconsistent Epistemic Graphs

In the previous sections we have focused on consistent graphs in our analysis. Given the fact that the coverage was detected based on the notion of epistemic entailment, the existing approaches are not entirely suitable for handling the inconsistent graphs, as seen in the following examples:

(a) $\mathcal{G}_1$ (b) $\mathcal{G}_2$

Figure 19: Examples of conflict–based argument graphs

Example 62. Let us consider the graph depicted in Figure 19a and the following set of constraints $\mathcal{C}$:

- $\varphi_1 = p(\mathtt{A}) > 0.5 \to p(\mathtt{C}) < 0.5 \wedge p(\mathtt{B}) < 0.5$
- $\varphi_2 = p(\mathtt{B}) > 0.5 \to p(\mathtt{A}) < 0.5 \wedge p(\mathtt{C}) < 0.5$
- $\varphi_3 = p(\mathtt{C}) > 0.5 \to p(\mathtt{A}) < 0.5 \wedge p(\mathtt{B}) < 0.5$
- $\varphi_4 = p(\mathtt{A}) < 0.5 \to p(\mathtt{B}) > 0.5 \wedge p(\mathtt{C}) > 0.5$
- $\varphi_5 = p(\mathtt{B}) < 0.5 \to p(\mathtt{A}) > 0.5 \wedge p(\mathtt{C}) > 0.5$
- $\varphi_6 = p(\mathtt{C}) < 0.5 \to p(\mathtt{A}) > 0.5 \wedge p(\mathtt{B}) > 0.5$
- $\varphi_7 = p(\mathtt{A}) \neq 0.5 \wedge p(\mathtt{B}) \neq 0.5 \wedge p(\mathtt{C}) \neq 0.5$

In other words, if an argument is accepted then its attackers and the arguments it attacks are rejected, if an argument is rejected then its attacker has to be accepted, and if the attackers of an argument are rejected then the argument has to be accepted. Finally, no argument can be left undecided[4].

The constraints above describe the stable extensions of this graph, i.e. conflict–free extensions attacking all arguments that are not accepted. Unfortunately, for this framework, there are no such extensions, and the set of constraints is inconsistent. Consequently, for any value $x \in [0,1]$ and argument $\mathtt{Y} \in \{\mathtt{A},\mathtt{B},\mathtt{C}\}$, it holds both that $\mathcal{C} \Vdash p(\mathtt{Y}) = x$ and $\mathcal{C} \Vdash p(\mathtt{Y}) \neq x$, which could be read as argument being both (default) covered and not covered. However, given the constraint φ_7, which from the start states that there are some values that the arguments cannot take, our intuition would be that the arguments are indeed covered.

Let us now consider adding a new argument D to our graph, as depicted in Figure 19b. We do not modify the set of constraints in any way. Hence, again it holds that for any $x \in [0,1]$, $\mathcal{C} \Vdash p(\mathtt{D}) = x$ and $\mathcal{C} \Vdash p(\mathtt{D}) \neq x$. However, despite the fact that $\mathcal{C}$ is inconsistent, our intuition should be that D is not covered.

[4]This could also be enforced using the non–neutral epistemic semantics, however, for the sake of the example we choose to embed it in the constraints.

In the described example, our intuition about whether an argument should be considered covered or not depended on the existence of an appropriate constraint in our set, independently of whether the whole set was consistent or not. Consequently, the coverage definitions from the previous sections could potentially be amended in a way that instead of considering the whole set of constraints $\mathcal{C}$, we analyze its consistent subsets:

Definition 5.24. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph. Then the collections of sets of **maximal consistent constraints** based on $\mathcal{C}$ is defined as $\mathsf{MaxCons}(\mathcal{C}) = \{\mathcal{C}' \subseteq \mathcal{C} \mid \mathcal{C}' \Vdash\!\!\!\!/ \perp$ and there is no $\mathcal{C}'' \subseteq \mathcal{C}$ s.t. $\mathcal{C}' \subset \mathcal{C}$ and $\mathcal{C}'' \Vdash\!\!\!\!/ \perp\}$.

By shifting the requirements previously imposed on $\mathcal{C}$ onto its consistent subsets, we obtain definitions that can be applied to inconsistent graphs. We can also observe that for a consistent set of constraints $\mathcal{C}$, $\mathsf{MaxCons}(\mathcal{C}) = \{\mathcal{C}\}$. Hence, for these graphs, the following notions coincide with the original ones:

Definition 5.25. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph, $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ an argument and $\mathsf{MaxCons}(\mathcal{C})$ the collection of maximal consistent constraint sets based on $\mathcal{C}$. Then $\mathtt{A}$ is **default covered** in X iff there is a value $x \in [0,1]$ and a set of constraints $Z \in \mathsf{MaxCons}(\mathcal{C})$ s.t. $Z \Vdash p(\mathtt{A}) \neq x$.

Definition 5.26. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph, $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ an argument and $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ a set of arguments. We say that $\mathtt{A}$ is:

- **partially covered by** F in X iff there exists a set of constraints $Z \in \mathsf{MaxCons}(\mathcal{C})$ and a constraint combination $\mathcal{CC}^F$ and a value $x \in [0,1]$ s.t. $\mathcal{CC}^F \cup Z \Vdash\!\!\!\!/ \perp$ and $\mathcal{CC}^F \cup Z \Vdash p(\mathtt{A}) \neq x$
- **fully covered by** F in X iff there exists a set of constraints $Z \in \mathsf{MaxCons}(\mathcal{C})$ s.t. for every constraint combination $\mathcal{CC}^F$ s.t. $\mathcal{CC}^F \cup Z \Vdash\!\!\!\!/ \perp$, there exists a value $x \in [0,1]$ s.t. $\mathcal{CC}^F \cup Z \Vdash p(\mathtt{A}) \neq x$

Unfortunately, this approach also has its drawbacks. In particular, it does not perform well on argument graphs in single forms, i.e. where the set of constraints is transformed into a single constraint through conjunction. We can consider the set $\mathcal{C}'$ consisting only of a single constraint φ_8 s.t. $\varphi_8 = \varphi_1 \wedge \ldots \wedge \varphi_7$ from our example. Our intuition is that the coverage results for $\mathcal{C}'$ and $\mathcal{C}$ should be the same. Hence, even though $\mathcal{C}$ and $\mathcal{C}'$ should be considered equivalent, testing whether an appropriate constraint exists is not a method for checking coverage that works for every type of graph. This issue could be addressed similarly as in the case of minimality and strong minimality of a set constraints, by which we understand that a graph can be transformed into a clausal form first and then the subsets of the constraints can be considered.

Nevertheless, two inconsistent sets of constraints can possess different $\mathsf{MaxCons}$ sets. Hence, unlike in the consistent case, the results for one graph cannot be shifted to another:

Proposition 5.27. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ and $X' = (\mathcal{G}, \mathcal{L}, \mathcal{C}')$ be inconsistent epistemic graphs. An argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ that is default (partially, fully) covered in X is not necessarily default (arbitrary partially, arbitrary fully) covered in X' and vice versa.*

Finally, we can observe that similar reasoning as in the case of argument coverage can be straightforwardly applied to relation coverage. In other words, rather than looking at $\mathcal{C}$ (or its closure) in analyzing the effectiveness of a given relation, we consider the maximal consistent constraints based on $\mathcal{C}$ (and their respective closures).

5.3 Constraints vs Relation Labellings

Labellings are useful for indicating the kind of influence one argument has on another. In epistemic graphs, the labels can be either provided during the instantiation process or, similarly as in abstract dialectical frameworks, derived from the constraints. This however begs the question whether the way a relation is labelled is really consistent with the way it is described by the constraints. While taking the labelling as input has the benefit of being informed by the method that has instantiated the graph from a given knowledge base, the derivation approach offers more understanding of the real impact a given relation has on the arguments connected to it. In this section we will focus on comparing the nature of a relation induced from the constraints and the nature defined by the labeling.

Inferring the type of a relation we are dealing with based on how the parent affects the target is not as trivial as one may think. The fact that we are working in a more fine–grained setting which also allows various ways in which arguments may interact, poses a number of challenges. Let us first consider a very simple framework in which an argument A attacks an argument B. The status we assign to B given the status of A changes depending on the semantics we use, even if we remain in a purely conflict–based framework. In the extension–based semantics that adhere to conflict–freeness, it should hold that the acceptance of A implies the rejection of B (i.e. if A is in an extension, then B cannot be). In the labeling–based semantics, and in particular in the admissible case, the acceptance of A implies non–acceptance of B (i.e. if A is assigned in, then B can be out or und). In the weighting and ranking–based semantics, the presence of A increases (or, depending on the approach, decreases) the degree assigned to B w.r.t. the degree that would be assigned to it if A was absent. We are thus dealing with a so–called weakening, i.e. B becomes less "acceptable" in a certain sense. Depending on the postulate we use, the epistemic semantics can tell us that B should be disbelieved, not believed, or believed at least or at most to a given degree, and more. Furthermore, we can derive additional postulates that could potentially handle certain psychological phenomena. For example, we can imagine a case in which a proponent claiming B is answered by his or her opponent with an argument A that, in the opinion of the proponent, is nonsensical and simply fallacious. It can happen that this behaviour of the opponent actually leads the proponent to believe B more strongly. This polarizing effect is not uncommon in arguments on political or religious topics. Finally, if A is understood as a reason for disbelieving B, then a decrease in the belief in A – even one not leading to actual disbelief – can lead to an increase in B – even if not to one causing belief. This is a rather simple case when our opinions become less extreme, when we realize that our reasons behind them are not as important as we thought, which is, for example, a desirable effect during psychotherapy.

Therefore, as we can observe, even a simple attack can have a number of different effects, and this is just one of many other types of relations that can be expressed within an epistemic graph. Additionally, even though two arguments can appear to be positively or negatively related on their own, taking into account the effects of other arguments in the graph might also bring to light other behaviours. Certain works on argument frameworks introduce the notions of indirect relations [30, 29, 25]. For example, one argument can support another, but at the same attack another of its supporters, thus serving as an indirect attacker. It can therefore happen that depending on the context in which we look at two arguments, the perception of the relation between them changes:

Example 63. Let us consider the following scenario with arguments A, B, C and D s.t. B and C group support A s.t. at least one of B and C needs to be believed in order to believe A, B supports D s.t. believing B implies believing D, and D attacks A. This can be depicted with the graph in Figure 20 and expressed with the following set of constraints $\mathcal{C}$:

- $\varphi_1 : p(\mathtt{A}) > 0.5 \to p(\mathtt{B}) > 0.5 \lor p(\mathtt{C}) > 0.5$
- $\varphi_2 : (p(\mathtt{D}) < 0.5 \land (p(\mathtt{B}) > 0.5 \lor p(\mathtt{C}) > 0.5)) \to p(\mathtt{A}) > 0.5$
- $\varphi_3 : p(\mathtt{B}) > 0.5 \to p(\mathtt{D}) > 0.5$
- $\varphi_4 : p(\mathtt{D}) > 0.5 \to p(\mathtt{A}) < 0.5$

If we were to decide on the nature of the B–A relation only from the constraint concerning both of them (i.e. constraints φ_1 and φ_2), then the supporting relation becomes quite apparent. However, if we were to take into account the interactions expressed in constraints φ_1 to φ_4, then we would observe that believing B implies believing D and thus disbelieving A, which is hardly a positive influence.

Example 64. Let us consider the following scenario with arguments A, B and C s.t. if A is believed, then unless C is believed, B is disbelieved. Thus, A carries out an attack that can be overruled by the support from C[5]. We can create the constraint $p(\mathtt{A}) > 0.5 \land p(\mathtt{C}) \leq 0.5 \to p(\mathtt{B}) < 0.5$ to reflect this. The interplay between A and C shows that despite the fact A has primarily a negative effect on B, believing A might not always imply disbelieving B due to the presence of other arguments.

[5]This is an example of how extended argumentation frameworks can be modelled [68]

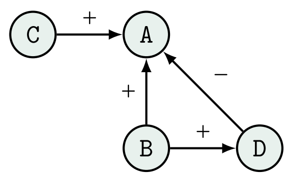


Figure 20: A bipolar labelled graph

Therefore, as we can see, both negative and positive relations can be interpreted in various ways, and their actual influence can change depending on the context in which they are analyzed. Hence, rather than forcing an attack to have a negative effect, we interpret it as a relation not having a positive effect and support as not having a negative effect. In this respect, our approach is similar to the one in abstract dialectical frameworks [18], which as seen in [67] subsumes a wide range of existing methods. However, as motivated by Example 63, we should additionally distinguish between local and global influence, the difference between them being whether all or some (parts of) constraints are taken into account.

What we would also like to observe is that selecting the constraints against which the relations should be tested, is not necessarily an objective process. Let us again look at Example 63:

Example 65. Let us come back to the graph depicted in Figure 20 and analyzed in Example 63. Let us consider replacing the last two constraints φ_3 and φ_4 with the formula $p(\mathtt{B}) > 0.5 \rightarrow p(\mathtt{D}) > 0.5 \wedge p(\mathtt{D}) > 0.5 \rightarrow p(\mathtt{A}) < 0.5$. This in no way affects the produced set of satisfying distributions. We can observe that the new constraint is a target constraint for A. Moreover, from it we can create another constraint $p(\mathtt{B}) > 0.5 \wedge p(\mathtt{D}) > 0.5 \rightarrow p(\mathtt{A}) < 0.5$. Again, adding it to the constraint set in no way affects the satisfying distributions. However, this constraint can be interpreted as a group attack on A by B and D, and if we were to check the local impact that B has on A, the intuition would be against removing this constraint from the testing set. Consequently, despite the logical equivalence of both the original and the modified sets of constraints, the perception of the relations stemming from them might not be the same.

Thus, similarly as in the case of relation coverage, determining the nature of a given relation depends on the constraints that we choose to analyze. Likewise, we will focus on the consistent graphs, and reuse of the methods from Section 5.2.3 in order to handle the inconsistent ones. Let us first consider a simplified, more ternary approach to relations. We will consider a relation attacking if it is semi–effective and s.t. an target argument that is not believed remains as such when the source is believed. In other words, we want to avoid situations when believing the source would lead to believing the target. Support can be defined in a similar fashion, and a dependent relation is seen as neither supporting nor attacking. However, we can observe that, due to the ternary nature of this approach, certain relations can be classified as both supporting and attacking in a way that is not intuitive. In frameworks such as ADFs [18], a relation that is both supporting and attacking is in fact redundant and can be safely removed from the graph without affecting the behaviour of the semantics [68]. Due to the fact that epistemic graphs offer a much more fine–grained approach to acceptability, relations that are both positive and negative do not need to be redundant, it is simply that their effect may be too subtle for this ternary approach. For example, we can consider the constraints $\{p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{B}) > 0.5, p(\mathtt{A}) < 0.5 \rightarrow p(\mathtt{B}) < 0.5, p(\mathtt{A}) > 0.5 \wedge P(\mathtt{C}) > 0.5 \rightarrow p(\mathtt{B}) > 0.9\}$. Thus, we can see that C has a positive effect on B – and a rather significant one at that – but due to the way it is activated, in the ternary approach the relation between C and B is seen as both supporting and attacking. Consequently, in addition to the attacking, supporting and dependent links, we also distinguish between the subtle and unspecified ones.

Definition 5.28. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph. Let $(\mathtt{A}, \mathtt{B})$ be a relation in $\mathsf{Arcs}(\mathcal{G})$, $Z \subseteq \mathsf{Closure}(\mathcal{C})$ a consistent set of epistemic constraints, $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ and $G = F \setminus \{\mathtt{A}\}$ sets of arguments. Then $(\mathtt{A}, \mathtt{B})$ is:

- **supporting w.r.t.** (Z, F) if it is semi–effective w.r.t. (Z, F) and for every every constraint combination $\mathcal{CC}^F$ s.t. $Z \cup \mathcal{CC}^F \nVdash \bot$ and $Z \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) > 0.5\} \nVdash \bot$, if $Z \cup \mathcal{CC}^F \Vdash p(\mathtt{B}) \geq 0.5$ then $Z \cup \mathcal{CC}^F|_G \cup \{p(\mathtt{A}) > 0.5\} \Vdash p(\mathtt{B}) \geq 0.5$.

- **attacking w.r.t.** (Z, F) if it is semi–effective w.r.t. (Z, F) and for every every constraint combination $\mathcal{CC}^F$ s.t. $Z \cup \mathcal{CC}^F \not\Vdash \bot$ and $Z \cup \mathcal{CC}^F|_G \cup \{p(\texttt{A}) > 0.5\} \not\Vdash \bot$, if $Z \cup \mathcal{CC}^F \Vdash p(\texttt{B}) \le 0.5$ then $Z \cup \mathcal{CC}^F|_G \cup \{p(\texttt{A}) > 0.5\} \Vdash p(\texttt{B}) \le 0.5$.
- **dependent w.r.t.** (Z, F) if it semi–effective but neither attacking nor supporting w.r.t. (Z, F)
- **subtle w.r.t.** (Z, F) if it is semi–effective and both attacking and supporting w.r.t. (Z, F)
- **unspecified w.r.t.** (Z, F) if it is not semi–effective w.r.t. (Z, F)

Depending on the choice of constraints and arguments that we use for testing, it can happen that a relation is seen as supporting or attacking due to vacuous truth. For example, we may never find an appropriate combination s.t. $Z \cup \mathcal{CC}^F \Vdash p(\texttt{B}) \ge 0.5$ ($Z \cup \mathcal{CC}^F \Vdash p(\texttt{B}) \le 0.5$), or we cannot find constraint combinations that would be consistent with Z. Consequently, we can also consider the following strengthening:

Definition 5.29. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph. Let $(\texttt{A}, \texttt{B})$ be a relation in $\mathsf{Arcs}(\mathcal{G})$, $Z \subseteq \mathsf{Closure}(\mathcal{C})$ a consistent set of epistemic constraints, $F \subseteq \mathsf{Parent}(\texttt{B})$ and $G = F \setminus \{\texttt{A}\}$ sets of arguments. Then $(\texttt{A}, \texttt{B})$ is **strongly supporting** (resp. attacking, dependent, subtle, unspecified) w.r.t. (Z, F) if for every every constraint combination $\mathcal{CC}^F$ it holds that $Z \cup \mathcal{CC}^F \not\Vdash \bot$ and $Z \cup \mathcal{CC}^F|_G \cup \{p(\texttt{A}) > 0.5\} \not\Vdash \bot$ and there is at least one constraint combination $\mathcal{CC}^F$ s.t. $Z \cup \mathcal{CC}^F \Vdash p(\texttt{B}) \ge 0.5$ (resp. $Z \cup \mathcal{CC}^F \Vdash p(\texttt{B}) \le 0.5$).

Example 66. Let us come back to Example 63. Our graph had four edges, namely $(\texttt{B}, \texttt{A})$, $(\texttt{C}, \texttt{A})$ and $(\texttt{B}, \texttt{D})$ labelled with $+$ and $(\texttt{D}, \texttt{A})$ labelled with $-$. Let us start with the relation $(\texttt{C}, \texttt{A})$. Concerning the relation between these two arguments, we can consider constraints φ_1 and φ_2 as most relevant. We can observe that $\mathcal{C} \Vdash \varphi_1 \wedge \varphi_2 \Vdash (p(\texttt{A}) \le 0.5 \vee p(\texttt{B}) > 0.5 \vee p(\texttt{C}) > 0.5) \wedge (p(\texttt{A}) > 0.5 \vee p(\texttt{C}) \le 0.5 \vee p(\texttt{D}) \ge 0.5)$. Hence, we can take $Z = \{(p(\texttt{A}) \le 0.5 \vee p(\texttt{B}) > 0.5 \vee p(\texttt{C}) > 0.5) \wedge (p(\texttt{A}) > 0.5 \vee p(\texttt{C}) \le 0.5 \vee p(\texttt{D}) \ge 0.5)\}$ as the set of constraints we test against. A possible choice of F in this case is $F = \{\texttt{B}, \texttt{C}, \texttt{D}\}$. We can observe that if we take the sets $W = \{p(\texttt{B}) = 0, p(\texttt{C}) = 0, p(\texttt{D}) = 0\}$ and $W' = \{p(\texttt{B}) = 0, p(\texttt{C}) = 1, p(\texttt{D}) = 0\}$, then $Z \cup W \Vdash p(\texttt{A}) \ne 1$ and $Z \cup W' \not\Vdash p(\texttt{A}) \ne 1$. Thus, the $(\texttt{C}, \texttt{A})$ relation is semi–effective w.r.t. (Z, F). Furthermore, for all value $y_1, y_2, y_3 \in [0, 1]$, $Z \cup \{p(\texttt{B}) = y_1, p(\texttt{C}) = y_2, p(\texttt{D}) = y_3\} \not\Vdash \bot$ and $Z \cup \{p(\texttt{B}) = y_1, p(\texttt{D}) = y_3\} \cup \{p(\texttt{C}) > 0.5\} \not\Vdash \bot$. We can observe that if $y_2 > 0.5$ and $y_3 < 0.5$, then $Z \cup \{p(\texttt{B}) = y_1, p(\texttt{C}) = y_2, p(\texttt{D}) = y_3\} \Vdash p(\texttt{A}) > 0.5$, and if $y_1 \le 0.5$ and $y_2 \le 0.5$, then $Z \cup \{p(\texttt{B}) = y_1, p(\texttt{C}) = y_2, p(\texttt{D}) = y_3\} \Vdash p(\texttt{A}) \le 0.5$. Otherwise, any probability can be assigned to $\texttt{A}$. Hence, for support, we only need to consider the first case, and amending the set of constraints with $p(\texttt{C}) > 0.5$ will not change the outcome. Thus, the $(\texttt{C}, \texttt{A})$ relation is strongly supporting w.r.t. (Z, F). We can consider the second case and amend the constraints in the same way to see that the relation is not attacking.

In order to analyze the $(\texttt{D}, \texttt{A})$ relation, we can take $F = \{\texttt{B}, \texttt{C}, \texttt{D}\}$ and $Z = \{\varphi_2, \varphi_4\}$ (i.e. $Z = \{(p(\texttt{D}) < 0.5 \wedge (p(\texttt{B}) > 0.5 \vee p(\texttt{C}) > 0.5)) \rightarrow p(\texttt{A}) > 0.5, p(\texttt{D}) > 0.5 \rightarrow p(\texttt{A}) < 0.5\}$) to show that the edge is strongly attacking. Concerning $(\texttt{B}, \texttt{D})$, we can use $Z = \{\varphi_3\}$ and $F = \{\texttt{B}\}$ to show that the relation is strongly supporting. However, under these assumptions we cannot find a constraint combination that would entail not believing $\texttt{D}$. Thus, the relation is attacking and therefore subtle (though not strongly) as well.

Finally, we come to $(\texttt{B}, \texttt{A})$. Let us start by looking at constraints φ_1 and φ_2. We can observe that $\mathcal{C} \Vdash \varphi_1 \wedge \varphi_2 \Vdash (p(\texttt{A}) \le 0.5 \vee p(\texttt{B}) > 0.5 \vee p(\texttt{C}) > 0.5) \wedge (p(\texttt{A}) > 0.5 \vee p(\texttt{B}) \le 0.5 \vee p(\texttt{D}) \ge 0.5)$. Hence, we can take $Z = \{(p(\texttt{A}) \le 0.5 \vee p(\texttt{B}) > 0.5 \vee p(\texttt{C}) > 0.5) \wedge (p(\texttt{A}) > 0.5 \vee p(\texttt{B}) \le 0.5 \vee p(\texttt{D}) \ge 0.5)\}$ and $F = \{\texttt{B}, \texttt{C}, \texttt{D}\}$ as our parameters. With this, we can straightforwardly adapt the analysis performed for $(\texttt{C}, \texttt{A})$ in order to show that the relation is strongly supporting and not attacking. Let us now take into account all of the constraints and assume $Z = \mathcal{C}$. We can observe that if F left the way it is, the $(\texttt{B}, \texttt{D})$ relation is in fact unspecified. This is due to the fact that once the values for $\texttt{C}$ and $\texttt{D}$ are set, modifying the value of $\texttt{B}$ leads either to inconsistency (caused by φ_3) or does not change anything anymore. We can therefore reduce the set F to $\{\texttt{B}, \texttt{C}\}$. At this point, we observe that for every $y_1, y_2 \in [0, 1]$, the set $W = \{p(\texttt{B}) = y_1, p(\texttt{D}) = y_2\}$ is consistent with Z. Furthermore, for $y_1 > 0.5$, $Z \cup W \Vdash p(\texttt{A}) < 0.5 \wedge p(\texttt{D}) > 0.5$, for $y_1 \le 0.5$ and $y_2 \le 0.5$, $Z \cup W \Vdash p(\texttt{A}) \le 0.5$, and for $y_1 \le 0.5$ and $y_2 > 0.5$ $\texttt{A}$ can take any probability. We can therefore show that $(\texttt{B}, \texttt{A})$ is strongly attacking w.r.t. (Z, F). However, since we cannot derive $p(\texttt{A}) \ge 0.5$, it is also supporting and subtle, even though not strongly.

In the previous definition we have dealt with positive and negative relations in a more ternary manner, i.e. it only mattered whether the parent and the target were believed, and not up to what degree. Thus, we can also use more refined methods, coming in the form of positive and negative monotony. In this case, if the belief in a node is greater than 0.5, and the other parent nodes have unchanged belief, then a higher belief in that parent node will ensure that there is a higher belief in the destination node. Similarly, the negative monotonic property ensures that if the belief in a node is greater than 0.5, and the other parent nodes have unchanged belief, then a higher belief in that parent node will ensure that there is lower belief in the destination node.

Definition 5.30. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph. Let $(\mathtt{A}, \mathtt{B})$ be a relation in $\mathsf{Arcs}(\mathcal{G})$, $Z \subseteq \mathsf{Closure}(\mathcal{C})$ a consistent set of epistemic constraints and $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ a set of arguments. Then $(\mathtt{A}, \mathtt{B})$ is:

- **positive monotonic** w.r.t. (Z, F) if for every $P, P' \in \mathsf{Sat}(Z)$ s.t.
  - $P(\mathtt{A}) > P'(\mathtt{A}) > 0.5$, and
  - for all $\mathtt{C} \in F$, if $\mathtt{C} \neq \mathtt{A}$ and $\mathtt{C} \neq \mathtt{B}$ then $P(\mathtt{C}) = P'(\mathtt{C})$,

  it holds that $P(\mathtt{B}) > P'(\mathtt{B})$.
- **negative monotonic** w.r.t. (Z, F) if for every $P, P' \in \mathsf{Sat}(Z)$ s.t.
  - $P(\mathtt{A}) > P'(\mathtt{A}) > 0.5$, and
  - for all $\mathtt{C} \in F$, if $\mathtt{C} \neq \mathtt{A}$ and $\mathtt{C} \neq \mathtt{B}$ then $P(\mathtt{C}) = P'(\mathtt{C})$,

  it holds that $P(\mathtt{B}) < P'(\mathtt{B})$.
- **non–monotonic dependent** w.r.t. (Z, F) if it is neither positive nor negative monotonic

Example 67. If we look at Example 66 once more, we can observe that w.r.t. to the previously analyzed (Z, F) pairs, all of the relations are non–monotonic dependent. The constraints, while they can specify whether the target argument should be believed or not, are not specific enough to state the precise degree of this belief. Thus, once the belief threshold is passed, increasing the degree to which the source is believed does not have to be paired with any increase (resp. decrease) For example, let us consider the $(\mathtt{C}, \mathtt{A})$ relation, which is strongly supporting w.r.t. the pair (Z, F) where $Z = \{(p(\mathtt{A}) \leq 0.5 \vee p(\mathtt{B}) > 0.5 \vee p(\mathtt{C}) > 0.5) \wedge (p(\mathtt{A}) > 0.5 \vee p(\mathtt{C}) \leq 0.5 \vee p(\mathtt{D}) \geq 0.5)\}$ and $F = \{\mathtt{B}, \mathtt{C}, \mathtt{D}\}$. In the table below we can see some examples of distributions satisfying Z. We can observe that the increase in the belief in C might lead to increase, decrease, or no change whatsoever in the belief in A. Thus, the relation can only be classified as non–monotonic dependent.

	A	B	C	D
P_1	0.6	0.6	0.6	0
P_2	0.8	0.6	0.7	0
P_3	0.55	0.6	0.7	0
P_4	0.6	0.6	0.7	0

Example 68. Let us now consider a simple graph $(\{\mathtt{A}, \mathtt{B}, \mathtt{C}\}, \{(\mathtt{B}, \mathtt{A}), (\mathtt{C}, \mathtt{A})\}$ where the $(\mathtt{B}, \mathtt{A})$ relation is labelled with + and the $(\mathtt{C}, \mathtt{A})$ relation is labelled with −, and we have a single constraint $\varphi : p(\mathtt{C}) + p(\mathtt{A}) - p(\mathtt{B}) = 1$. Let us focus on the $(\mathtt{B}, \mathtt{A})$ relation. Let us assume that we fix the probability of C to a value x and that $y = 1 - x$. Under this assumption, any probability distribution P satisfying our constraints would be s.t. $P(\mathtt{C}) = x$ and $P(\mathtt{A}) = y + P(\mathtt{B})$. Hence, any increase in the belief in B will result in a proportional increase in the belief in A, and we can show that this relation is positive monotonic w.r.t. $(\{\varphi\}, \{\mathtt{B}, \mathtt{C}\})$. In a similar fashion we can show $(\mathtt{C}, \mathtt{A})$ to be negative monotonic under the same parameters.

The presented definitions could be used to verify whether the relation labels are in some way reflected by the constraints or, if possible, to assign labels to relations when they are missing. A possible - though not the only - way to do so is shown in the definition below.

Definition 5.31. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph. We say that $\mathcal{L}$ is **(strongly) consistent** w.r.t. $\mathcal{C}$ if for every $(\mathtt{A}, \mathtt{B}) \in \mathsf{Arcs}(\mathcal{G})$, the following holds:

- if $+ \in \mathcal{L}((\mathtt{A}, \mathtt{B}))$, then there exists a consistent set $Z \subseteq \mathsf{Closure}(\mathcal{C})$ and a set $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ s.t. $(\mathtt{A}, \mathtt{B})$ is (strongly) supporting w.r.t. (Z, F)
- if $- \in \mathcal{L}((\mathtt{A}, \mathtt{B}))$, then there exists a consistent set $Z \subseteq \mathsf{Closure}(\mathcal{C})$ and a set $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ s.t. $(\mathtt{A}, \mathtt{B})$ is (strongly) attacking w.r.t. (Z, F)
- if $* \in \mathcal{L}((\mathtt{A}, \mathtt{B}))$, then there exists a consistent set $Z \subseteq \mathsf{Closure}(\mathcal{C})$ and a set $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ s.t. $(\mathtt{A}, \mathtt{B})$ is (strongly) dependent w.r.t. (Z, F)
- if $\mathcal{L}((\mathtt{A}, \mathtt{B})) = \varnothing$, then there exists a consistent set $Z \subseteq \mathsf{Closure}(\mathcal{C})$ and a set $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ s.t. $(\mathtt{A}, \mathtt{B})$ is (strongly) unspecified w.r.t. (Z, F)

We say that $\mathcal{L}$ is **monotonic consistent** w.r.t. $\mathcal{C}$ if for every $(\mathtt{A}, \mathtt{B}) \in \mathsf{Arcs}(\mathcal{G})$, the following holds:

- if $+ \in \mathcal{L}((\mathtt{A}, \mathtt{B}))$, then there exists a consistent set $Z \subseteq \mathsf{Closure}(\mathcal{C})$ and a set $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ s.t. $(\mathtt{A}, \mathtt{B})$ is positive monotonic w.r.t. (Z, F)
- if $- \in \mathcal{L}((\mathtt{A}, \mathtt{B}))$, then there exists a consistent set $Z \subseteq \mathsf{Closure}(\mathcal{C})$ and a set $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ s.t. $(\mathtt{A}, \mathtt{B})$ is negative monotonic w.r.t. (Z, F)
- if $* \in \mathcal{L}((\mathtt{A}, \mathtt{B}))$, then there exists a consistent set $Z \subseteq \mathsf{Closure}(\mathcal{C})$ and a set $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ s.t. $(\mathtt{A}, \mathtt{B})$ is non–monotonic dependent w.r.t. (Z, F)

In this case, we could either use the set $\{+, -\}$ to denote subtle relations, or introduce a new label in order to avoid ambiguity.

These approaches can be further refined in the future by putting restrictions on how the Z and F sets are chosen, imposing certain ranking on the relations (for example, if a relation is seen as strongly supporting and not strongly attacking, strong support could take precedence) and/or making the label conditions even stronger (for example, we can demand that $\mathcal{L}((\mathtt{A}, \mathtt{B})) = \varnothing$ iff it $(\mathtt{A}, \mathtt{B})$ unspecified w.r.t. every (Z, F) pair).

Example 69. We can consider the analysis performed in Example 66 to show that the labeling proposed for the graph from Example 63 is strongly consistent with the analyzed set of constraints. The same analysis also shows that it is not the only possible consistent labeling. Following the analysis in Example 67, we can also argue that a labeling that assigns $*$ to every relation would be more adequate based on the monotonicity analysis. We can observe that the labeling for the graph from Example 68 is monotonic consistent with the assumed constraint.

6 Computational Issues

In this section, we will analyze some of the computational properties of epistemic graphs. For this, recall that P denotes the class of (decision) problems decidable in deterministic polynomial time and NP denotes the class of (decision) problems decidable in non-deterministic polynomial time. For any complexity class C, $\mathsf{co}C$ denotes its complement. We will also make use of the polynomial hierarchy that can be defined as follows. Let $\Sigma_0^p = \Delta_0^p = \Pi_0^p = \mathsf{P}$ and define Σ_i^p, Δ_i^p, and Π_i^p for $i > 0$ recursively:

$$\Sigma_{i+1}^p = \mathsf{NP}^{\Sigma_i^p}$$
$$\Pi_{i+1}^p = \mathsf{coNP}^{\Sigma_i^p}$$
$$\Delta_{i+1}^p = \mathsf{P}^{\Sigma_i^p}$$

In particular, $\Sigma_1^p = \mathsf{NP}^{\Sigma_0^p} = \mathsf{NP}^{\mathsf{P}} = \mathsf{NP}$, $\Pi_1^p = \mathsf{coNP}$, and $\Delta_1^p = \mathsf{P}$.

The class DP contains (decision) problems that are intersections of a problem in NP and a problem in coNP i. e., $\mathsf{DP} = \{L_1 \cap L_2 \mid L_1 \in \mathsf{NP}, L_2 \in \mathsf{coNP}\}$ (note that this is not the same as $\mathsf{NP} \cap \mathsf{coNP}$). The class

$\mathsf{FP}^{\mathsf{NP}}$ contains *functional* problems that can be solved in polynomial time with access to an NP oracle. We refer the readers to [63] for a detailed introduction to these classes.

Although we are dealing with a combination of probabilistic statements and argument graphs, the core language of epistemic constraints and its semantics is similar to Nilsson-style probabilistic logics [59, 45]. Thus, many properties regarding computational complexity of our proposal can be derived from the properties of these logics. However, there is one particular addition in our language as we allow a limited form of nesting logical expressions. For example, consider the constraint:

$$\varphi : (p(\mathtt{A} \wedge \mathtt{B}) \geq 0.7) \vee (p(\neg \mathtt{C} \vee \mathtt{B}) \leq 0.2)$$

Here, we can observe two levels of logical expression, one inside the (basic) epistemic atoms $p(\cdot)$ (i.e. the argument terms), and one connecting different atoms using logical connectives. In contrast, classical Nilsson-style probabilistic logics usually only consider as knowledge bases sets of valued epistemic atoms with $\leq, \geq, =$ as comparison operators[6] and not sets of general valued epistemic formulae. Hence, while $p(\mathtt{A}) = 0.5$ and $p(\mathtt{A}) + p(\mathtt{B} \wedge \neg \mathtt{C}) = 1$ fits this format, $p(\mathtt{A}) = 0.5 \vee p(\mathtt{B}) \geq 0.7$ does not. Allowing $\neq, <, >$ as comparison operators and considering general valued epistemic formulae makes our approach more expressive, as there are certain properties satisfied by Nilsson-style probabilistic logics and not necessarily by epistemic constraints. In order to show this, we need certain additional topological notions.

Definition 6.1. Let $\mathcal{P}$ be a set of belief distributions over $\mathsf{Nodes}(\mathcal{G})$. The **convex combination** of $P_1, P_2 \in \mathcal{P}$ w.r.t. $\delta \in [0,1]$ is the belief distribution $P^{\delta}_{P_1,P_2}$ defined as

$$P^{\delta}_{P_1,P_2}(\Gamma) = \delta P_1(\Gamma) + (1-\delta) P_2(\Gamma)$$

for all $\Gamma \subseteq \mathsf{Nodes}(\mathcal{G})$. A set of belief distributions $\mathcal{P}$ is called **convex** if for all $P_1, P_2 \in \mathcal{P}$ and $\delta \in [0,1]$, $P^{\delta}_{P_1,P_2} \in \mathcal{P}$.

Definition 6.2. Let $\mathcal{P}$ be a set of belief distributions over $\mathsf{Nodes}(\mathcal{G})$. A **path** π between two belief distributions $P_1, P_2 \in \mathcal{P}$ is a continuous function $\pi : [0,1] \to \mathcal{P}$ with $\pi(0) = P_1$ and $\pi(1) = P_2$. A set of belief distributions $\mathcal{P}$ is called **connected** if for all $P_1, P_2 \in \mathcal{P}$ there is a path π between P_1 and P_2 with $\pi(x) \in \mathcal{P}$ for all $x \in [0,1]$

Note that convexity of a set always implies its connectedness.

Example 70. Let $\mathsf{Nodes}(G) = \{\mathtt{A}\}$ and consider the sets $\mathcal{P}_1 = \{P \mid P(\mathtt{A}) \leq 0.4\}$, $\mathcal{P}_2 = \{P \mid P(\mathtt{A}) \geq 0.6\}$, and $\mathcal{P}_3 = \mathcal{P}_1 \cup \mathcal{P}_2$ of probability functions. First observe that $\mathcal{P}_1$ is a convex set: for any $\delta \in [0,1]$, $P_1, P_2 \in \mathcal{P}_1$ we have

$$P^{\delta}_{P_1,P_2}(\mathtt{A}) = \delta P_1(\mathtt{A}) + (1-\delta) P_2(\mathtt{A}) \leq \delta 0.4 + (1-\delta) 0.4 = 0.4$$

and therefore $P^{\delta}_{P_1,P_2} \in \mathcal{P}_1$. Similarly, $\mathcal{P}_2$ is convex. However, $\mathcal{P}_3$ is not convex, as for $P_1 \in \mathcal{P}_1 \subseteq \mathcal{P}_3$ with $P_1(\mathtt{A}) = 0.4$ and $P_2 \in \mathcal{P}_2 \subseteq \mathcal{P}_3$ with $P_2(\mathtt{A}) = 0.6$ we have that

$$P^{0.5}_{P_1,P_2}(\mathtt{A}) = 0.5 P_1(\mathtt{A}) + 0.5 P_2(\mathtt{A}) = 0.5 * 0.4 + 0.5 * 0.6 = 0.5$$

and therefore $P^{0.5}_{P_1,P_2} \notin \mathcal{P}_3$. In fact, $\mathcal{P}_3$ is not connected as every path π with $\pi(0) = P_1$ and $\pi(1) = P_2$ satisfies $\pi(x)(\mathtt{A}) = 0.5$ for *some* $x \in [0,1]$[7] and therefore is not entirely in $\mathcal{P}_3$.

The last property we consider concerns a given set of probability distributions being closed. For a sequence $P_1, P_2, \ldots$ with $P_i \in \mathcal{P}$ such that for all $\Gamma \subseteq \mathsf{Nodes}(\mathcal{G})$, $P^*(\Gamma) = \lim_{i \to \infty} P_i(\Gamma)$ exists, we abbreviate this by $P^* = \lim_{i \to \infty} P_i$.

Definition 6.3. A set of belief distributions $\mathcal{P}$ is called **closed** if for every sequence $P_1, P_2, \ldots$ with $P_i \in \mathcal{P}$ such that $P^* = \lim_{i \to \infty} P_i$ exists, $P^* \in \mathcal{P}$.

[6] More precisely, Nilsson-style probabilistic logics use linear probabilistic constraints, which are semantically equivalent to valued epistemic atoms using only $\leq, \geq, =$ as comparison operators.

[7] This is due to the intermediate value theorem.

Example 71. Let $\mathsf{Nodes}(G) = \{\mathtt{A}\}$ and consider the set $\mathcal{P} = \{P \mid P(\mathtt{A}) < 1\}$ of probability functions. Consider now the sequence P_i, $i > 1$ with

$$P_i(\varnothing) = 1/i \qquad\qquad P_i(\{\mathtt{A}\}) = 1 - 1/i$$

Observe that $P_i(\mathtt{A}) = 1 - 1/i$ and therefore $P_i \in \mathcal{P}$ for all $i > 1$. However $\lim_{i\to\infty} P_i = P$ exists and is defined as

$$P(\varnothing) = \lim_{i\to\infty} 1/i = 0 \qquad\qquad P(\{\mathtt{A}\}) = \lim_{i\to\infty} 1 - 1/i = 1$$

with $P(\mathtt{A}) = 1$ and therefore $P \notin \mathcal{P}$. It follows that $\mathcal{P}$ is not closed.

Classical Nilsson-style probabilistic logics, and therefore also the language consisting only of valued epistemic atoms with $\leq, \geq, =$ as operators, always yield closed and convex sets of satisfying distributions:

Theorem 6.4. *Let $\mathcal{C}$ be a set of valued epistemic atoms using only $\leq, \geq, =$ as comparison operators. Then $\mathsf{Sat}(\mathcal{C})$ is closed and convex.*

Adding $\neq, <, >$ as comparison operators and using general valued epistemic formulae allows us to create sets of constraints that do not need to obey these properties, thus endowing our system with additional expressivity.

Example 72. Consider the constraint $\varphi : p(\mathtt{A}) \geq 0.7 \vee p(\mathtt{A}) \leq 0.4$. We can show that the set $\mathsf{Sat}(\varphi)$ is closed but not convex. Therefore, $\mathsf{Sat}(\varphi)$ cannot be represented by constraints with valued epistemic atoms alone. Similarly, for the constraint $\psi : p(\mathtt{A}) > 0$ it can be seen that $\mathsf{Sat}(\psi)$ is convex but not closed and for $\phi : p(\mathtt{A}) \neq 0.5$ we have that $\mathsf{Sat}(\phi)$ is neither convex nor closed.

This relation between our proposal and Nilsson–style probabilistic logics allows us to state the *probabilistic constraint satisfiability problem* (PCSAT), which will serve as a basis for further results. This problem concerns verifying whether the set of constraints is consistent, i.e. if there exists at least one distribution satisfying it:

PCSAT	**Input**:	a set of epistemic constraints $\mathcal{C}$
	Output:	TRUE iff $\mathsf{Sat}(\mathcal{C}) \neq \varnothing$

Although our approach is more expressive than Nilsson–style probabilistic logics, our additions do not change the computational complexity of the involved reasoning tasks. Thus, the computational complexity of PCSAT can be derived from the fact that the general PSAT problem (probabilistic satisfiability) is NP-complete [40].

Theorem 6.5. *PCSAT is NP-complete.*

The problem remains intractable even when only considering (reasonable) restricted valued languages. The restricted probabilistic constraint satisfiability problem is stated as:

resPCSAT	**Input**:	a set of constraints $\mathcal{C}$ and a reasonable restricted value set Π
	Output:	TRUE iff $\mathsf{Sat}(\mathcal{C}, \Pi) \neq \varnothing$

For resPCSAT we have the following result.

Theorem 6.6. *resPCSAT is NP-complete.*

Although the PCSAT and resPCSAT problems tell us whether a set of constraints is satisfiable or not, they do not inform us about the values that arguments may take in the satisfying distributions. We therefore consider the following three decision and one functional problems. PCSAT-VAL and PCSAT-NVAL concern an argument being believed to a given degree by some or all satisfying distributions. PCSAT-INT tells us whether the collection of all values that an argument may take forms a particular interval, and the similar functional problem FPCSAT-INT provides us with this interval.

PCSAT-VAL	**Input**:	a set of constraints $\mathcal{C}$, an argument A, a value $x \in [0,1]$
	Output:	TRUE iff there is $P \in \mathsf{Sat}(\mathcal{C})$ with $P(\mathtt{A}) = x$.
PCSAT-NVAL	**Input**:	a set of constraints $\mathcal{C}$, an argument A, a value $x \in [0,1]$
	Output:	TRUE iff for every $P \in \mathsf{Sat}(\mathcal{C})$, $P(\mathtt{A}) = x$.
PCSAT-INT	**Input**:	a set of constraints $\mathcal{C}$, an argument A, values $u, l \in [0,1]$ s.t. $u \leq l$
	Output:	TRUE iff $[u,l] = \{x \mid P \in \mathsf{Sat}(\mathcal{C}), P(\mathtt{A}) = x\}$.
FPCSAT-INT	**Input**:	a set of constraints $\mathcal{C}$, an argument A
	Output:	values $u, l \in [0,1]$ with $u \leq l$ and $[u,l] = \{x \mid P \in \mathsf{Sat}(\mathcal{C}), P(\mathtt{A}) = x\}$.

From the observations concerning PCSAT and resPCSAT, the results pertaining to the PCSAT-VAL, PCSAT-INT and FPCSAT-INT problems follow directly, cf. [57, 53]. It also easy to adopt the existing results in order to analyze the complexity of PCSAT-NVAL.

Corollary 6.7. *PCSAT-VAL is NP-complete, PCSAT-INT is DP-complete, and FPCSAT-INT is $\mathsf{FP}^{\mathsf{NP}}$-complete.*

Theorem 6.8. *PCSAT-NVAL is coNP-complete.*

With these results at hand, we can move on to the analysis of epistemic graphs. From now on we assume that the graphs we are working with are finite, i.e. the sets of arguments and constraints are finite and the constraints are of finite length. In abstract argumentation, we typically consider the following computational complexity problems – verification, existence, skeptical and credulous acceptance. The verification problem refers to checking whether a given extension, labeling, or any other type of element, adheres to a given semantics. The existence problem concerns checking whether the output produced by a given semantics is not empty. Finally, credulous and skeptical concerns whether a given argument is accepted in some or all answers produced by a given semantics. We will now phrase these problems in terms of epistemic graphs and analyze their complexity. Let us start with the existence problem, for the epistemic semantics we have defined in Section 3.3 can be stated as follows:

EXIST-σ	**Input**:	an epistemic graph $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$, a semantics σ
	Output:	TRUE iff $\sigma(X) \neq \emptyset$

Verifying whether the output of the satisfaction semantics is not empty for a given epistemic graph is equivalent to checking whether there is at least one satisfying distribution for the constraints of this graph. Thus, in this case, the existence problem is the same as PCSAT. The results concerning acceptance, rejection, undecided and information minimizing or maximizing semantics depend on the semantics they are parameterized with (recall Definition 3.18). We can observe that due to the nature of these approaches, an appropriate satisfying distribution will exist as long as the underlying semantics possesses one. Thus, for example, the existence problem for the acceptance maximizing semantics under the satisfaction semantics is the same as for the satisfaction semantics itself, and is NP-complete. Our findings are summarized in Table 5, and given the fact that PCSAT and resPCSAT are of the same complexity, we can expect the tabulated results to hold also when a reasonable restricted value set is considered instead of the whole unit interval $[0,1]$.

Let us now consider the verification problem, which concerns whether a given probabilistic distribution adheres to a given semantics:

VER-σ	**Input**:	an epistemic graph $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$, a probability distribution $P \in \mathsf{Dist}(\mathcal{G})$, a semantics σ
	Output:	TRUE iff $P \in \sigma(X)$

In principle, the verification problem formulated in the above manner will be in P w.r.t. the size of the input. However, this is not due to the simplicity of the verification procedure, but due to the input size. In particular, we can observe that the description of a probability distribution is exponential in size w.r.t. the number of arguments. Consequently, for the purpose of this analysis, we will consider a different approach to verification, working with partial descriptions of probability distributions and epistemic graphs we will refer to as uni–argument:

Definition 6.9. An epistemic graph $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ is **uni–argument** iff for every constraint $\varphi \in \mathcal{C}$, every term $\alpha \in \mathsf{FTerms}(\varphi)$ is of the form $\mathtt{A}$, where $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$.

Definition 6.10. Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph and $P \in \mathsf{Dist}(\mathcal{G})$ a probability distribution. An **argument–based partial description** of P is a function $ParP : \mathsf{Nodes}(\mathcal{G}) \to [0,1]$ s.t. $ParP(\mathtt{A}) = P(\mathtt{A})$ for every $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$. With $\mathsf{Full}(ParP)$ we denote the set of of probability distribution from $\mathsf{Dist}(\mathcal{G})$ s.t. $ParP$ is their argument–based partial description.

Allowing only single argument terms to appear in the constraints is quite limiting and would not allow us to easily model some of the examples we have considered in this paper. However, we note that the majority of the translations from existing argumentation frameworks to epistemic graphs will in fact lead to such constraints, and thus this subclass is still interesting to analyze. For such graphs, it is easy to see that the presented partial description of a probability distribution is sufficient for verifying the majority of the semantics we have introduced.

UVER-σ	**Input**:	a uni–argument epistemic graph $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$, an argument–based partial description $ParP$ of a distribution $P \in \mathsf{Dist}(\mathcal{G})$, a semantics σ
	Output:	TRUE iff $\mathsf{Full}(ParP) \subseteq \sigma(X)$

With the exception of UVER-SAT, verifying that the distributions defined by a given partial description are acceptance (rejection, undecided, information) minimizing or maximizing corresponds to checking whether the existing set of constraints becomes inconsistent when extended with a new constraint describing a distribution that accepts (rejects, assigns undecided, rejects and accepts) further or fewer arguments than the one provided that input. Thus, the verification problems for the remaining semantics are complementary to PCSAT (resPCSAT) and can be shown to be in coNP or coNP-complete (see Table 5).

The probabilistic versions of credulous and skeptical acceptance ask whether an argument is believed in some or all probability distributions adhering to a given semantics. We can phrase these problems in the following manner:

CRED-σ	**Input**:	an epistemic graph $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$, a semantics σ and an argument $\mathtt{A}$
	Output:	TRUE iff $P(\mathtt{A}) > 0.5$ for some $P \in \sigma(X)$.
SKEPT-σ	**Input**:	an epistemic graph $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$, a semantics σ, and an argument $\mathtt{A}$
	Output:	TRUE iff $P(\mathtt{A}) > 0.5$ for all $P \in \sigma(X)$.

Verifying that an argument is believed (or assigned any particular status in general) by at least one satisfying distribution is not that different from adding a constraint stating that it has the desired status and checking whether it is satisfiable along with the existing constraints. Thus, the complexity of this problem can be derived from PCSAT. Furthermore, the complexity of credulous acceptance of satisfaction semantics also applies to semantics aiming to maximize acceptance or information. Unfortunately, for the minimizing semantics or semantics attempting to maximize other argument assignments, it appears that we go one level higher in polynomial hierarchy. Hence, the remaining problems are for now more likely to be in Σ_2^p rather than in NP. Verifying that an argument is believed in all distributions meeting a given semantics can be achieved by checking that there is no distribution under a given semantics in which this argument is not believed. With the exception of the satisfaction semantics, these problems appear to either in Π_2^p or Π_2^p-complete.

The summary of our findings can be seen in Table 5. Again, we can expect these statements to hold also when we consider restricted satisfiability according to a given reasonable restricted value set instead of the whole unit interval $[0,1]$. In a similar fashion we could provide results concerning skeptical or credulous rejection of an argument, or the argument being undecided in some or all distributions. For an argument being credulously skeptically assigned a particular value, we can build upon the PCSAT-VAL and PCSAT-NVAL, which basically address these problems for the satisfaction semantics.

Finally, we provide a preliminary result concerning default coverage (see Definition 5.13), which is erivable from the fact that PCSAT-INT is DP-complete. In particular, if PCSAT-INT returns FALSE for a given argument and the $[0,1]$ interval, then we can find a value that cannot be taken on by the argument and thus have default coverage:

	EXIST	UVER	CRED	SKEPT
SAT	NP-c	P	NP-c	coNP-c
SAT-AMAX	NP-c	coNP-c	NP-c	Π_2^p-c
SAT-AMIN	NP-c	in coNP	in Σ_2^p	in Π_2^p
SAT-RMAX	NP-c	coNP-c	in Σ_2^p	Π_2^p-c
SAT-RMIN	NP-c	in coNP	in Σ_2^p	in Π_2^p
SAT-UMAX	NP-c	in coNP	in Σ_2^p	in Π_2^p
SAT-UMIN	NP-c	coNP-c	Σ_2^p-c	Π_2^p-c
SAT-IMAX	NP-c	coNP-c	NP-c	Π_2^p-c
SAT-IMIN	NP-c	in coNP	in Σ_2^p	in Π_2^p

Table 5: Computational complexity of the existence, verification, credulous and skeptical acceptance problems for epistemic semantics. We use the following notation; SAT denotes the satisfaction semantics, SAT-·MAX and SAT-·MIN denote a given maximizing/minimizing semantics parameterized with satisfaction semantics, where $\cdot \in \{A, R, U, I\}$ stands for acceptance, rejection, undecided and information respectively.

DEFCOV **Input**: a consistent epistemic graph $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ and an argument A
Output: TRUE iff A is default covered in X.

Theorem 6.11. *DEFCOV is in coDP.*

In the future, we intend to provide further computational complexity results for the problems relating to belief minimizing and maximizing semantics. Furthermore, given that the epistemic semantics do not need to be universally defined, it would be interesting to study the problem of non–empty skeptical acceptance, i.e. the problem verifying that a given argument is accepted in all distributions adhering to a given semantics and that at least one such distribution exists. Finally, there is also a range of problems associated with verifying various forms of coverage and labeling consistency.

7 Comparison with the Literature

7.1 Comparison with Epistemic Postulates

The constraints in epistemic graphs quite naturally generalize the epistemic postulates [83, 46, 52, 71]. Given the fact that in the epistemic graphs we can decide whether a given property should hold for a particular argument or not, the desired postulate needs to be repeated for every element of the framework. Nevertheless, the general method is straightforward:

Example 73. Let us come back to Example 7 and consider the conflict–based framework $(\mathcal{G}, \mathcal{L})$ where $\mathcal{G} = (\{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}, \mathtt{E}\}, \{(\mathtt{A},\mathtt{B}), (\mathtt{C},\mathtt{B}), (\mathtt{C},\mathtt{D}), (\mathtt{D},\mathtt{C}), (\mathtt{D},\mathtt{E}), (\mathtt{E},\mathtt{E})\})$ and $\mathcal{L}$ assigns only the $-$ label to every edge, as depicted in Figure 7.

Let us assume that we want the rational postulate (i.e. if the attacker is believed, then the attacker is not believed) to hold in this framework. By considering this property for every argument (or, more accurately, for every attack) separately, we obtain the following set of constraints $\mathcal{C}_1$:

- $p(\mathtt{A}) > 0.5 \rightarrow p(\mathtt{B}) \leq 0.5$
- $p(\mathtt{C}) > 0.5 \rightarrow p(\mathtt{B}) \leq 0.5$
- $p(\mathtt{C}) > 0.5 \rightarrow p(\mathtt{D}) \leq 0.5$
- $p(\mathtt{D}) > 0.5 \rightarrow p(\mathtt{C}) \leq 0.5$
- $p(\mathtt{D}) > 0.5 \rightarrow p(\mathtt{E}) \leq 0.5$
- $p(\mathtt{E}) > 0.5 \rightarrow p(\mathtt{E}) \leq 0.5$

Let us now assume that we want the trusting postulate to hold, i.e. if all attackers of a given argument are disbelieved, then the argument is believed. This brings us to the following set of constraints $\mathcal{C}_2$:

- $p(\mathtt{A}) > 0.5$
- $(p(\mathtt{A}) < 0.5 \land p(\mathtt{C}) < 0.5) \to p(\mathtt{B}) > 0.5$
- $p(\mathtt{D}) < 0.5 \to p(\mathtt{C}) > 0.5$
- $p(\mathtt{C}) < 0.5 \to p(\mathtt{D}) > 0.5$
- $(p(\mathtt{E}) < 0.5 \land p(\mathtt{D}) < 0.5) \to p(\mathtt{E}) > 0.5$

7.2 Comparison with Constrained Argumentation Frameworks

Our proposal shares certain similarities with the constrained argumentation frameworks [31]. The purpose of this work was to allow a Dung's graph to be augmented with a single external constraint, representing certain restrictions that (for reasons unknown to the abstract system) were considered desirable by, for example, the user, and which were not necessarily reflected by the structure of the graph.

Definition 7.1. Let $PROP_S$ be a propositional language defined in the usual inductive way from a set S of propositional symbols, boolean constants $\top$, $\bot$ and the connectives $\neg, \land, \lor, \leftrightarrow$ and $\to$. A **constrained argumentation framework** is a tuple $(\mathcal{G}, \mathcal{L}, \mathsf{PC})$ where $(\mathcal{G}, \mathcal{L})$ is a labelled graph s.t. $\mathcal{L}$ assigns only $-$ to all edges, and PC is a propositional formula from $PROP_{\mathsf{Nodes}(\mathcal{G})}$.

A set of arguments Ext can be transformed into a two–valued interpretation (i.e. arguments contained in the set are mapped to $\mathbf{t}$ and those outside the set to $\mathbf{f}$) which can later be verified to be or not to be a model of PC. If it is, we say that Ext satisfies PC. A constrained admissible extension is then simply an admissible extension that satisfies PC. A similar idea holds for the stable extensions, i.e. conflict–free extensions that attack all other arguments. Finally, a constrained preferred extension is a maximal constrained admissible set.

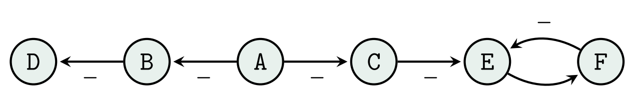


Figure 21: A conflict–based argument graph

Example 74 (Adapted from [31])**.** Let us consider the labelled graph from Figure 21 augmented with the constraint $\mathsf{PC} = \neg\mathtt{A} \lor \neg\mathtt{D} \lor \neg\mathtt{E}$. The admissible extensions of the graph (without the constraint) are $\varnothing$, $\{\mathtt{A}\}$, $\{\mathtt{A},\mathtt{D}\}$, $\{\mathtt{A},\mathtt{D},\mathtt{E}\}$, $\{\mathtt{A},\mathtt{E}\}$, $\{\mathtt{A},\mathtt{F}\}$ and $\{\mathtt{A},\mathtt{D},\mathtt{F}\}$. Once the constraint is applied, the set $\{\mathtt{A},\mathtt{D},\mathtt{E}\}$ has to be removed. The preferred extensions (i.e. maximal admissible extensions) of the graph are initially $\{\mathtt{A},\mathtt{D},\mathtt{E}\}$ and $\{\mathtt{A},\mathtt{D},\mathtt{F}\}$. However, if we take the constraint into account, we obtain the sets $\{\mathtt{A},\mathtt{E}\}$ and $\{\mathtt{A},\mathtt{D},\mathtt{F}\}$. The stable extensions of the labelled graph are the same as the preferred ones, however, in the constrained graph only $\{\mathtt{A},\mathtt{D},\mathtt{F}\}$ meets the requirements.

Let us now show how the constrained framework can be handled by the epistemic graphs. The epistemic postulates can be used to retrieve the labeling–based semantics for Dung's graphs [71], where believed arguments are mapped to in, disbelieved to out, and neither to und. Later, from those labelings we can extract the extensions by focusing on the arguments marked as in [9]. Moreover, in Section 7.1 we have observed that postulates can be straightforwardly generalized by the epistemic graphs. Thus, by reusing the previously developed methods and transforming PC into an epistemic constraint, we can retrieve the desired extensions.

Example 75. Let us continue Example 74. By using the method from [71], we obtain the following set $\mathcal{C}_1$ that is associated with the admissible semantics:

- $p(\mathtt{A}) \geq 0.5$
- $p(\mathtt{B}) < 0.5 \rightarrow p(\mathtt{A}) > 0.5$
- $p(\mathtt{C}) < 0.5 \rightarrow p(\mathtt{A}) > 0.5$
- $p(\mathtt{D}) < 0.5 \rightarrow p(\mathtt{B}) > 0.5$
- $p(\mathtt{E}) < 0.5 \rightarrow p(\mathtt{C}) > 0.5 \vee p(\mathtt{F}) > 0.5$
- $p(\mathtt{F}) < 0.5 \rightarrow p(\mathtt{E}) > 0.5$
- $p(\mathtt{B}) > 0.5 \rightarrow p(\mathtt{A}) < 0.5$
- $p(\mathtt{C}) > 0.5 \rightarrow p(\mathtt{A}) < 0.5$
- $p(\mathtt{D}) > 0.5 \rightarrow p(\mathtt{B}) < 0.5$
- $p(\mathtt{E}) > 0.5 \rightarrow p(\mathtt{C}) < 0.5 \wedge p(\mathtt{F}) < 0.5$
- $p(\mathtt{F}) > 0.5 \rightarrow p(\mathtt{E}) < 0.5$

In Table 6 we have listed all the ternary distributions satisfying $\mathcal{C}_1$. It is easy to see that the sets of believed arguments obtained from these distributions coincide with the admissible extensions of our labelled graph. The epistemic representation of the PC constraint is $p(\mathtt{A}) \leq 0.5 \vee p(\mathtt{D}) \leq 0.5 \vee p(\mathtt{E}) \leq 0.5$. By adding it to the set $\mathcal{C}_1$, we obtain the constraint set $\mathcal{C}_1'$, which excludes P_{14} and thus the only distribution producing the epistemic extension $\{\mathtt{A}, \mathtt{D}, \mathtt{E}\}$. By enforcing information maximality along with the $\mathcal{C}_1$ and $\mathcal{C}_1'$ constraints, we obtain either distributions P_{14} and P_{16} or P_9 and P_{16}, which are associated with the desired preferred extensions. By replacing information maximality with the non–neutral restriction, we get P_{14} and P_{16} for $\mathcal{C}_1$ and only P_{16} for $\mathcal{C}_1'$. Thus, the stable extensions are retrieved as intended as well.

	$P(\mathtt{A})$	$P(\mathtt{B})$	$P(\mathtt{C})$	$P(\mathtt{D})$	$P(\mathtt{E})$	$P(\mathtt{F})$	$\mathcal{C}_1$	$\mathcal{C}_1'$	$\mathcal{C}_1^{InfMax}$	$\mathcal{C'}_1^{InfMax}$	$\mathcal{C}_1^{Bin}$	$\mathcal{C'}_1^{Bin}$
P_1	0.5	0.5	0.5	0.5	0.5	0.5	✓	✓	×	×	×	×
P_2	1	0.5	0.5	0.5	0.5	0.5	✓	✓	×	×	×	×
P_3	1	0	0.5	0.5	0.5	0.5	✓	✓	×	×	×	×
P_4	1	0.5	0	0.5	0.5	0.5	✓	✓	×	×	×	×
P_5	1	0	0	0.5	0.5	0.5	✓	✓	×	×	×	×
P_6	1	0	0.5	1	0.5	0.5	✓	✓	×	×	×	×
P_7	1	0	0	1	0.5	0.5	✓	✓	×	×	×	×
P_8	1	0.5	0	0.5	1	0	✓	✓	×	×	×	×
P_9	1	0	0	0.5	1	0	✓	✓	×	✓	×	×
P_{10}	1	0.5	0.5	0.5	0	1	✓	✓	×	×	×	×
P_{11}	1	0	0.5	0.5	0	1	✓	✓	×	×	×	×
P_{12}	1	0.5	0	0.5	0	1	✓	✓	×	×	×	×
P_{13}	1	0	0	0.5	0	1	✓	✓	×	×	×	×
P_{14}	1	0	0	1	1	0	✓	×	✓	×	✓	×
P_{15}	1	0	0.5	1	0	1	✓	✓	×	×	×	×
P_{16}	1	0	0	1	0	1	✓	✓	✓	✓	✓	✓

Table 6: Ternary distributions associated with the set of constraints $\mathcal{C}_1$ from Example 75.

7.3 Comparison with Encodings of Dung's Semantics

For both practical and research purposes, the classical extension–based Dung's semantics have been encoded into other logics, including the propositional logic [11, 12]. In this approach, the extensions of a given graph under a given semantics are associated with the models of a particular formula. For example, the conflict–free extensions of a graph $(\mathcal{G}, \mathcal{L})$ with only the $-$ labels can be obtained from:

$$\Phi_1 = \bigwedge_{(\mathtt{A},\mathtt{B})\in\mathcal{L}^-(\mathcal{G})} (\neg\mathtt{A} \vee \neg\mathtt{B})$$

If we interpret $\neg\mathtt{A}$ and $\neg\mathtt{B}$ as $p(\mathtt{A}) \leq 0.5$ and $p(\mathtt{B}) \leq 0.5$, we can observe that what we obtain is the rational postulate. The arguments that are believed in the distributions satisfying this postulate coincide with the conflict–free extensions as noted in [71]. Consequently, the epistemic constraints can be seen as a probabilistic generalization of the [11] encodings. Although the epistemic formulae associated with a given semantics may be less concise than the propositional ones due to the fact that they are broken into smaller properties expressed by the postulates and are aimed at labelings rather than extensions, there are natural connections between the two.

7.4 Comparison with Weighted and Ranking–Based Semantics

In the recent years, argumentation semantics have received considerable attention, in particular those producing weightings or rankings on arguments rather than extensions or labelings [23, 24, 55, 1, 5, 3, 2, 76, 17, 4, 6]. We can find works considering applicable to frameworks using only the attack relation, only the support relation, both of them, those that allow arguments to have initial weights as well as those that do not. Given certain structural similarities between these approaches and epistemic semantics, it is natural to compare them. Although in both cases what we receive can be seen as "assigning numbers from $[0,1]$" to arguments (either as side or end product), probabilities in the epistemic approach are interpreted as belief, while weights remain abstract and open to a number of possible instantiations. Thus, while they can tell us whether an argument is more or less acceptable than another argument, they do not state whether it is acceptable at all. As a result, many of the postulates set out in the weighted and ranking–based methods are, by design, counter–intuitive in the epistemic approach, even though they can be perfectly applicable in other scenarios.

Another major difference between the epistemic graphs and the weighted or ranking semantics is that in the latter, the patterns set out by the semantics have to be global, which leads to side effects not desirable in the epistemic approach. In particular, two arguments supported and attacked by the same sets of arguments will be assigned the same value (assuming their initial weights are similar, if applicable). Furthermore, it is the value assigned to a given argument that plays a role, not the argument itself. Apart from the loss of context–sensitivity, it also has the side effect of forcing a single interpretation of support or attack throughout the framework. Let us consider an example where a student considers applying to two universities U_1 and U_2. Having good grades GG increases the chances of that student being admitted. Having sport achievements SA increases the chances of getting to U_1 and U_2, but in case of U_2 it is also a necessary condition. The way SA supports U_1 and U_2 is thus different and a low score on SA should affect the scores of U_1 and U_2 differently. Nevertheless, weighted and ranking semantics will produce the same values for U_1 and U_2.

Another property of the weighted and ranking semantics (and that is not enforced in the epistemic approach) is that given the values of the parents, a single value of the target is returned. This may be a restriction if we want the flexibility to express a margin of error or vagueness. Depending on how the constraints are defined in the epistemic approach, we can force the target to take on a single probability as well as allow it any value from a given range. Consequently, we have a certain form of control over specificity in the epistemic graphs. A more relaxed approach can be useful in modelling imperfect agents or incomplete situations, and such tasks can pose certain difficulties to the recalled weighted semantics.

Finally, we would like to note that in the ranking–based semantics, it is often assumed that arguments not connected in the graph should not affect each other (i.e. the so–called independence axiom). As exemplified in Section 7.2, the epistemic graphs are allowed to be affected by external knowledge, which is not allowed in the weighted approaches.

In conclusion, we can observe that despite certain high–level similarities, the weighted and epistemic approaches do not have that much in common. Although one can argue that it is possible to represent certain weighting functions as constraints and the other way around, particularly if multiplication or division were allowed in the latter, we would either obtain constraints that violate the meaning of epistemic probabilities or semantics that do not conform to the required axioms.

7.5 Comparison with Abstract Dialectical Frameworks

Epistemic graphs share certain similarities with abstract dialectical frameworks (ADFs) [20, 80, 81, 18, 65, 66, 82, 67]. Before we compare the two approaches, we briefly review ADFs and some of their semantics.

Definition 7.2. An **abstract dialectical framework** (ADF) is a tuple $(\mathcal{G}, \mathcal{L}, \mathcal{AC})$, where $(\mathcal{G}, \mathcal{L})$ is a labelled graph and $\mathcal{AC} = \{\mathcal{AC}_{\mathtt{A}}\}_{\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})}$ is a set of **acceptance conditions**. An acceptance condition is a total function $\mathcal{AC}_{\mathtt{A}} : 2^{\mathsf{Parent}(\mathtt{A})} \to \{in, out\}$. An arc $(\mathtt{A}, \mathtt{B}) \in \mathsf{Arcs}(\mathcal{G})$ is:

- **supporting** iff for no $R \subseteq \mathsf{Parent}(\mathtt{B})$ we have that $\mathcal{AC}_{\mathtt{B}}(R) = in$ and $\mathcal{AC}_{\mathtt{B}}(R \cup \{\mathtt{A}\}) = out$.
- **attacking** iff for no $R \subseteq \mathsf{Parent}(\mathtt{B})$ we have that $\mathcal{AC}_{\mathtt{B}}(R) = out$ and $\mathcal{AC}_{\mathtt{B}}(R \cup \{\mathtt{A}\}) = in$.
- **dependent** iff it is neither supporting nor attacking.

In the labeling–based semantics for ADFs, we use three–valued interpretations which assigns truth values $\{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$ to arguments that are compared according to precision (information) ordering: $\mathbf{u} \leq_i \mathbf{t}$ and $\mathbf{u} \leq_i \mathbf{f}$. The pair $(\{\mathbf{t}, \mathbf{f}, \mathbf{u}\}, \leq_i)$ forms a complete meet–semilattice with the meet operation $\sqcap$ assigning values in the following way: $\mathbf{t} \sqcap \mathbf{t} = \mathbf{t}$, $\mathbf{f} \sqcap \mathbf{f} = \mathbf{f}$ and $\mathbf{u}$ in all other cases. These notions can be easily extended to interpretations. For two interpretations v and v' on $\mathsf{Nodes}(\mathcal{G})$, $v \leq_i v'$ iff for every argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, $v(\mathtt{A}) \leq_i v'(\mathtt{A})$. In the case v is three and v' two–valued (i.e. contains no $\mathbf{u}$ mappings), we say that v' extends v[8]. The set of all two–valued interpretations extending v is denoted $[v]_2$. The meet of two interpretations $v \sqcap v'$ is an interpretation v'' s.t. $v''(\mathtt{A}) = v(\mathtt{A}) \sqcap v'(\mathtt{A})$ for $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$. We use v^x to denote a set of arguments mapped to x by v, where $x \in \{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$. Given an acceptance condition $\mathcal{AC}_{\mathtt{A}}$ for an argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ and an interpretation v, we define a shorthand $v(\mathcal{AC}_{\mathtt{A}})$ as $\mathcal{AC}_{\mathtt{A}}(v^{\mathbf{t}} \cap \mathsf{Parent}(\mathtt{A}))$[9].

Definition 7.3. Let $D = (\mathcal{G}, \mathcal{L}, \mathcal{AC})$ be an ADF, $\mathcal{V}$ the set of all three–valued interpretations defined on $\mathsf{Nodes}(\mathcal{G})$, $\mathtt{A}$ an argument in $\mathsf{Nodes}(\mathcal{G})$ and v an interpretation in $\mathcal{V}$. The **three–valued characteristic operator** of D is a function s.t. $\Gamma(v) = v'$ where $v'(a) = \bigsqcap_{w \in [v]_2} w(\mathcal{AC}_a)$. An interpretation v is:

- a **three–valued model** iff for all $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$, $v(\mathtt{A}) \neq \mathbf{u}$ implies that $v(\mathtt{A}) = v(\mathcal{AC}_{\mathtt{A}})$.
- an **admissible** labeling iff $v \leq_i \Gamma(v)$.
- a **complete** labeling iff $v = \Gamma(v)$.
- a **preferred** labeling iff it is $\leq_i$–maximal admissible labeling.
- a **grounded** labeling iff it is the least fixpoint of Γ.

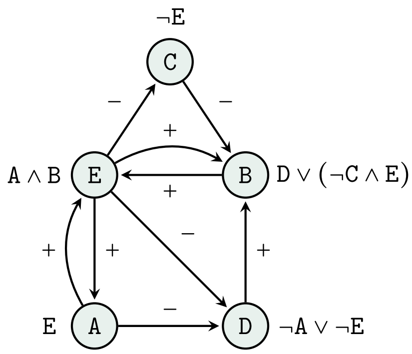
Example 76. The admissible, complete, preferred and grounded labelings of the ADF depicted in Figure 22a are visible in Table 22b.

So ADFs allow for the acceptance of an argument to be determined as the Boolean combination of the acceptance of its parents. Having an acceptance condition for each node is similar in spirit to having constraints for epistemic graphs. Both structures can handle relations that are positive, negative, or neither. However, there are some fundamental differences between ADFs and epistemic graphs.

The acceptance conditions can tell us whether an argument is accepted or rejected based on the acceptance of its parents. In contrast, the epistemic constraints can produce probability assignments in the unit interval that depend on the degrees of belief in other arguments, which offers a much more fine–grained perspective. It also allows epistemic graph to easily handle some forms of support, such as the abstract or deductive supports, which are normally too weak to be expressed in ADFs or require certain translations [67, 68]. The constraints also allow us to define a range of values that an argument may take on in given circumstances as well as a single particular value, and thus offers more flexibility in modelling the acceptability of an argument. Furthermore, in epistemic graphs the constraints are assigned per graph, not per

[8]This means that the elements mapped originally to $\mathbf{u}$ are now assigned either $\mathbf{t}$ or $\mathbf{f}$.

[9]By the abuse of notation, in this case we treat in as synonymous with $\mathbf{t}$ and out with $\mathbf{f}$



(a) Example of an ADF

	A	B	C	D	E	ADM	CMP	PREF	GRD
v_1	**u**	**u**	**u**	**u**	**u**	✓	✓	×	✓
v_2	**f**	**u**	**u**	**u**	**f**	✓	×	×	×
v_3	**f**	**u**	**t**	**u**	**f**	✓	×	×	×
v_4	**f**	**u**	**u**	**t**	**f**	✓	×	×	×
v_5	**f**	**u**	**t**	**t**	**f**	✓	×	×	×
v_6	**f**	**t**	**u**	**t**	**f**	✓	×	×	×
v_7	**t**	**t**	**f**	**u**	**t**	✓	×	×	×
v_8	**t**	**t**	**f**	**f**	**t**	✓	✓	✓	×
v_9	**f**	**t**	**t**	**t**	**f**	✓	✓	✓	×

(b) Labelings of the presented ADF

Figure 22: An example of an ADF and its admissible (ADM), complete (CMP), preferred (PREF) and grounded (GRD) labelings.

argument, and we can handle situations where the belief in one argument might depend not just on its parents, but also on other arguments for reasons known only to the agent. Although links in ADFs can be seen as optional and derivable from the conditions, the arguments present in the condition of a given argument are understood as its parents. Therefore, a relation between them is forced in the graph, independently of whether it is consistent with the way the graph was instantiated or not. Additionally, the completeness of acceptance conditions is obligatory in ADFs, while the completeness of epistemic constraints is optional and requiring it should be motivated by a given application. This control may be useful in user modelling, where we are not yet sure how a given argument and its associated relations are perceived by the user.

These differences show that epistemic graphs are quite distinct from ADFs. Nevertheless, it is possible for epistemic graphs to model ADFs. We will show how this can be achieved based on an example.

Example 77. Let us come back to the ADF from Example 76 and Figure 22a. We will now show how acceptance conditions can be transformed into constraints s.t. the labelings extracted from the probabilistic distributions correspond to the ADF labelings under a given semantics.

Let us start with the admissible semantics and focus on argument E. If we were to create a truth table for its condition A ∧ B, we would observe that if E is to be accepted, then A and B have to be true, and if E is to be rejected, then A or B has to be false. This rather straightforwardly translates to the following constraints. If argument E is believed, then A and B have to be believed. If argument E is disbelieved, then either A or B has to be disbelieved. By performing a similar analysis for the remaining arguments, we obtain the following collection $\mathcal{C}_1$ of constraints:

- $p(\mathtt{A}) > 0.5 \to p(\mathtt{E}) > 0.5$, $p(\mathtt{A}) < 0.5 \to p(\mathtt{E}) < 0.5$
- $p(\mathtt{B}) > 0.5 \to p(\mathtt{D}) > 0.5 \vee (p(\mathtt{C}) < 0.5 \wedge p(\mathtt{E}) > 0.5)$, $p(\mathtt{B}) < 0.5 \to p(\mathtt{D}) < 0.5 \wedge (p(\mathtt{C}) > 0.5 \vee p(\mathtt{E}) < 0.5)$
- $p(\mathtt{C}) > 0.5 \to p(\mathtt{E}) < 0.5$, $p(\mathtt{C}) < 0.5 \to p(\mathtt{E}) > 0.5$
- $p(\mathtt{D}) > 0.5 \to p(\mathtt{A}) < 0.5 \vee p(\mathtt{E}) < 0.5$, $p(\mathtt{D}) < 0.5 \to p(\mathtt{A}) > 0.5 \wedge p(\mathtt{E}) > 0.5$
- $p(\mathtt{E}) > 0.5 \to p(\mathtt{A}) > 0.5 \wedge p(\mathtt{B}) > 0.5$, $p(\mathtt{E}) < 0.5 \to p(\mathtt{A}) < 0.5 \vee p(\mathtt{B}) < 0.5$

What is therefore happening is that for every (propositional) acceptance condition we create two constraints that are the epistemic adaptations of the formulas $\mathsf{X} \to \mathcal{AC}_{\mathsf{X}}$ and $\neg\mathsf{X} \to \neg\mathcal{AC}_{\mathsf{X}}$, where, assuming that the consequent is in a form without nested negations, a positive literal Z is transformed into an epistemic atom $p(\mathsf{Z}) > 0.5$ and a negative literal ¬Z becomes $p(\mathsf{Z}) < 0.5$. The list of all and only ternary distributions satisfying the $\mathcal{C}_1$ set of constraints is visible in Table 7. It is easy to verify that the epistemic labelings (i.e. labelings obtained by marking believed arguments as $\mathbf{t}$, disbelieved as $\mathbf{f}$, and neither as $\mathbf{u}$) corresponding to them are exactly the admissible labelings of our ADF.

Modifying the aforementioned list of constraints in order to obtain the distributions corresponding to complete labelings is quite straightforward and basically boils down to replacing implication with an iff in the rules of $\mathcal{C}_1$. The core difference between the admissible and complete ADF labelings is that while in both cases accepting an argument implied satisfying its acceptance condition, in the admissible case having a satisfied condition did not necessarily mean the argument is accepted. There was more freedom in the choice of the undecided assignment. By removing this, we obtain the set of constraints $\mathcal{C}_2$, and as seen in Table 7, it selects the distributions whose epistemic labelings are complete in our ADF:

- $p(\mathtt{A}) > 0.5 \leftrightarrow p(\mathtt{E}) > 0.5, p(\mathtt{A}) < 0.5 \leftrightarrow p(\mathtt{E}) < 0.5$
- $p(\mathtt{B}) > 0.5 \leftrightarrow p(\mathtt{D}) > 0.5 \vee (p(\mathtt{C}) < 0.5 \wedge p(\mathtt{E}) > 0.5), p(\mathtt{B}) < 0.5 \leftrightarrow p(\mathtt{D}) < 0.5 \wedge (p(\mathtt{C}) > 0.5 \vee p(\mathtt{E}) < 0.5)$
- $p(\mathtt{C}) > 0.5 \leftrightarrow p(\mathtt{E}) < 0.5, p(\mathtt{C}) < 0.5 \leftrightarrow p(\mathtt{E}) > 0.5$
- $p(\mathtt{D}) > 0.5 \leftrightarrow p(\mathtt{A}) < 0.5 \vee p(\mathtt{E}) < 0.5, p(\mathtt{D}) < 0.5 \leftrightarrow p(\mathtt{A}) > 0.5 \wedge p(\mathtt{E}) > 0.5$
- $p(\mathtt{E}) > 0.5 \leftrightarrow p(\mathtt{A}) > 0.5 \wedge p(\mathtt{B}) > 0.5, p(\mathtt{E}) < 0.5 \leftrightarrow p(\mathtt{A}) < 0.5 \vee p(\mathtt{B}) < 0.5$

By applying, for example, information maximality to either $\mathcal{C}_1$ or $\mathcal{C}_2$ we can retrieve the preferred labelings of our ADF. Given that the grounded labeling is the least informative complete one, applying information minimality to $\mathcal{C}_2$ gives us the desired answer.

	$P(\mathtt{A})$	$P(\mathtt{B})$	$P(\mathtt{C})$	$P(\mathtt{D})$	$P(\mathtt{E})$	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_1^{InfMax}$	$\mathcal{C}_2^{InfMin}$
P_1	0.5	0.5	0.5	0.5	0.5	✓	✓	×	✓
P_2	0	0.5	0.5	0.5	0	✓	×	×	×
P_3	0	0.5	1	0.5	0	✓	×	×	×
P_4	0	0.5	0.5	1	0	✓	×	×	×
P_5	0	0.5	1	1	0	✓	×	×	×
P_6	0	1	0.5	1	0	✓	×	×	×
P_7	1	1	0	0.5	1	✓	×	×	×
P_8	1	1	0	0	1	✓	✓	✓	×
P_9	0	1	1	1	0	✓	✓	✓	×

Table 7: Ternary satisfying distributions for the epistemic graph from Example 77. $InfMax$ and $InfMin$ stand for information maximal and minimal semantics respectively.

This example shows us that it is possible for the epistemic graphs to handle ADFs under the labeling–based semantics, even though providing a full translation for any type of condition may be more involved than the presented approach. Given the fact that ADFs can subsume a number of different frameworks [68], it is also possible for the epistemic graphs to express many more approaches to argumentation than we recall here.

There are certain generalizations of ADFs that are relevant in the context of our work. In [69], a probabilistic version of ADFs has been introduced. However, this work follows the constellation interpretation of a probability, not the epistemic one, which leads to significantly different modelling [46, 72]. In a recent work [41], a new version of weighted ADFs has been proposed, in which conditions no longer map subsets of parents of a given argument to in or out, but take values assigned to the parents and state a specific value that should be assigned to the target. These values can be abstract entities with some form of ordering between them as well as numbers from the $[0, 1]$ interval. The information ordering present in the original ADFs is then adopted accordingly and then the definition of the existing operator–based semantics (admissible, grounded, preferred, complete) remains unchanged.

Despite certain possible overlaps, weighted ADFs are incomparable to epistemic graphs. On one hand, similarly as in original ADFs, condition completeness and limiting the conditions to depend only on the parents of a given argument is enforced. Furthermore, unlike epistemic constraints, weighted acceptance conditions are very specific in the sense that a given combination of values assigned to a given argument leads to a precise, defined outcome. Therefore, even the rational epistemic postulate (i.e. if the attacker is believed, then the attackee is not believed) defined for the Dung's framework cannot be expressed in weighted ADFs, as a particular value assigned to the attacker is matched with a range of possible values of the attackee, not one specific outcome. Consequently, there are properties expressible with epistemic graphs, but not with weighted ADFs. On the other hand, weighted ADFs are not specialized for handling probabilities, and therefore can take as input further unspecified values, not only numbers. Thus, we can construct scenarios handled by weighted ADFs, but not by epistemic graphs. Additionally, even if values from the $[0, 1]$ interval are considered, for computational reasons they are amended with a special element indicating that a given value is undefined and the interpretation of this element is different than the one of neither agreeing nor disagreeing in the epistemic proposal.

7.6 Comparison with Bayesian Networks

A Bayesian network (or a causal probabilistic network) is an acyclic directed graph in which each node denotes a random variable (a variable that can be instantiated with an element from some set of events) and each arc denotes causal influence of one random variable on another [64, 8]. Random variables can be used to represent propositions that are either "true" or "false". For example, if the random variable is `car-battery-is-flat`, then it can be instantiated with the event `car-battery-is-flat`, or the event $\neg$`car-battery-is-flat`.

A key advantage of a Bayesian network is the use of independence assumptions that can be derived from the graph structure. These independence assumptions allow for the joint probability distribution for the random variables in a graph to be decomposed into a number of smaller joint probability distributions. This makes the acquisition and use of probabilistic information much more efficient.

Superficially, there are some similarities between Bayesian networks and epistemic graphs. Both have a graphical representation of the influence of one node on another where the nodes can be used to represent statements. Furthermore, the influences by one node on another can change the belief in the target node, and this change can be either positive or negative.

However, Bayesian networks and epistemic graphs are significantly different in their underlying representation and in the way they work as we clarify here:

1. A Bayesian network is used with a single probability distribution whereas the constraints associated with an epistemic graph allow for multiple probability distributions that satisfy the constraints;

2. A Bayesian network updates a random variable by taking on a specific instantiation, and that there is no longer any doubt about that instantiation (e.g. in the case of a random variable being updated by taking on the value "true", then there is no longer any doubt or uncertainty about the value of the random variable being "true"), whereas with epistemic graphs, if the belief in a node is updated, it can be of any value in the unit interval (e.g. for an argument A, that is current believed to degree 0.7, we may choose to update it to degree 0.3), and so this means epistemic graphs can reflect uncertainty in updating; and

3. A Bayesian network propagates updates by conditioning, which is a specific kind of constraint (e.g. for a graph with two variables α and β, after updating α, the propagated belief in β is $P(\beta \mid \alpha)$), whereas the framework for epistemic graphs provides a rich language for specifying a wide variety of constraints between the two variables.

The motivations for Bayesian networks and epistemic graphs are also different. Bayesian networks are for modelling normative reasoning (i.e. they model how we should reason with a set of random variables with given set of influences between them). In contrast, epistemic graphs are intended to reflect how people may choose to reason with the uncertainty concerning arguments. So with epistemic graphs, we may model how some people may regard the relative belief in a set of arguments, but it does not mean that they are correct in any normative sense, rather it is just a way of modelling their perspective or behaviour.

8 Discussion

In this paper, we have generalized the epistemic approach to probabilistic argumentation by introducing the notion of epistemic graphs which define how arguments influence each other through the use of epistemic constraints. We showed how epistemic graphs can be used to model a real–life scenario and provided an extensive study of graphs' properties. We have also created a proof theory for reasoning with the constraints that is both sound and complete, analyzed various ways in which the constraints can affect arguments and relations between them, and provided a study concerning the computational complexity of various aspects of the epistemic graphs. We have also compared our research to other relevant works in argumentation and Bayesian networks. Our proposal meets the requirements postulated in the introduction:

Modelling fine–grained acceptability Epistemic graphs can express varying degrees of belief we have in arguments and these beliefs can be harnessed and restricted through the use of epistemic constraints, as seen in Section 3. The beliefs can be easily associated with the traditional notions of acceptability [71] and, in contrast to more abstract forms of scoring and ranking arguments, provide a clearer meaning of the values associated with arguments.

Modelling positive and negative relations between arguments With epistemic graphs, we can model various types of relations between arguments, including positive, negative or mixed, as exemplified in Section 3.2 and further studied in Section 5.3. Furthermore, they can also handle relations marked as group or binary (for example, two attackers need to be believed in order for the target to be disbelieved versus at least one attacker needs to be believed for the target to be disbelieved), as well as interactions that could be marked as recursive (see Example 19). Finally, in our analysis of the nature of various relations, we have also discussed how the views on the influence one argument has over another change depending on whether local or global perspective is taken into account.

Modelling context–sensitivity Two structurally similar graphs can be assigned different sets of epistemic constraints. An agent is allowed to have different opinions on similar graphs and adopt them according to his or her needs, be it caused by the actual content of the arguments, agent's preferences or knowledge, and the way an agent understands the arguments. Thus, there is no requirement for the same graphs being evaluated in the same fashion under the same epistemic semantics. For example, we can easily create two different sets of constraints for the two scenarios considered in Example 3 that represent the agent's opinions. Epistemic graphs can also deal with restrictions that are not necessarily reflected in the structure of the graph. For example, one can define constraints concerning unrelated arguments, as discussed in Section 7.2.

Modelling different perspectives Agents do not need to adhere to a uniform perspective on a given problem. They can perceive arguments and relations between them differently, and thus find different arguments believable or not, as seen in Examples 1, 4 and 18. Furthermore, even arguments sharing some similarities in their views can respond differently when put in the same situation. Such behaviour could have been observed in Example 2, and it would not be problematic to create constraints that handle rejecting certain arguments differently.

Modelling imperfect agents The freedom in defining constraints and beliefs in arguments allows agents to express their views freely, independently of whether they are deemed rational or not or are strongly affected by cognitive biases. For example, two logically conflicting arguments do not need to be accompanied by constraints reflecting this conflict. Furthermore, agents do not necessarily need to adhere to various types of semantics [71], and epistemic constraints could be used to grasp their views more accurately.

Modelling incomplete graphs An argument graph might not reflect all the knowledge an agent has and that is relevant to a given problem, as seen in Example 2. Consequently, a given argument can be believed or disbelieved without any apparent justification, as seen in Section 3.4. It is however not difficult to create constraints stating that a given argument should be assigned a particular score. It is also possible to not create any constraints at all if it is not known how an agent views the interactions between arguments, and thus provide no coverage to arguments or relations, as seen in Section 5.2.

Although our analysis of epistemic graphs is extensive, there are still various topics to be considered. The currently proposed epistemic graph semantics can be further refined in order to take the additional information contained in the structure of the graph, but not in the constraints, into account. We could, for example, consider a coverage–based family of semantics, where the status assigned to a given argument can depend on the level of coverage it possesses. The presented results concerning the computational issues of our formalisms also need to be extended in order to account for certain hardness results, the problems relating to belief minimizing and maximizing semantics, the non–empty skeptical acceptance, as well as the problems associated with verifying various forms of coverage and labeling consistency.

Another issue we want to explore concerns how the constraints can be obtained. Crowd–sourcing opinions on arguments is a popular method for obtaining data [26, 70, 71]. Such data concerning beliefs in arguments and whether arguments are seen as related could be analyzed with, for example, machine learning techniques, in order to construct appropriate constraints.

In the future we would like to explore the use of epistemic graphs for practical applications, in particular for computational persuasion. Applying the existing epistemic approach to modelling persuadee's beliefs in arguments has produced methods for updating beliefs during a dialogue [47, 49, 51], efficient representation and reasoning with the probabilistic user model [42], modelling uncertainty in belief distributions [50], for learning belief distributions [44], and harnessing decision rules for optimizing the choice of arguments based on the user model [43]. These methods can be further developed in the context of epistemic graphs in order to provide a well understood theoretical and computationally viable framework for applications such as behaviour change.

The epistemic approach is not the only form of probabilistic argumentation. Another popular method relies on constellation probabilities [56, 46, 37] in which we can consider a number of argument graphs, each one having a probability of being the "real graph". Incorporating constellation probabilities in epistemic graphs would, for example, allow for a more refined handling of agents whose argument graphs are not complete but have a chance of containing certain arguments. Furthermore, it is also possible to allow epistemic constraints to express beliefs in arguments as well as in the relations between them, similarly as done in [72]. Consequently, further developments of the epistemic graphs are an interesting topic for future work.

Finally, we will also investigate algorithms and implementations aimed at handling epistemic graphs. This can be done through devising dedicated solutions as well as by introducing appropriate translations to, for example, propositional logic, as indicated by the results in Sections 4 and 6. Further possibilities concern employing SMT solvers or constraint logic programming.

References

[1] L. Amgoud and J. Ben-Naim. Ranking-based semantics for argumentation frameworks. In W. Liu, V. S. Subrahmanian, and J. Wijsen, editors, *Proceedings of SUM'13*, pages 134–147. Springer, 2013.

[2] L. Amgoud and J. Ben-Naim. Axiomatic foundations of acceptability semantics. In C. Baral, J. Delgrande, and F. Wolter, editors, *Proceedings of KR'16*, pages 2–11. AAAI Press, 2016.

[3] L. Amgoud and J. Ben-Naim. Evaluation of arguments from support relations: Axioms and semantics. In S. Kambhampati, editor, *Proceedings of IJCAI'16*, pages 900–906. AAAI Press, 2016.

[4] L. Amgoud and J. Ben-Naim. Evaluation of arguments in weighted bipolar graphs. In A. Antonucci, L. Cholvy, and O. Papini, editors, *Proceedings of ECSQARU'17*, volume 10369 of *LNCS*, pages 25–35. Springer, 2017.

[5] L. Amgoud, J. Ben-Naim, D. Doder, and S. Vesic. Ranking arguments with compensation-based semantics. In C. Baral, J. Delgrande, and F. Wolter, editors, *Proceedings of KR'16*, pages 12–21. AAAI Press, 2016.

[6] L. Amgoud, J. Ben-Naim, D. Doder, and S. Vesic. Acceptability semantics for weighted argumentation frameworks. In *Proceedings of IJCAI'17*. AAAI Press, 2017. Forthcoming.

[7] L. Amgoud and H. Prade. Using arguments for making and explaining decisions. *Artificial Intelligence*, 173(3):413 – 436, 2009.

[8] D. Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[9] P. Baroni, M. Caminada, and M. Giacomin. An introduction to argumentation semantics. *The Knowledge Engineering Review*, 26(4):365–410, 2011.

[10] P. Baroni, F. Cerutti, M. Giacomin, and G. Guida. AFRA: Argumentation framework with recursive attacks. *International Journal of Approximate Reasoning*, 52(1):19–37, 2011.

[11] P. Besnard and S. Doutre. Checking the acceptability of a set of arguments. In J. P. Delgrande and T. Schaub, editors, *Proceedings of NMR'04*, pages 59–64, 2004.

[12] P. Besnard, S. Doutre, and A. Herzig. Encoding argument graphs in logic. In A. Laurent, O. Strauss, B. Bouchon-Meunier, and R. R. Yager, editors, *Proceedings of IPMU'14*, pages 345–354. Springer, 2014.

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

[14] P. Besnard and A. Hunter. Constructing argument graphs with deductive arguments: A tutorial. *Argument & Computation*, 5(1):5–30, 2014.

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

[16] G. Boella, D. Gabbay, L. van der Torre, and S. Villata. Support in abstract argumentation. In P. Baroni, F. Cerutti, M. Giacomin, and G. R. Simari, editors, *Proceedings of COMMA'10*, volume 216 of *FAIA*, pages 111–122. IOS Press, 2010.

[17] E. Bonzon, J. Delobelle, S. Konieczny, and N. Maudet. A comparative study of ranking-based semantics for abstract argumentation. In D. Schuurmans and M. Wellman, editors, *Proceedings of AAAI'16*, pages 914–920. AAAI Press, 2016.

[18] G. Brewka, S. Ellmauthaler, H. Strass, J. P. Wallner, and S. Woltran. Abstract dialectical frameworks revisited. In F. Rossi, editor, *Proceedings of IJCAI'13*, pages 803–809. AAAI Press, 2013.

[19] G. Brewka, S. Polberg, and S. Woltran. Generalizations of Dung frameworks and their role in formal argumentation. *IEEE Intelligent Systems*, 29(1):30–38, Jan 2014.

[20] G. Brewka and S. Woltran. Abstract dialectical frameworks. In F. Lin, U. Sattler, and M. Truszczynski, editors, *Proceedings of KR'10*, pages 102–111. AAAI Press, 2010.

[21] E. Cabrio and S. Villata. A natural language bipolar argumentation approach to support users in online debate interactions. *Argument & Computation*, 4(3):209–230, 2013.

[22] M. Caminada and D. M. Gabbay. A logical account of formal argumentation. *Studia Logica*, 93:109–145, 2009.

[23] C. Cayrol and M. Lagasquie-Schiex. Gradual valuation for bipolar argumentation frameworks. In L. Godo, editor, *Proceedings of ECSQARU'05*, volume 3571 of *LNCS*, pages 366–377. Springer, 2005.

[24] C. Cayrol and M. Lagasquie-Schiex. Graduality in argumentation. *Journal of Artificial Intelligence Research*, 23:245–297, 2005.

[25] C. Cayrol and M.-C. Lagasquie-Schiex. Bipolarity in argumentation graphs: Towards a better understanding. *International Journal of Approximate Reasoning*, 54(7):876–899, 2013.

[26] F. Cerutti, N. Tintarev, and N. Oren. Formal arguments, preferences, and natural language interfaces to humans: an empirical evaluation. In T. Schaub, G. Friedrich, and B. O'Sullivan, editors, *Proceedings of ECAI'14*, volume 263 of *FAIA*, pages 1033–1034. IOS Press, August 2014.

[27] V. Chvátal. *Linear Programming*. Freeman, San Francisco, 1983.

[28] S. Coste-Marquis, C. Devred, S. Konieczny, M. Lagasquie-Schiex, and P. Marquis. On the merging of dung's argumentation systems. *Artificial Intelligence*, 171(10-15):730–753, 2007.

[29] S. Coste-Marquis, C. Devred, and P. Marquis. Inference from controversial arguments. In G. Sutcliffe and A. Voronkov, editors, *Proceedings of LPAR'05*, volume 3835 of *LNCS*, pages 606–620. Springer, 2005.

[30] S. Coste-Marquis, C. Devred, and P. Marquis. Prudent semantics for argumentation frameworks. In A. Lim, editor, *Proceedings of ICTAI'05*, pages 568–572. IEEE Computer Society, 2005.

[31] S. Coste-Marquis, C. Devred, and P. Marquis. Constrained argumentation frameworks. In P. Doherty, J. Mylopoulos, and C. Welty, editors, *Proceedings of KR'06*, pages 112–122. AAAI Press, 2006.

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

[33] P. E. Dunne and T. J. M. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 141(1–2):187–203, 2002.

[34] P. E. Dunne and M. Caminada. Computational complexity of semi-stable semantics in abstract argumentation frameworks. In S. Hölldobler, C. Lutz, and H. Wansing, editors, *Proceedings of JELIA'06*, volume 5293 of *LNCS*, pages 153–165. Springer, 2008.

[35] W. Dvořák, M. Järvisalo, J. P. Wallner, and S. Woltran. Complexity-sensitive decision procedures for abstract argumentation. *Artificial Intelligence*, 206:53 – 78, 2014.

[36] W. Dvořák and S. Woltran. Complexity of semi-stable and stage semantics in argumentation frameworks. *Information Processing Letters*, 110(11):425 – 430, 2010.

[37] B. Fazzinga, S. Flesca, and F. Parisi. On the complexity of probabilistic abstract argumentation frameworks. *ACM Transactions on Computational Logic*, 16(3):22:1–22:39, 2015.

[38] J. Fox and S. K. Das. *Safe and sound - artificial intelligence in hazardous applications*. MIT Press, 2000.

[39] D. M. Gabbay. Semantics for higher level attacks in extended argumentation frames part 1: Overview. *Studia Logica*, 93(2-3):357–381, 2009.

[40] G. Georgakopoulos, D. Kavvadias, and C. H. Papadimitriou. Probabilistic satisfiability. *Journal of Complexity*, 4:1–11, 1988.

[41] J. P. W. Gerhard Brewka, Hannes Strass and S. Woltran. Weighted abstract dialectical frameworks. In *Proceedings of AAAI'18*, 2018. In Press.

[42] E. Hadoux and A. Hunter. Computationally viable handling of beliefs in arguments for persuasion. In N. Bourbakis, A. Esposito, A. Mali, and M. Alamaniotis, editors, *Proceedings of ICTAI'16*, pages 319–326. IEEE, 2016.

[43] E. Hadoux and A. Hunter. Strategic sequences of arguments for persuasion using decision trees. In S. Singh and S. Markovitch, editors, *Proceedings of AAAI'17*, pages 1128–1134. AAAI Press, 2017.

[44] E. Hadoux and A. Hunter. Learning and updating user models for subpopulations in persuasive argumentation using beta distributions. In *Proceedings of AAMAS'18*, 2018. In Press.

[45] P. Hansen and B. Jaumard. Probabilistic satisfiability. In *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, pages 321–367. Springer, 2000.

[46] A. Hunter. A probabilistic approach to modelling uncertain logical arguments. *International Journal of Approximate Reasoning*, 54(1):47–81, 2013.

[47] A. Hunter. Modelling the persuadee in asymmetric argumentation dialogues for persuasion. In Q. Yang and M. Wooldridge, editors, *Proceedings of IJCAI'15*, pages 3055–3061. AAAI Press, 2015.

[48] A. Hunter. Computational persuasion with applications in behaviour change. In P. Baroni, T. F. Gordon, T. Scheffler, and M. Stede, editors, *Proceedings of COMMA'16*, volume 287 of *FAIA*, pages 5–18. IOS Press, 2016.

[49] A. Hunter. Persuasion dialogues via restricted interfaces using probabilistic argumentation. In S. Schockaert and P. Senellart, editors, *Proceedings of SUM'16*, volume 9858 of *LNCS*, pages 184–198. Springer, 2016.

[50] A. Hunter. Two dimensional uncertainty in persuadee modelling in argumentation. In G. A. Kaminka, M. Fox, P. Bouquet, E. Hüllermeier, V. Dignum, F. Dignum, and F. van Harmelen, editors, *Proceedings of ECAI'16*, volume 285 of *FAIA*, pages 150–157. IOS Press, 2016.

[51] A. Hunter and N. Potyka. Updating probabilistic epistemic states in persuasion dialogues. In A. Antonucci, L. Cholvy, and O. Papini, editors, *Proceedings of ECSQARU'17*, volume 10369 of *LNCS*, pages 46–56. Springer, 2017.

[52] A. Hunter and M. Thimm. Probabilistic argumentation with incomplete information. In T. Schaub, G. Friedrich, and B. O'Sullivan, editors, *Proceedings of ECAI'14*, volume 263 of *FAIA*, pages 1033–1034. IOS Press, 2014.

[53] A. Hunter and M. Thimm. Probabilistic reasoning with abstract argumentation frameworks. *Journal of Artificial Intelligence Research*, 59:565–611, 2017.

[54] D. Kontarinis and F. Toni. Identifying malicious behavior in multi-party bipolar argumentation debates. In M. Rovatsos, G. Vouros, and V. Julian, editors, *Proceedings of EUMAS'15*, pages 267–278. Springer, 2016.

[55] J. Leite and J. Martins. Social abstract argumentation. In T. Walsh, editor, *Proceedings of IJCAI'11*, pages 2287–2292. AAAI Press, 2011.

[56] H. Li, N. Oren, and T. J. Norman. Probabilistic argumentation frameworks. In S. Modgil, N. Oren, and F. Toni, editors, *Proceedings of TAFA'11*, volume 7132 of *LNCS*, pages 1–16. Springer, 2011.

[57] T. Lukasiewicz. Probabilistic default reasoning with conditional constraints. *Annals of Mathematics and Artifical Intelligence*, 34:35–88, 2002.

[58] S. Modgil. Reasoning about preferences in argumentation frameworks. *Artificial Intelligence*, 173(9-10):901–934, 2009.

[59] N. Nilsson. Probabilistic logic. *Artificial intelligence*, 28(1):71–87, 1986.

[60] F. Nouioua and V. Risch. Argumentation frameworks with necessities. In S. Benferhat and J. Grant, editors, *Proceedings of SUM'11*, volume 6929 of *LNCS*, pages 163–176. Springer, 2011.

[61] J. Ogden. *Health Psychology: A Textbook*. Open University Press, 5 edition, 2012.

[62] N. Oren and T. J. Norman. Semantics for evidence-based argumentation. In P. Besnard, S. Doutre, and A. Hunter, editors, *Proceedings of COMMA'08*, volume 172 of *FAIA*, pages 276–284. IOS Press, 2008.

[63] C. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[64] J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

[65] S. Polberg. Extension-based semantics of abstract dialectical frameworks. In U. Endriss and J. Leite, editors, *Proceedings of STAIRS'14*, volume 264 of *FAIA*, pages 240–249. IOS Press, 2014.

[66] S. Polberg. Revisiting extension–based semantics of abstract dialectical frameworks. Technical Report DBAI-TR-2015-88, Institute for Information Systems, Technical University of Vienna, 2015.

[67] S. Polberg. Understanding the Abstract Dialectical Framework. In L. Michael and A. Kakas, editors, *Proceedings of JELIA'16*, volume 10021 of *LNCS*, pages 430–446. Springer, 2016.

[68] S. Polberg. Intertranslatability of abstract argumentation frameworks. Technical Report DBAI-TR-2017-104, Institute for Information Systems, Technical University of Vienna, 2017.

[69] S. Polberg and D. Doder. Probabilistic abstract dialectical frameworks. In E. Fermé and J. Leite, editors, *Proceedings of JELIA'14*, volume 8761 of *LNCS*, pages 591–599. Springer, 2014.

[70] S. Polberg and A. Hunter. Empirical methods for modelling persuadees in dialogical argumentation. In *Proceedings of ICTAI'17*, 2017. In Press.

[71] S. Polberg and A. Hunter. Empirical evaluation of abstract argumentation: Supporting the need for bipolar and probabilistic approaches. *International Journal of Approximate Reasoning*, 93:487 – 543, 2018.

[72] S. Polberg, A. Hunter, and M. Thimm. Belief in attacks in epistemic probabilistic argumentation. In S. Moral, O. Pivert, D. Sánchez, and N. Marín, editors, *Proceedings of SUM'17*, volume 10564 of *LNCS*, pages 223–236. Springer, 2017.

[73] S. Polberg and N. Oren. Revisiting support in abstract argumentation systems. In S. Parsons, N. Oren, C. Reed, and F. Cerutti, editors, *Proceedings of COMMA'14*, volume 266 of *FAIA*, pages 369–376. IOS Press, 2014.

[74] H. Prakken. On support relations in abstract argumentation as abstractions of inferential relations. In T. Schaub, G. Friedrich, and B. O'Sullivan, editors, *Proceedings ECAI'14*, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pages 735–740. IOS Press, 2014.

[75] F. Pu, J. Luo, Y. Zhang, and G. Luo. Argument ranking with categoriser function. In R. Buchmann, C. V. Kifor, and J. Yu, editors, *Proceedings of KSEM'14*, pages 290–301. Springer International Publishing, 2014.

[76] A. Rago, F. Toni, M. Aurisicchio, and P. Baroni. Discontinuity-free decision support with quantitative argumentation debates. In C. Baral, J. Delgrande, and F. Wolter, editors, *Proceedings of KR'16*, pages 63–73. AAAI Press, 2016.

[77] I. Rahwan, M. Madakkatel, J. Bonnefon, R. Awan, and S. Abdallah. Behavioural experiments for assessing the abstract argumentation semantics of reinstatement. *Cognitive Science*, 34(8):1483–1502, 2010.

[78] R. Reiter. *On Closed World Data Bases*, pages 55–76. Springer, 1978.

[79] A. Rosenfeld and S. Kraus. Strategical argumentative agent for human persuasion. In G. A. Kaminka, M. Fox, P. Bouquet, E. Hüllermeier, V. Dignum, F. Dignum, and F. van Harmelen, editors, *Proceedings of ECAI'16*, volume 285 of *FAIA*, pages 320–328. IOS Press, 2016.

[80] H. Strass. Approximating operators and semantics for abstract dialectical frameworks. *Artificial Intelligence*, 205:39 – 70, 2013.

[81] H. Strass. Instantiating knowledge bases in abstract dialectical frameworks. In J. Leite, T. C. Son, P. Torroni, L. van der Torre, and S. Woltran, editors, *Proceedings of CLIMA'13*, volume 8143 of *LNCS*, pages 86–101. Springer, 2013.

[82] H. Strass and J. P. Wallner. Analyzing the Computational Complexity of Abstract Dialectical Frameworks via Approximation Fixpoint Theory. *Artificial Intelligence*, 226:34–74, 2015.

[83] M. Thimm. A probabilistic semantics for abstract argumentation. In L. D. Raedt, C. Bessiere, D. Dubois, P. Doherty, P. Frasconi, F. Heintz, and P. Lucas, editors, *Proceedings of ECAI'12*, volume 242 of *FAIA*, pages 750–755. IOS Press, 2012.

[84] Z. Zeng. Context-based and explainable decision making with argumentation. In *Proceedings of AAMAS'18*, 2018. In Press.

9 Proof Appendix

Lemma 4.7. *The following hold:*

- *If* Π *is a nonempty restricted value set, then* $0 \in \Pi$.
- *If* Π *is a reasonable restricted value set, then* $0 \in \Pi$.
- *A restricted value set* Π *is reasonable iff* $1 \in \Pi$.

Proof.
- If Π is nonempty, then there exists $x \in [0,1]$ s.t. $x \in \Pi$. By the definition of the restricted value set, $x - x \in \Pi$. Hence, $0 \in \Pi$.
- Let $\mathsf{Nodes}(\mathcal{G}) \neq \emptyset$. Since Π is reasonable, then $\mathsf{Dist}(\mathcal{G}, \Pi) \neq \emptyset$. Hence, there exists $P \in \mathsf{Dist}(\mathcal{G}, \Pi)$ and for every $X \subseteq \mathsf{Nodes}(\mathcal{G})$, there exist a value $y \in [0,1]$ s.t. $P(X) = y$. Hence, $y \in \Pi$ and Π is nonempty. Thus, based on the previous part of this proof, $0 \in \Pi$.
- If Π is a reasonable restricted value set, then for any nonempty graph, $\mathsf{Dist}(\mathcal{G}, \Pi) \neq \emptyset$. Hence, we can find $x_1, \ldots, x_n \in \Pi$ s.t. $\sum_{i=1}^{n} x_i = 1$. Since Π is a restricted value set, then $x_1 + x_2 = y_1 \in \Pi$, $y_1 + x_3 = y_2 \in \Pi$, $\ldots$, $y_{n-2} + x_n = 1 \in \Pi$.

  Let Π be a nonempty restricted value set s.t. $1 \in \Pi$. By the previous parts of this proof, $0 \in \Pi$. Thus, for any graph $\mathcal{G}$ s.t. $\mathsf{Nodes}(\mathcal{G}) \neq \emptyset$, we can create a trivial distribution P s.t. $P(\emptyset) = 1$ and $\forall X \subseteq \mathsf{Nodes}(\mathcal{G})$ s.t $X \neq \emptyset$, $P(X) = 0$. Consequently, $\mathsf{Dist}(\mathcal{G}, \Pi) \neq \emptyset$ and Π is reasonable.

$\square$

Proposition 4.8. *Let* Π *be a nonempty restricted value set,* $x \in \Pi$ *a value,* $\# \in \{=, \neq, \geq, \leq, >, <\}$ *an inequality, and* $(*_1, \ldots, *_k)$ *a sequence of operators where* $*_i \in \{+, -\}$ *and* $k \geq 0$. *Let* $max(\Pi)$ *denote the maximal value of* Π. *The following hold:*

- $\Pi^x_\# = \emptyset$ *if and only if:*
  1. $\Pi = \{0\}$ *and* $\# = \neq$, *or*
  2. $\#$ *is* $>$ *and* $x = max(\Pi)$, *or*
  3. $\#$ *is* $<$ *and* $x = 0$.
- $\Pi^{x,(*_1,\ldots,*_k)}_\# = \emptyset$ *if and only if:*
  1. $k = 0$ *and* $\Pi^x_\# = \emptyset$, *or*
  2. $k > 0$, $\#$ *is* $>$, $x = max(\Pi)$ *and for no* $*_i$, $*_i = +$, *or*
  3. $k > 0$, $\#$ *is* $>$ *and* $\Pi = \{0\}$, *or*
  4. $k > 0$, $\#$ *is* $<$, $x = 0$ *and for no* $*_i$, $*_i = -$, *or*
  5. $k > 0$, $\#$ *is* $<$ *and* $\Pi = \{0\}$.
  6. $k > 0$, $\#$ *is* $\neq$, $\Pi = \{0\}$.

Proof. Let us focus on the first case. It is easy to see that if any of the conditions are met, then $\Pi^x_\# = \emptyset$. What remains to be shown is that if $\Pi^x_\# = \emptyset$, then one of these conditions has to be satisfied. Assume it is not the case, i.e. $\Pi^x_\# = \emptyset$, but none of the conditions is satisfied. In order for the conditions to be not met, either of the following cases has to hold:

- $\Pi \neq \{0\}$ and $\#$ is neither $>$ nor $<$. Hence, $\#$ is $\geq, \leq, =$ or $\neq$. Clearly, $x \in \Pi^x_\geq$, $x \in \Pi^x_\leq$ and $x \in \Pi^x_=$, hence the produced set cannot be empty for these inequalities. Since $\Pi \neq \{0\}$ and $\Pi \neq \emptyset$, by using Lemma 4.7 we can show that there exists $y \in \Pi$ s.t. $y \neq 0$. Thus, if $x = 0$, then $y \in \Pi^x_\neq$, and of $x \neq 0$, then $0 \in \Pi^x_\neq$. Thus, under the aforementioned assumptions, $\Pi^x_\# \neq \emptyset$ and we reach a contradiction.

- $\Pi \neq \{0\}$ and $\#$ is not $>$ and $x \neq 0$. We can observe that $\#$ can be either $<, \geq, \leq, =$ or $\neq$. Clearly, $x \in \Pi^x_{\geq}$, $x \in \Pi^x_{\leq}$ and $x \in \Pi^x_{=}$, hence the produced set cannot be empty for these inequalities. Using the same argument as in the first point, we can observe that there exists $y \in \Pi$ s.t. $y \neq 0$. It also clearly follows that $y > 0$, hence $0 \in \Pi^x_{<}$. Finally, since $x \neq 0$, then $0 \in \Pi^x_{\neq}$. Thus, in all possible cases, $\Pi^x_{\#} \neq \varnothing$ and we reach a contradiction.

- $\Pi \neq \{0\}$ and $x \neq max(\Pi)$ and $\#$ is not $<$. We can observe that $\#$ can be either $>, \geq, \leq, =$ or $\neq$. Clearly, $x \in \Pi^x_{\geq}$, $x \in \Pi^x_{\leq}$ and $x \in \Pi^x_{=}$, hence the produced set cannot be empty for these inequalities. Using the same argument as in the first point, we can observe that there exists $y \in \Pi$ s.t. $y \neq 0$ and $y = max(\Pi)$. Hence, $y \in \Pi^x_{\neq}$. Finally, since $x \neq max(\Pi)$, then $y > x$, and $y \in\in \Pi^x_{>}$. Thus, in all possible cases, $\Pi^x_{\#} \neq \varnothing$ and we reach a contradiction.

- $\Pi \neq \{0\}$ and $x \neq max(\Pi)$ and $x \neq 0$. Since $\#$ is not constrained, it can be either $>, <, \geq, \leq, =$ or $\neq$. Clearly, $x \in \Pi^x_{\geq}$, $x \in \Pi^x_{\leq}$ and $x \in \Pi^x_{=}$, hence the produced set cannot be empty for these inequalities. Using the same argument as in the first point, we can observe that there exists $y \in \Pi$ s.t. $y \neq 0$ and $y = max(\Pi)$. Since $x \in \Pi$, $x \neq 0$, $x \neq max(\Pi)$ and $0 \neq max(\Pi)$, then then $0 < x < y$. Consequently, $0 \in \Pi^x_{\neq}$, $0 \in \Pi^x_{<}$ and $y \in \Pi^x_{>}$. Thus, in all possible cases, $\Pi^x_{\#} \neq \varnothing$ and we reach a contradiction.

- $\# \notin \{\neq, >, <\}$. This means that $\#$ is either $\geq$, $\leq$ or $=$. Clearly, $x \in \Pi^x_{\geq}$, $x \in \Pi^x_{\leq}$ and $x \in \Pi^x_{=}$. Thus, in all possible cases, $\Pi^x_{\#} \neq \varnothing$ and we reach a contradiction.

- $\#$ is not $\neq$ and $\#$ is not $>$ and $x \neq 0$. This means that $\#$ is either $<, \geq, \leq$ or $=$. Clearly, $x \in \Pi^x_{\geq}$, $x \in \Pi^x_{\leq}$ and $x \in \Pi^x_{=}$. Furthermore, since $\Pi \neq \varnothing$, then by Lemma 4.7 $0 \in \Pi$. Thus, if $x \neq 0$, then $0 \in \Pi^x_{<}$. Hence, in all possible cases, $\Pi^x_{\#} \neq \varnothing$ and we reach a contradiction.

- $\#$ is not $\neq$ and $x \neq max(\Pi)$ and $\#$ is not $<$. This means that $\#$ is either $>, \geq, \leq$ or $=$. Clearly, $x \in \Pi^x_{\geq}$, $x \in \Pi^x_{\leq}$ and $x \in \Pi^x_{=}$. Furthermore, since $x \neq max(\Pi)$ and $\Pi \neq \varnothing$, then there exists $y \in \Pi$ s.t. $y \in \Pi$ and $y > x$. Hence, $y \in \Pi^x_{>}$. Thus, in all possible cases, $\Pi^x_{\#} \neq \varnothing$ and we reach a contradiction.

- $\#$ is not $\neq$ and $x \neq max(\Pi)$ and $x \neq 0$. This means that $\#$ is either $>, <, \geq, \leq$ or $=$. Clearly, $x \in \Pi^x_{\geq}$, $x \in \Pi^x_{\leq}$ and $x \in \Pi^x_{=}$. We therefore need to analyze $<$ and $>$. Since $x \in \Pi$ and $x \neq max(\Pi)$, then $x < max(\Pi)$, and as $x \neq 0$, $x > 0$. Hence, $0 < x < max(\Pi)$, and it holds that $0 \in \Pi^x_{<}$ and $max(\Pi) \in \Pi^x_{>}$. Thus, in all possible cases, $\Pi^x_{\#} \neq \varnothing$ and we reach a contradiction.

This proves that $\Pi^x_{\#} = \varnothing$ if and only if one of the listed conditions is met.

Let us now analyze the combination sets. It is easy to verify that if any of the conditions is met, then the resulting combination set is indeed empty. Let us therefore show that if the combination set is empty, then one of the conditions is met. Let us assume that it is not the case, i.e. $\Pi^{x,(*_1,\ldots,*_k)}_{\#} = \varnothing$ but no condition is satisfied.

Let $\Pi^{x,(*_1,\ldots,*_k)}_{\#} = \varnothing$ and assume that $k = 0$. This means that $\Pi^{x,(*_1,\ldots,*_k)}_{\#} = \{(v) \mid v \in \Pi^x_{\#}\}$. Thus, $\Pi^{x,(*_1,\ldots,*_k)}_{\#}$ is empty iff $\Pi^x_{\#}$ is empty. However, this means that we satisfy the first condition and thus reach a contradiction with our assumptions.

Let $\Pi^{x,(*_1,\ldots,*_k)}_{\#} = \varnothing$ and assume that $k > 0$. Since Π is nonempty, then by Lemma 4.7, $0 \in \Pi$. Thus, for any x, any k and any sequence $(*_1, \ldots, *_k)$, we can create a trivial tuple $(v_1, \ldots, v_{k+1})$ s.t. $v_1 *_1 v_2 \ldots v_k *_k v_{k+1} = x$. This is simply achieved by setting $v_1 = x$ and $v_i = 0$, where $i > 1$. Hence, for $\# \in \{\geq, \leq, =\}$, clearly $\Pi^{x,(*_1,\ldots,*_k)}_{\#} \neq \varnothing$. Let us therefore focus on $\# \in \{>, <, \neq\}$ and start with $>$. Let $y = max(\Pi)$. If $x \neq y$, then $y > x$, and we can create a sequence $(v_1, \ldots, v_{k+1})$ s.t. $v_1 *_1 v_2 \ldots v_k *_k v_{k+1} > x$ by setting $v_1 = y$ and $v_i = 0$, where $i > 1$. Hence, we reach a contradiction for this case. If $x = y$, then if $y \neq 0$ (recall that values from Π belong in the unit interval), then $y + y > x$. Consequently, if there is at least one j s.t. $*_j = +$, we can create a sequence $(v_1, \ldots, v_{k+1})$ s.t. $v_1 *_1 v_2 \ldots v_k *_k v_{k+1} > x$ by setting $v_1 = v_j = y$ and $v_i = 0$, where $i > 1$ and $i \neq j$. If there is no addition present (i.e. we only have subtractions), then it is easy to see that the maximal value we can obtain from our formula is when $v_1 = y$ and the remaining values are set to 0. Thus, in this case, $v_1 *_1 v_2 \ldots v_k *_k v_{k+1} = y$ and since $x = y$, then our combination set is empty. However, this scenario coincides with one of our conditions that was not supposed to be satisfied, and we reach a contradiction. We are therefore left with the case where $x = y = 0$. Since $y = max(\Pi)$, then clearly $\Pi = \{0\}$ and independently of the used arithmetic operators and values, every formula will always amount to 0. Since $0 \not> 0$, our combination set is empty. However, this scenario is again covered by one of our conditions and we reach a contradiction.

Let us now focus on $<$. It is easy to see that since $0 \in \Pi$, then as long as $x \neq 0$, we can observe that $v_1 *_1 v_2 \dots v_k *_k v_{k+1} < x$ for $v_i = 0$. Hence, in such a case, the combination set would never be empty. Thus, consider the case where $x = 0$. If $\Pi = \{0\}$, then the smallest value obtainable by $v_1 *_1 v_2 \dots v_k *_k v_{k+1}$ is 0, and the combination set is therefore empty. However, this is already covered by one of our conditions, and we reach a contradiction. If $\Pi \neq \{0\}$, then as long as there is at least one $*_j$ s.t. $*_j = -$, we can obtain a formula producing a value smaller than 0 and the combination set is nonempty. If the sequence of operations does not contain any subtractions, then the smallest value obtainable by $v_1 *_1 v_2 \dots v_k *_k v_{k+1}$ is again 0, and the combination set is empty. However, this again is covered by one of our conditions and we reach a contradiction.

Finally, we can consider $\neq$. Let $y = max(\Pi)$. If $y \neq 0$ and $x \neq y$, then a tuple s.t. the first position is y and every other value is 0 will be in the combination set irrespective of the sequence of operators. If $y \neq 0$ and $x = y$, then a tuple of 0's will be in the combination set irrespective of the sequence of operators. If $y = 0$ then $\Pi = \{0\}$ and therefore $x = 0$ as well. In this case, independently of the sequence of operators, every possible formula will evaluate to 0 and the combination set will be empty. However, this case is covered by one of our conditions, and we reach a contradiction.

Therefore, we have shown that the combination set is empty iff one of our conditions is met. □

Proposition 4.10. *Let Π be a restricted value set, $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$ a set of epistemic formulae, and $\psi \in \mathsf{VFormulae}(\mathcal{G})$ an epistemic formula. If $\Phi \Vdash \psi$ then $\Phi \Vdash_\Pi \psi$.*

Proof. Assume $\Phi \Vdash \psi$. Therefore, $\mathsf{Sat}(\Phi) \subseteq \mathsf{Sat}(\psi)$. Therefore, $\mathsf{Sat}(\Phi) \cap \mathsf{Dist}(\mathcal{G}, \Pi) \subseteq \mathsf{Sat}(\psi) \cap \mathsf{Dist}(\mathcal{G}, \Pi)$. Therefore, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Sat}(\psi, \Pi)$. Therefore, $\Phi \Vdash_\Pi \psi$.

□

Proposition 4.11. *Let $\Pi_1 \subseteq \Pi_2$ be restricted value sets, $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi_1)$ a set of epistemic formulae, and $\psi \in \mathsf{VFormulae}(\mathcal{G})$ an epistemic formula. If $\Phi \Vdash_{\Pi_2} \psi$ then $\Phi \Vdash_{\Pi_1} \psi$.*

Proof. Let X, Y, W, Z be sets of elements s.t. $W \subseteq Z$. It is easy to show that if $X \cap Z \subseteq Y \cap Z$, then $X \cap Z \cap W \subseteq Y \cap Z \cap W$. Since, $W \subseteq Z$, then $X \cap Z \cap W = X \cap W$ and $Y \cap Z \cap W = Y \cap W$. Thus, $X \cap Z \subseteq Y \cap Z$ implies $X \cap W \subseteq Y \cap W$ when $W \subseteq Z$.

We can show that if $\Pi_1 \subseteq \Pi_2$, then $\mathsf{Dist}(\Pi_1) \subseteq \mathsf{Dist}(\Pi_2)$. Hence, using the above analysis, it is easy to prove that if $\mathsf{Sat}(\Phi) \cap \mathsf{Dist}(\Pi_2) \subseteq \mathsf{Sat}(\psi) \cap \mathsf{Dist}(\Pi_2)$ then $\mathsf{Sat}(\Phi) \cap \mathsf{Dist}(\Pi_1) \subseteq \mathsf{Sat}(\psi) \cap \mathsf{Dist}(\Pi_1)$. Thus, $\Phi \Vdash_{\Pi_2} \psi$ then $\Phi \Vdash_{\Pi_1} \psi$. □

Proposition 4.16. *Let $P \in \mathsf{Dist}(\mathcal{G})$ be a probability distribution and φ^P its associated epistemic formula. Then $\{P\} = \mathsf{Sat}(\varphi^P)$.*

Proof. Let us assume that the arguments in $\mathcal{G}$ are ordered according to some ordering. Let $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \dots, c_j\}$ be the collection of all argument complete propositional formulae for $\mathcal{G}$ and $\varphi^P = p(c_1) = x_1 \wedge p(c_2) = x_2 \wedge \dots \wedge p(c_j) = x_j$, where $x_i = P(c_i)$, the epistemic formula associated with P.

By definition, P is an assignment, where the elements of the powerset of arguments are mapped to numerical values s.t. these values add up to 1. Every set of arguments in the powerset can be described with a binary number, where if an i-th digit is 1, then the i-th argument is in the set, and it if its 0, then it is not in the set. We can observe that every argument complete formula has precisely one model which is trivially constructed – if the i-th argument appears as a positive literal in the formula, then it is in the model, if it appears as a negative literal, then it is not in the model.

We can therefore observe that a given complete formula encodes precisely one set of arguments from the powerset and the epistemic atom involving it demands that the value assigned to the formula is the same as the value assigned to the corresponding set by the probability distribution. It is therefore easy to see that φ^P is satisfied only by P. Hence, $\{P\} = \mathsf{Sat}(\varphi^P)$. □

Proposition 4.18. *Let Π be a reasonable restricted value set, $\psi \in \mathsf{VFormulae}(\mathcal{G},\Pi)$ be a restricted epistemic formula and φ its distribution disjunctive normal form. Then* $\mathsf{Sat}(\psi,\Pi) = \mathsf{Sat}(\varphi,\Pi)$.

Proof. Assume $\mathsf{Sat}(\psi,\Pi) = \varnothing$. Then, $\varphi : \bot$ and $\mathsf{Sat}(\bot,\Pi) = \varnothing$. Hence, $\mathsf{Sat}(\psi,\Pi) = \mathsf{Sat}(\varphi,\Pi)$.

Assume $\mathsf{Sat}(\psi,\Pi) \neq \varnothing$. Let $\varphi : \varphi^{P_1} \vee \varphi^{P_2} \ldots \vee \varphi^{P_n}$. Then, $\mathsf{Sat}(\varphi,\Pi) = \mathsf{Sat}(\varphi^{P_1},\Pi) \cup \mathsf{Sat}(\varphi^{P_2},\Pi) \cup \ldots \cup \mathsf{Sat}(\varphi^{P_n},\Pi)$. Since $P_i \in \mathsf{Sat}(\psi,\Pi)$, then $P_i \in \mathsf{Dist}(\mathcal{G},\Pi)$ as well. Thus, based on Proposition 4.16, $\mathsf{Sat}(\varphi^{P_i},\Pi) = \{P_i\}$. We can therefore show that $\mathsf{Sat}(\varphi,\Pi) = \{P_1,\ldots,P_n\} = \mathsf{Sat}(\psi,\Pi)$. $\square$

Proposition 4.20. *Let $\# \in \{=,\neq,\geq,\leq,>,<\}$ be the set of inequality relationships, let Π be a reasonable restricted value set, and let $\Pi^x_\# = \{y \in \Pi \mid x\#y\}$ be the subset obtained according to the value x and relationship $\#$. The following hold, where $\vdash$ is propositional consequence relation, $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G},\Pi)$, $\phi,\psi \in \mathsf{BFormulae}(\mathcal{G},\Pi)$ and $x \in \Pi$.*

1. $\Phi \Vdash_\Pi p(\alpha) \geq 0$
2. $\Phi \Vdash_\Pi p(\alpha) \leq 1$
3. $\Phi \Vdash_\Pi p(\top) = 1$
4. $\Phi \Vdash_\Pi p(\bot) = 0$
5. $\Phi \Vdash_\Pi p(\alpha) > x$ *implies* $\Phi \Vdash_\Pi p(\alpha) \geq x$
6. $\Phi \Vdash_\Pi p(\alpha) < x$ *implies* $\Phi \Vdash_\Pi p(\alpha) \leq x$
7. $\Phi \Vdash_\Pi p(\alpha) = x$ *implies* $\Phi \Vdash_\Pi p(\alpha) \geq x$
8. $\Phi \Vdash_\Pi p(\alpha) = x$ *implies* $\Phi \Vdash_\Pi p(\alpha) \leq x$
9. $\Phi \Vdash_\Pi p(\alpha) \neq x$ *iff* $\Phi \Vdash_\Pi p(\alpha) > x \vee p(\alpha) < x$
10. $\Phi \Vdash_\Pi p(\alpha) \geq x$ *iff* $\Phi \Vdash_\Pi p(\alpha) > x \vee p(\alpha) = x$
11. $\Phi \Vdash_\Pi p(\alpha) \leq x$ *iff* $\Phi \Vdash_\Pi p(\alpha) < x \vee p(\alpha) = x$
12. $\Phi \Vdash_\Pi p(\alpha) > x$ *iff* $\Phi \Vdash_\Pi p(\alpha) \geq x \wedge p(\alpha) \neq x$
13. $\Phi \Vdash_\Pi p(\alpha) < x$ *iff* $\Phi \Vdash_\Pi p(\alpha) \leq x \wedge p(\alpha) \neq x$
14. $\Phi \Vdash_\Pi p(\alpha) = x$ *iff* $\Phi \Vdash_\Pi p(\alpha) \geq x \wedge p(\alpha) \leq x$
15. $\Phi \Vdash_\Pi p(\alpha) > x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) \leq x)$
16. $\Phi \Vdash_\Pi p(\alpha) < x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) \geq x)$
17. $\Phi \Vdash_\Pi p(\alpha) \leq x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) > x)$
18. $\Phi \Vdash_\Pi p(\alpha) \geq x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) < x)$
19. $\Phi \Vdash_\Pi p(\alpha) = x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) \neq x)$
20. $\Phi \Vdash_\Pi p(\alpha) \neq x$ *iff* $\Phi \Vdash_\Pi \neg(p(\alpha) = x)$
21. $\Phi \Vdash_\Pi p(\alpha) = x$ *and* $\{\alpha\} \vdash \beta$ *implies* $\Phi \Vdash_\Pi p(\beta) \geq x$
22. $\Phi \Vdash_\Pi p(\alpha) = x$ *and* $\{\beta\} \vdash \alpha$ *implies* $\Phi \Vdash_\Pi p(\beta) \leq x$
23. *if* $\{\alpha\} \vdash \beta$ *and* $\{\beta\} \vdash \alpha$ *then* $\Phi \Vdash_\Pi p(\alpha) = x$ *iff* $\Phi \Vdash_\Pi p(\beta) = x$
24. $\Phi \Vdash_\Pi p(\alpha) > x$ *and* $x \geq y$ *implies* $\Phi \Vdash_\Pi p(\alpha) > y$
25. $\Phi \Vdash_\Pi p(\alpha) > x$ *and* $x > y$ *implies* $\Phi \Vdash_\Pi p(\alpha) > y$
26. $\Phi \Vdash_\Pi p(\alpha) \geq x$ *and* $x \geq y$ *implies* $\Phi \Vdash_\Pi p(\alpha) \geq y$

27. $\Phi \Vdash_{\Pi} p(\alpha) \geq x$ *and* $x > y$ *implies* $\Phi \Vdash_{\Pi} p(\alpha) > y$

28. $\Phi \Vdash_{\Pi} p(\alpha) < x$ *and* $x \leq y$ *implies* $\Phi \Vdash_{\Pi} p(\alpha) < y$

29. $\Phi \Vdash_{\Pi} p(\alpha) < x$ *and* $x < y$ *implies* $\Phi \Vdash_{\Pi} p(\alpha) < y$

30. $\Phi \Vdash_{\Pi} p(\alpha) \leq x$ *and* $x \leq y$ *implies* $\Phi \Vdash_{\Pi} p(\alpha) \leq y$

31. $\Phi \Vdash_{\Pi} p(\alpha) \leq x$ *and* $x < y$ *implies* $\Phi \Vdash_{\Pi} p(\alpha) < y$

32. $\Phi \Vdash_{\Pi} p(\alpha) = x \wedge p(\alpha) = y$ *where* $x \neq y$ *iff* $\Phi \Vdash_{\Pi} \bot$

33. $\Phi \Vdash_{\Pi} p(\alpha \vee \beta) = v$ *iff* $\Phi \Vdash_{\Pi} \bigvee_{(x,y,z) \in \Pi_{=}^{v,(+,-)}} (p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z)$

34. $\Phi \Vdash_{\Pi} p(\alpha \wedge \beta) = v$ *iff* $\Phi \Vdash_{\Pi} \bigvee_{(x,y,z) \in \Pi_{=}^{v,(+,-)}} (p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \vee \beta) = z)$

35. $\Phi \Vdash_{\Pi} p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z$ *and* $(0 > x + y - z$ *or* $x + y - z > 1)$ *implies* $\Phi \Vdash_{\Pi} \bot$

36. $\Phi \Vdash_{\Pi} p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \vee \beta) = z$ *and* $(0 > x + y - z$ *or* $x + y - z > 1)$ *implies* $\Phi \Vdash_{\Pi} \bot$

Proof.

1. We prove that $\Phi \Vdash_{\Pi} p(\alpha) \geq 0$. We can observe that $\Phi \Vdash_{\Pi} \top$ by propositional rule P2. Thus, by combining the propositional rules and the basic rule B1, $\Phi \Vdash_{\Pi} p(\alpha) \geq 0$.

2. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) \leq 1$ similarly to previous point.

3. We can prove that $\Phi \Vdash_{\Pi} p(\top) = 1$ similarly to previous points.

4. We can prove that $\Phi \Vdash_{\Pi} p(\bot) = 0$ similarly to previous points.

5. We prove that $\Phi \Vdash_{\Pi} p(\alpha) > x$ implies $\Phi \Vdash_{\Pi} p(\alpha) \geq x$. Based on enumeration rule E1, $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi_{>}^{x}} p(\alpha) = v$ or $\Phi \Vdash_{\Pi} \bot$ if $\Pi_{>}^{x} = \varnothing$. If $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi_{>}^{x}} p(\alpha) = v$, then based on propositional rule P1, $\Phi \Vdash_{\Pi} (\bigvee_{v \in \Pi_{>}^{x}} p(\alpha) = v) \vee p(\alpha) = x$, which by E1 is equivalent to $\Phi \Vdash_{\Pi} p(\alpha) \geq x$. $\Phi \Vdash_{\Pi} \bot$, then based on Propositional rule P1 once more, $\Phi \Vdash_{\Pi} p(\alpha) \geq x$.

6. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) < x$ implies $\Phi \Vdash_{\Pi} p(\alpha) \leq x$ similarly to the previous point.

7. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) = x$ implies $\Phi \Vdash_{\Pi} p(\alpha) \geq x$ similarly to the previous point.

8. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) = x$ implies $\Phi \Vdash_{\Pi} p(\alpha) \leq x$ similarly to the previous point.

9. We prove that $\Phi \Vdash_{\Pi} p(\alpha) \neq x$ iff $\Phi \Vdash_{\Pi} p(\alpha) > x \vee p(\alpha) < x$. Based on rule E1, $\Phi \Vdash_{\Pi} p(\alpha) \neq x$ iff $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi_{\neq}^{x}} p(\alpha) = v$ (we observe that since Π is a reasonable restricted value set, then $\{0,1\} \subseteq \Pi$ and hence $\Pi_{\neq}^{x} \neq \varnothing$ by Proposition 4.8). It is easy to see that it is equivalent to either (a) $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi_{>}^{x}} p(\alpha) = v \vee \bigvee_{v \in \Pi_{<}^{x}} p(\alpha) = v$ if $\Pi_{>}^{x} \neq \varnothing$ and $\Pi_{<}^{x} \neq \varnothing$, or (b) $\Phi \Vdash_{\Pi} \bot \vee \bigvee_{v \in \Pi_{<}^{x}} p(\alpha) = v$ if $\Pi_{>}^{x} = \varnothing$ and $\Pi_{<}^{x} \neq \varnothing$, or (c) $\Phi \Vdash_{\Pi} \bot \vee \bigvee_{v \in \Pi_{>}^{x}} p(\alpha) = v$ if $\Pi_{>}^{x} \neq \varnothing$ and $\Pi_{<}^{x} = \varnothing$. We can observe that it cannot be the case that $\Pi_{>}^{x} = \Pi_{<}^{x} = \varnothing$, as we are dealing with a reasonable restricted value sets. Nevertheless, based on rule E1, this is equivalent to $\Phi \Vdash_{\Pi} p(\alpha) > x \vee p(\alpha) < x$. Therefore, $\Phi \Vdash_{\Pi} p(\alpha) \neq x$ iff $\Phi \Vdash_{\Pi} p(\alpha) > x \vee p(\alpha) < x$.

10. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) \geq x$ iff $\Phi \Vdash_{\Pi} p(\alpha) > x \vee p(\alpha) = x$ similarly to the previous point.

11. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) \leq x$ iff $\Phi \Vdash_{\Pi} p(\alpha) < x \vee p(\alpha) = x$ similarly to the previous point.

12. We prove that $\Phi \Vdash_{\Pi} p(\alpha) > x$ iff $\Phi \Vdash_{\Pi} p(\alpha) \geq x \wedge p(\alpha) \neq x$. As shown in the previous parts of this proof, $\Phi \Vdash_{\Pi} p(\alpha) > x$ implies $\Phi \Vdash_{\Pi} p(\alpha) \geq x$. Furthermore, $\{p(\alpha) > x\} \vdash p(\alpha) > x \vee p(\alpha) < x$, hence $\Phi \Vdash_{\Pi} p(\alpha) > x \vee p(\alpha) < x$ by P1. By the previous parts of this proof it thus holds that $\Phi \Vdash_{\Pi} p(\alpha) \neq x$. Consequently, we can use P1 again to show that if $\Phi \Vdash_{\Pi} p(\alpha) > x$ then $\Phi \Vdash_{\Pi} p(\alpha) \geq x \wedge p(\alpha) \neq x$.

   Let now $\Phi \Vdash_{\Pi} p(\alpha) \geq x \wedge p(\alpha) \neq x$. Thus, by the previous points of this proof, $\Phi \Vdash_{\Pi} (p(\alpha) > x \vee p(\alpha) = x) \wedge (p(\alpha) > x \vee p(\alpha) < x)$. We can use enumeration rule E1 and the propositional rule P1 to show that $\Phi \Vdash_{\Pi} p(\alpha) > x$. Hence, if $\Phi \Vdash_{\Pi} p(\alpha) \geq x \wedge p(\alpha) \neq x$ then $\Phi \Vdash_{\Pi} p(\alpha) > x$ and this concludes the proof.

13. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) < x$ iff $\Phi \Vdash_{\Pi} p(\alpha) \leq x \wedge p(\alpha) \neq x$ similarly to the previous point.

14. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) = x$ iff $\Phi \Vdash_{\Pi} p(\alpha) \geq x \wedge p(\alpha) \leq x$ similarly to the previous point.

15. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) > x$ iff $\Phi \Vdash_{\Pi} \neg(p(\alpha) \leq x)$ using enumeration rules E1 and E2.

16. We prove that $\Phi \Vdash_{\Pi} p(\alpha) < x$ iff $\Phi \Vdash_{\Pi} \neg(p(\alpha) \geq x)$ using enumeration rules E1 and E4.

17. We prove that $\Phi \Vdash_{\Pi} p(\alpha) \leq x$ iff $\Phi \Vdash_{\Pi} \neg(p(\alpha) > x)$ using enumeration rules E1 and E3.

18. We prove that $\Phi \Vdash_{\Pi} p(\alpha) \geq x$ iff $\Phi \Vdash_{\Pi} \neg(p(\alpha) < x)$ using enumeration rules E1 and E5.

19. We prove that $\Phi \Vdash_{\Pi} p(\alpha) = x$ iff $\Phi \Vdash_{\Pi} \neg(p(\alpha) \neq x)$. Using the propositional rules we can show that $\Phi \Vdash_{\Pi} p(\alpha) = x$ iff $\Phi \Vdash_{\Pi} \neg(\neg(p(\alpha) = x))$. By one of the previous points $\Phi \Vdash_{\Pi} \neg(\neg(p(\alpha) = x))$ iff $\Phi \Vdash_{\Pi} \neg(\neg(p(\alpha) \geq x \wedge p(\alpha) \leq x))$, which by the propositional rules and previous parts of this proof is equivalent to $\Phi \Vdash_{\Pi} \neg(p(\alpha) < x \vee p(\alpha) > x))$ and thus to $\Phi \Vdash_{\Pi} \neg(p(\alpha) \neq x)$.

20. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) \neq x$ iff $\Phi \Vdash_{\Pi} \neg(p(\alpha) = x)$ using the previous point and the propositional rules.

21. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) = x$ and $\{\alpha\} \vdash \beta$ implies $\Phi \Vdash_{\Pi} p(\beta) \geq x$ using the previously proved rule $\Phi \Vdash_{\Pi} p(\alpha) \geq x$ iff $\Phi \Vdash_{\Pi} p(\alpha) > x \vee p(\alpha) = x$, subject rule S2 and the propositional rules.

22. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) = x$ and $\{\beta\} \vdash \alpha$ implies $\Phi \Vdash_{\Pi} p(\beta) \leq x$ using the previously proved rule $\Phi \Vdash_{\Pi} p(\alpha) \leq x$ iff $\Phi \Vdash_{\Pi} p(\alpha) < x \vee p(\alpha) = x$, subject rule S3 and the propositional rules.

23. We now prove that if $\{\alpha\} \vdash \beta$ and $\{\beta\} \vdash \alpha$ then $\Phi \Vdash_{\Pi} p(\alpha) = x$ iff $\Phi \Vdash_{\Pi} p(\beta) = x$. From the previous points of this proof, if $\Phi \Vdash_{\Pi} p(\alpha) = x$ and $\{\alpha\} \vdash \beta$ then $\Phi \Vdash_{\Pi} p(\beta) \geq x$. Furthermore, if $\Phi \Vdash_{\Pi} p(\alpha) = x$ and $\{\beta\} \vdash \alpha$ then $\Phi \Vdash_{\Pi} p(\beta) \leq x$. Hence, we can use propositional rule P1 to show that if $\{\alpha\} \vdash \beta$ and $\{\beta\} \vdash \alpha$ and $\Phi \Vdash_{\Pi} p(\alpha) = x$, then $\Phi \Vdash_{\Pi} p(\beta) \geq x \wedge p(\beta) \leq x$, which by derivable rule 14 is equivalent to $\Phi \Vdash_{\Pi} p(\beta) = x$. The right to left direction can be proved in a similar fashion.

24. We prove that $\Phi \Vdash_{\Pi} p(\alpha) > x$ and $x \geq y$ implies $\Phi \Vdash_{\Pi} p(\alpha) > y$. Based on enumeration rule E1, $\Phi \Vdash_{\Pi} p(\alpha) > x$ iff $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi^x_>} p(\alpha) = v$ or $\Phi \Vdash_{\Pi} \perp$ when $\Pi^x_> = \varnothing$. If it is the latter case, then by using P1 we can show that if $\Phi \Vdash_{\Pi} \perp$ then $\Phi \Vdash_{\Pi} p(\alpha) > y$. Let us focus on the former case. Since $\Pi^x_> \subseteq \Pi^y_>$, it is easy to show that $\{\bigvee_{v \in \Pi^x_>} p(\alpha) = v\} \vdash \bigvee_{v \in \Pi^y_>} p(\alpha) = v$. Consequently, $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi^y_>} p(\alpha) = v$ and therefore $\Phi \Vdash_{\Pi} p(\alpha) > y$.

25. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) > x$ and $x > y$ implies $\Phi \Vdash_{\Pi} p(\alpha) > y$ similarly to the previous point. We note that $\Pi^x_> \subset \Pi^y_>$.

26. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) \geq x$ and $x \geq y$ implies $\Phi \Vdash_{\Pi} p(\alpha) \geq y$ similarly to the previous points.

27. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) \geq x$ and $x > y$ implies $\Phi \Vdash_{\Pi} p(\alpha) > y$. We note that $\Pi^x_\geq \subset \Pi^y_>$.

28. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) < x$ and $x \leq y$ implies $\Phi \Vdash_{\Pi} p(\alpha) < y$ similarly to the previous points.

29. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) < x$ and $x < y$ implies $\Phi \Vdash_{\Pi} p(\alpha) < y$ similarly to the previous points.

30. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) \leq x$ and $x \leq y$ implies $\Phi \Vdash_{\Pi} p(\alpha) \leq y$ similarly to the previous points.

31. We can prove that $\Phi \Vdash_{\Pi} p(\alpha) \leq x$ and $x < y$ implies $\Phi \Vdash_{\Pi} p(\alpha) < y$ similarly to the previous points.

32. We now prove that $\Phi \Vdash\models_{\Pi} p(\alpha) = x \wedge p(\alpha) = y$ where $x \neq y$ iff $\Phi \Vdash\models_{\Pi} \bot$. Using the propositional rules and derivable rule 19, $\Phi \Vdash\models_{\Pi} p(\alpha) = x \wedge p(\alpha) = y$ iff $\Phi \Vdash\models_{\Pi} p(\alpha) = x \wedge \neg(p(\alpha) \neq y)$. By enumeration rule E1 and the propositional rules, $\Phi \Vdash\models_{\Pi} p(\alpha) = x \wedge \neg(p(\alpha) \neq y)$ iff $\Phi \Vdash\models_{\Pi} p(\alpha) = x \wedge \neg(\bigvee_{v \in \Pi^{y}_{\neq}} p(\alpha) = v)$ (we note that since Π is a reasonable restricted value set, $\{0, 1\} \subseteq \Pi$ and $\Pi^{y}_{\neq} \neq \varnothing$ by Proposition 4.8). By De Morgan's laws, the right hand side formula is equivalent to $\Phi \Vdash\models_{\Pi} p(\alpha) = x \wedge (\bigwedge_{v \in \Pi^{y}_{\neq}} \neg(p(\alpha) = v))$. Assume that $\Pi^{y}_{\neq} = \{v_1, \ldots, v_k\}$. Hence, by associative laws, the above formula is equivalent to $\Phi \Vdash\models_{\Pi} p(\alpha) = x \wedge \neg(p(\alpha) = v_1) \ldots \wedge \neg(p(\alpha) = v_k)$. Given that $x \neq y$, $\exists v_i \in \bigwedge_{v \in \Pi^{y}_{\neq}}$ s.t. $v_i = x$. Thus, by negation laws, out formula is equivalent to $\Phi \Vdash\models_{\Pi} \bot$, which concludes our proof.

33. We show that $\Phi \Vdash_{\Pi} p(\alpha \vee \beta) = v$ iff $\Phi \Vdash_{\Pi} \bigvee_{(x,y,z) \in \Pi^{v,(+,-)}_{=}} (p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z)$. Let us consider the left to right direction and assume that $\Phi \Vdash_{\Pi} p(\alpha \vee \beta) = x$. Based on the probabilistic rule PR1, $\Phi \Vdash_{\Pi} \bigvee_{(\tau_1,\tau_2,\tau_3,\tau_4) \in \Pi^{0,(-,-,+)}_{=}} (p(\alpha \vee \beta) = \tau_1 \wedge p(\alpha) = \tau_2 \wedge p(\beta) = \tau_3 \wedge p(\alpha \wedge \beta) = \tau_4)$. Through the use of propositional rules we can show that $\Phi \Vdash_{\Pi} p(\alpha \vee \beta) = v$ iff $\Phi \Vdash_{\Pi} p(\alpha \vee \beta) = v \wedge \bigvee_{(\tau_1,\tau_2,\tau_3,\tau_4) \in \Pi^{0,(-,-,+)}_{=}} (p(\alpha \vee \beta) = \tau_1 \wedge p(\alpha) = \tau_2 \wedge p(\beta) = \tau_3 \wedge p(\alpha \wedge \beta) = \tau_4))$. Through the use of propositional rule (in particular, distributive and identity laws) and derivable rule 32, we can observe that the above formula is equivalent to equivalent to $\Phi \Vdash_{\Pi} \bigvee_{(x,v_2,v_3,v_4) \in \Pi^{0,(-,-,+)}_{=}} (p(\alpha \vee \beta) = x \wedge p(\alpha) = \tau_2 \wedge p(\beta) = \tau_3 \wedge p(\alpha \wedge \beta) = \tau_4)$. This can be further shown to be equivalent to $\Phi \Vdash_{\Pi} p(\alpha \vee \beta) = x \wedge \bigvee_{(v_1,v_2,v_3) \in \Pi^{x,(+,-)}_{=}} (p(\alpha) = v_1 \wedge p(\beta) = v_2 \wedge p(\alpha \wedge \beta) = v_3)$. Hence, by P1, $\Phi \Vdash_{\Pi} \bigvee_{(v_1,v_2,v_3) \in \Pi^{x,(+,-)}_{=}} (p(\alpha) = v_1 \wedge p(\beta) = v_2 \wedge p(\alpha \wedge \beta) = v_3)$. The right to left direction can be proved in a similar fashion.

34. We can show that $\Phi \Vdash_{\Pi} p(\alpha \wedge \beta) = x$ iff $\bigvee_{(x,y,z) \in \Pi^{v,(+,-)}_{=}} (p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \vee \beta) = z)$ in the same way as the previous point of this proof.

35. We will now show that $\Phi \Vdash_{\Pi} p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z$ and $(0 > x + y - z$ or $x + y - z > 1)$ implies $\Phi \Vdash_{\Pi} \bot$. Let us consider the first case. Based on derivable rule 1, $\Phi \Vdash_{\Pi} p(\alpha \wedge \beta) \geq 0$, which through the use of enumeration rule E1 is equivalent to $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi} p(\alpha \wedge \beta) = v$ (we observe that based on Proposition 4.8, $\Pi^{0}_{\geq} \neq \varnothing$). Hence, using the previous points of this proposition, we can show that this is equivalent to $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi} (\bigvee_{(x',y',z') \in \Pi^{v,(+,-)}_{=}} p(\alpha) = x' \wedge p(\beta) = y' p(\alpha \wedge \beta) = z')$. We can observe that $\bigcup_{v \in \Pi} \Pi^{v,(+,-)}_{=} \subseteq \Pi^{0,(+,-)}_{\geq}$. Thus, through the use of propositional rule, it follows that $\Phi \Vdash_{\Pi} \bigvee_{(x',y',z') \in \Pi^{0,(+,-)}_{\geq}} p(\alpha) = x' \wedge p(\beta) = y' p(\alpha \wedge \beta) = z'$. Since $\Phi \Vdash_{\Pi} p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z$ and $0 > x + y - z$ by assumption, then based on propositional rule, $\Phi \Vdash_{\Pi} p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z \wedge \bigvee_{(x',y',z') \in \Pi^{0,(+,-)}_{\geq}} p(\alpha) = x' \wedge p(\beta) = y' p(\alpha \wedge \beta) = z'$. We can observe that $(x, y, z) \notin \Pi^{0,(+,-)}_{\geq}$. Thus, for every $(x', y', z') \in \Pi^{0,(+,-)}_{\geq}$, either $x \neq x'$, or $y \neq y'$, or $z \neq z'$. We can therefore use derivable rule 32 along with propositional rules to show that $\Phi \Vdash_{\Pi} \bot$.

    Let us now consider the second case. Based on derivable rule 1, $\Phi \Vdash_{\Pi} p(\alpha \wedge \beta) \leq 1$, which through the use of enumeration rule E1 is equivalent to $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi} p(\alpha \wedge \beta) = v$. We can repeat the previous reasoning to show that this implies that $\Phi \Vdash_{\Pi} \bigvee_{(x',y',z') \in \Pi^{1,(+,-)}_{\leq}} p(\alpha) = x' \wedge p(\beta) = y' p(\alpha \wedge \beta) = z'$. Since $\Phi \Vdash_{\Pi} p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z$ and $x + y - z > 1$ by assumption, then based on propositional rule, $\Phi \Vdash_{\Pi} p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \wedge \beta) = z \wedge \bigvee_{(x',y',z') \in \Pi^{1,(+,-)}_{\leq}} p(\alpha) = x' \wedge p(\beta) = y' p(\alpha \wedge \beta) = z'$. We can observe that $(x, y, z) \notin \Pi^{1,(+,-)}_{\leq}$. Thus, for every $(x', y', z') \in \Pi^{0,(+,-)}_{\geq}$, either $x \neq x'$, or $y \neq y'$, or $z \neq z'$. We can therefore use derivable rule 32 along with propositional rules to show that $\Phi \Vdash_{\Pi} \bot$.

36. We can show that $\Phi \Vdash_{\Pi} p(\alpha) = x \wedge p(\beta) = y \wedge p(\alpha \vee \beta) = z$ and $(0 > x + y - z$ or $x + y - z > 1)$ implies $\Phi \Vdash_{\Pi} \bot$ similarly to the previous point of this proof.

□

Proposition 4.21. *Let Π be a restricted value set. For $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$, and $\psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$, if $\Phi \Vdash_{\Pi} \psi$ then $\Phi \Vdash\models_{\Pi} \psi$.*

Proof. We can show each proof rule is sound. We first consider the **basic rules**:

- Consider proof rule 1. We need to show that $\Phi \Vdash_{\Pi} p(\alpha) \geq 0$ iff $\Phi \Vdash_{\Pi} \top$. We can observe that $\mathsf{Sat}(\top, \Pi) = \mathsf{Dist}(\Pi)$. Furthermore, by definition, $\mathsf{Sat}(p(\alpha) \geq 0, \Pi) = \{P' \in \mathsf{Dist}(\mathcal{G}) \mid P'(\alpha) \geq 0\} \cap \mathsf{Dist}(\Pi)$. It is easy to see that $\{P' \in \mathsf{Dist}(\mathcal{G}) \mid P'(\alpha) \geq 0\} = \mathsf{Dist}(\mathcal{G})$ for any α. Since $\mathsf{Dist}(\Pi) \subseteq \mathsf{Dist}(\mathcal{G})$, $\mathsf{Sat}(p(\alpha) \geq 0, \Pi) = \mathsf{Dist}(\Pi) = \mathsf{Sat}(\top, \Pi)$. Thus, we can show that $\Phi \Vdash_{\Pi} p(\alpha) \geq 0$ iff $\Phi \Vdash_{\Pi} \top$.

- Proof rules 2 to 4 can be proved in a similar fashion.

We now consider the **enumeration rules**:

- Consider proof rule 1. We need to show that $\Phi \Vdash_{\Pi} p(\alpha)\#x$ iff ($\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi^x_{\#}} p(\alpha) = v$ if $\Pi^x_{\#} \neq \varnothing$ and $\Phi \Vdash_{\Pi} \perp$ otherwise). We first consider $\#$ being $>$ and start with the case where $\Pi^x_{>} = \varnothing$. Since Π is a reasonable restricted value set, $\{0, 1\} \subseteq \Pi$, and therefore $\Pi^x_{>} = \varnothing$ iff $x = 1$. It is easy to see that $\mathsf{Sat}(p(\alpha) > 1, \Pi) = \varnothing = \mathsf{Sat}(\perp, \Pi)$. Hence, the rule is sound in this case. Now consider $\Pi^x_{>} \neq \varnothing$. For every $P' \in \mathsf{Sat}(p(\alpha) > x, \Pi)$, $P'(\alpha) \in \Pi^x_{>}$, and for every $v \in \Pi^x_{>}$, there is a $P^* \in \mathsf{Sat}(p(\alpha) > x, \Pi)$ s.t. $P^*(\alpha) = v$. Therefore, $\mathsf{Sat}(p(\alpha) > x, \Pi) = \mathsf{Sat}(\bigvee_{v \in \Pi^x_{>}} p(\alpha) = v, \Pi)$, and $\Phi \Vdash_{\Pi} p(\alpha)\#x$ iff $\Phi \Vdash_{\Pi} \bigvee_{v \in \Pi^x_{>}} p(\alpha) = v$. The results for other operators can be obtained in a similar fashion.

- Consider proof rule 2. We need to show that $\Phi \Vdash_{\Pi} p(\alpha) > x$ iff $\Phi \Vdash_{\Pi} \neg(\bigvee_{v \in \Pi^x_{\leq}} p(\alpha) = v)$. For every $P' \in \mathsf{Sat}(p(\alpha) > x, \Pi)$, $P'(\alpha) \in \Pi^x_{>}$, and for every $v \in \Pi^x_{>}$, there is a $P^* \in \mathsf{Sat}(p(\alpha) > x, \Pi)$, $P^*(\alpha) = v$. Therefore, it holds that $\mathsf{Sat}(p(\alpha) > x, \Pi) = \mathsf{Sat}(\top, \Pi) \setminus \mathsf{Sat}(\bigvee_{v \in \Pi^x_{\leq}} p(\alpha) = v, \Pi) = \mathsf{Sat}(\neg(\bigvee_{v \in \Pi^x_{\leq}} p(\alpha) = v), \Pi)$. Hence, $\Phi \Vdash_{\Pi} p(\alpha) > x$ iff $\Phi \Vdash_{\Pi} \neg(\bigvee_{v \in \Pi^x_{\leq}} p(\alpha) = v)$.

- Soundness for the proof rules 3 to 5 is obtained in the same way as for rule 2. For rule 3, we only observe that if $\Pi^x_{<} = \varnothing$, then based on the properties of Π, $x = 0$ and $p(\alpha) \geq 0$ is a tautology (and is therefore equivalent to $\neg\perp$). Similar observations can be made for rule 5, i.e. if $\Pi^x_{>} = \varnothing$, then based on the properties of Π, $x = 1$.

We now consider the **subject rules**:

- Consider proof rule 1. We need to show that if $\Phi \Vdash_{\Pi} p(\alpha) > x$ and $\{\alpha\} \vdash \beta$ then $\Phi \Vdash_{\Pi} p(\beta) > x$. Assume $\Phi \Vdash_{\Pi} p(\alpha) > x$ and $\{\alpha\} \vdash \beta$. Therefore, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Sat}(p(\alpha) > x, \Pi)$. Since $\{\alpha\} \vdash \beta$, then for any probability distribution $P \in \mathsf{Sat}(p(\alpha) > x, \Pi)$, $P(\alpha) \leq P(\beta)$. Consequently, $P(\beta) > x$ for any such P, and it holds that $\mathsf{Sat}(p(\alpha) > x, \Pi) \subseteq \mathsf{Sat}(p(\beta) > x, \Pi)$. Thus, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Sat}(P(\beta) > x, \Pi)$ and $\Phi \Vdash_{\Pi} p(\beta) > x$.

- Remaining subject rules can be proved in a similar fashion.

We now consider the **probabilistic rule**. We can observe that for any probability distribution P and terms $\alpha, \beta \in \mathsf{Terms}(\mathcal{G})$, $P(\alpha \vee \beta) = P(\alpha) + P(\beta) - P(\alpha \wedge \beta)$. Therefore, $P(\alpha \vee \beta) - P(\alpha) - P(\beta) + P(\alpha \wedge \beta) = 0$. This can be easily checked by analyzing the definition of the probability of a term (Definition 3.2). Hence, assuming that $P(\alpha \vee \beta) = x$, $P(\alpha) = y$, $P(\beta) = z$ and $P(\alpha \wedge \beta) = v$, it holds that $x - y - z + v = 0$. Consequently, $(x, y, z, v) \in \Pi^{0,(-,-,+)}_{=}$. Thus, it holds that $\mathsf{Sat}(\bigvee_{(x,y,z,v) \in \Pi^{0,(-,-,+)}_{=}} (p(\alpha \vee \beta) = x \wedge p(\alpha) = y \wedge p(\beta) = z \wedge p(\alpha \wedge \beta) = v), \Pi) = \mathsf{Dist}(\mathcal{G}, \Pi)$. Since for any set of probabilistic formulae Φ, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Dist}(\mathcal{G}, \Pi)$, it holds that $\Phi \Vdash_{\Pi} \bigvee_{(x,y,z,v) \in \Pi^{0,(-,-,+)}_{=}} (p(\alpha \vee \beta) = x \wedge p(\alpha) = y \wedge p(\beta) = z \wedge p(\alpha \wedge \beta) = v)$.

We now consider the **propositional rules**.

- Assume $\Phi \Vdash_{\Pi} \alpha_1, \ldots \Phi \Vdash_{\Pi} \alpha_n$ and $\{\alpha_1, \ldots, \alpha_n\} \vdash \beta$. So for each $i \in \{1, \ldots, n\}$, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Sat}(\alpha_i, \Pi)$. Furthermore, based on the definition of Sat, if $\{\alpha_1, \ldots, \alpha_n\} \vdash \beta$, then it holds that $\mathsf{Sat}(\{\alpha_1, \ldots, \alpha_n\}) \subseteq \mathsf{Sat}(\beta)$. Hence, $\mathsf{Sat}(\{\alpha_1, \ldots, \alpha_n\}, \Pi) \subseteq \mathsf{Sat}(\beta, \Pi)$ as well. Therefore, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Sat}(\beta, \Pi)$. Hence, $\Phi \Vdash_{\Pi} \beta$.

- Assume $\Phi \vdash \varphi$. Consequently, $\mathsf{Sat}(\Phi) \subseteq \mathsf{Sat}(\varphi)$. Hence, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Sat}(\varphi, \Pi)$ as well. Therefore, $\Phi \Vdash_{\Pi} \varphi$.

This shows that our system is indeed sound. $\square$

Proposition 4.22. *Let $\# \in \{=, \neq, \geq, \leq, >, <\}$ be the set of inequality relationships, let Π be a reasonable restricted value set, and let $\Pi^x_\# = \{y \in \Pi \mid x\#y\}$ be the subset obtained according to the value x and relationship $\#$. Let $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \ldots, c_j\}$ be the collection of all argument complete propositional formulae for $\mathcal{G}$ and $T^{\Pi}_{v,k} = \Pi^{v,(+,\ldots,+)}_{=}$ s.t. the length of $(+,\ldots,+)$ is $k-1$ be the collection of k–tuples of values from Π that sum up to $v \in \Pi$. The following hold, where $\vdash$ is propositional consequence relation, $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$, $\phi, \psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$ and $x \in \Pi$. The following hold:*

1. *for consistent formulae $\alpha_1, \ldots \alpha_m \in \mathsf{Terms}(\mathcal{G})$, if for all $1 \leq i, j \leq m$ s.t. $i \neq j$ it holds that $\alpha_i \wedge \alpha_j \vdash \bot$, then $\Phi \Vdash_\Pi p(\alpha_1 \vee \ldots \vee \alpha_m) = x$ iff $\Phi \Vdash_\Pi \bigvee_{(\tau_1,\ldots,\tau_m) \in T^{\Pi}_{x,m}} (p(\alpha_1) = \tau_1 \wedge \ldots \wedge p(\alpha_m) = \tau_m)$*

2. *$\Phi \Vdash_\Pi \bigvee_{(\tau_1,\ldots,\tau_j) \in T^{\Pi}_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$*

3. *$\Phi \Vdash_\Pi \psi$ iff $\Phi \Vdash_\Pi \varphi$, where φ is the distribution disjunctive normal form of ψ*

Proof. 1. We prove this property by induction. We show the cases for $m = 1$ and, for clarity, $m = 2$, and then prove that if the property holds for $m = k$, then it holds for $m = k + 1$ as well.

- Let us start with $m = 1$. Then $T^{\Pi}_{v,1} = \Pi^{v,()}_{=}$ and $\Phi \Vdash_\Pi p(\alpha_1) = v$ iff $\Phi \Vdash_\Pi p(\alpha_1) = v$, which is clearly true.
- Let $m = 2$. Let us consider the left to right direction. By derivable rule 33, $\Phi \Vdash_\Pi p(\alpha_1 \vee \alpha_2) = v$ iff $\Phi \Vdash_\Pi \bigvee_{(x,y,z) \in \Pi^{v,(+,-)}_{=}} (p(\alpha_1) = x \wedge p(\alpha_2) = y \wedge p(\alpha_1 \wedge \alpha_2) = z)$. Since $\alpha_1 \wedge \alpha_2 \vdash \bot$ and obviously, $\bot \vdash \alpha_1 \wedge \alpha_2$, we can use derivable rule 23 to show that $p(\alpha_1 \wedge \alpha_2) = z$ is equivalent to $p(\bot) = z$. Based on this, the propositional rule P1 and the 4th basic rule, we can therefore show that $\Phi \Vdash_\Pi \bigvee_{(x,y,z) \in \Pi^{v,(+,-)}_{=}} (p(\alpha_1) = x \wedge p(\alpha_2) = y \wedge p(\alpha_1 \wedge \alpha_2) = z)$ iff $\Phi \Vdash_\Pi \bigvee_{(x,y,z) \in \Pi^{v,(+,-)}_{=}} (p(\alpha_1) = x \wedge p(\alpha_2) = y \wedge p(\bot) = z \wedge p(\bot) = 0)$. We can also use derivable rule 32 to show that when $z \neq 0$, $p(\bot) = z \wedge p(\bot) = 0$ is equivalent to $\bot$. Hence, for any $(x, y, z) \in \Pi^{v,(+,-)}_{=}$ s.t. $z \neq 0$, the conjunctive clause $(p(\alpha_1) = x \wedge p(\alpha_2) = y \wedge p(\alpha_1 \wedge \alpha_2) = z)$ is also equivalent to $\bot$ through the use of propositional rule P1 (in particular, domination law[10]). Thus, using the propositional rule P1 once more, our formula is equivalent to $\Phi \Vdash_\Pi \bigvee_{(x,y,0) \in \Pi^{v,(+,-)}_{=}} (p(\alpha_1) = x \wedge p(\alpha_2) = y \wedge p(\alpha_1 \wedge \alpha_2) = 0)$. Since $p(\bot) = 0$ is a tautology, then so is $p(\alpha_1 \wedge \alpha_2) = 0$, and again by P1 (in particular, identity law) we can show that our formula is equivalent to $\Phi \Vdash_\Pi \bigvee_{(x,y,0) \in \Pi^{v,(+,-)}_{=}} (p(\alpha_1) = x \wedge p(\alpha_2) = y)$, which is exactly $\Phi \Vdash_\Pi \bigvee_{(\tau_1,\tau_2) \in T^{\Pi}_{v,2}} (p(\alpha_1) = \tau_1 \wedge p(\alpha_2) = \tau_2)$. The right to left direction can be shown in a similar fashion.
- Assume that for $m = k$ our property is true, i.e. if our assumptions hold, then $\Phi \Vdash_\Pi p(\alpha_1 \vee \ldots \vee \alpha_k) = v$ iff $\Phi \Vdash_\Pi \bigvee_{(\tau_1,\ldots,\tau_k) \in T^{\Pi}_{v,k}} (p(\alpha_1) = \tau_1 \wedge \ldots \wedge p(\alpha_k) = \tau_k)$. We need to show that for $m = k + 1$, if our assumptions hold, then $\Phi \Vdash_\Pi p(\alpha_1 \vee \ldots \vee \alpha_{k+1}) = v$ iff $\Phi \Vdash_\Pi \bigvee_{(\tau_1,\ldots,\tau_{k+1}) \in T^{\Pi}_{v,k+1}} (p(\alpha_1) = \tau_1 \wedge \ldots \wedge p(\alpha_{k+1}) = \tau_{k+1})$. We focus on the left to right direction first. By derivable rule 33, $\Phi \Vdash_\Pi p(\alpha_1 \vee \ldots \vee \alpha_{k+1}) = v$ iff $\Phi \Vdash_\Pi \bigvee_{(x,y,z) \in \Pi^{v,(+,-)}_{=}} (p(\alpha_1 \vee \ldots \vee \alpha_k) = x \wedge p(\alpha_{k+1}) = y \wedge p((\alpha_1 \vee \ldots \vee \alpha_k) \wedge \alpha_{k+1}) = z)$. Given that for $i \in \{1, \ldots, k\}$, $\alpha_{k+1} \wedge \alpha_i \vdash \bot$, then based on the 4th derivable rule and propositional rule P1 we can observe that for any $(x, y, z) \in \Pi^{v,(+,-)}_{=}$ s.t. $z \neq 0$, $(p(\alpha_1 \vee \ldots \vee \alpha_k) = x \wedge p(\alpha_{k+1}) = y \wedge p((\alpha_1 \vee \ldots \vee \alpha_k) \wedge \alpha_{k+1}) = z)$ is equivalent to $\bot$. Thus, using the propositional rule P1 once more, our formula is equivalent to $\Phi \Vdash_\Pi \bigvee_{(x,y,0) \in \Pi^{v,(+,-)}_{=}} (p(\alpha_1 \vee \ldots \vee \alpha_k) = x \wedge p(\alpha_{k+1}) = y \wedge p((\alpha_1 \vee \ldots \vee \alpha_k) \wedge \alpha_{k+1}) = 0)$. Since $(\alpha_1 \vee \ldots \vee \alpha_k) \wedge \alpha_{k+1})$ is equivalent to $\bot$, then $p((\alpha_1 \vee \ldots \vee \alpha_k) \wedge \alpha_{k+1}) = 0$ is a tautology. Thus, again by P1, our formula is equivalent to $\Phi \Vdash_\Pi \bigvee_{(x,y,0) \in \Pi^{v,(+,-)}_{=}} (p(\alpha_1 \vee \ldots \vee \alpha_k) = x \wedge p(\alpha_{k+1}) = y)$, which is the same as $\Phi \Vdash_\Pi \bigvee_{(\tau_1,\tau_2) \in T^{\Pi}_{v,2}} (p(\alpha_1 \vee \ldots \vee \alpha_k) = \tau_1 \wedge p(\alpha_{k+1}) = \tau_2)$. Since the proposition holds for $m = k$ by assumption, then our formula is equivalent to:

$$\Phi \Vdash_\Pi \bigvee_{(\tau_1,\tau_2) \in T^{\Pi}_{v,2}} ((\bigvee_{(\tau'_1,\ldots,\tau'_k) \in T^{\Pi}_{\tau_1,k}} (p(\alpha_1) = \tau'_1 \wedge \ldots \wedge p(\alpha_k) = \tau'_k)) \wedge p(\alpha_{k+1}) = \tau_2)$$

[10] Recall that the domination logical equivalence law states that $q \vee \top \equiv \top$ and $q \wedge \bot \equiv \bot$.

Now, as Π is finite, then so is $T^{\Pi}_{v,2}$. Let us assume that $T^{\Pi}_{v,2}$ is of the form $\{(\tau_{1,1}, \tau_{2,1}), (\tau_{1,2}, \tau_{2,2}), \ldots, (\tau_{1,p}, \tau_{2,p})\}$. Our formula can therefore be rewritten as

$$\Phi \Vdash_{\Pi} ((\bigvee_{(\tau'_{1,1},\ldots,\tau'_{k,1}) \in T^{\Pi}_{\tau_{1,1},k}} (p(\alpha_1) = \tau'_{1,1} \wedge \ldots \wedge p(\alpha_k) = \tau'_{k,1})) \wedge p(\alpha_{k+1}) = \tau_{2,1})$$
$$\vee \ldots \vee$$
$$((\bigvee_{(\tau'_{1,p},\ldots,\tau'_{k,p}) \in T^{\Pi}_{\tau_{1,p},k}} (p(\alpha_1) = \tau'_{1,p} \wedge \ldots \wedge p(\alpha_k) = \tau'_{k,p})) \wedge p(\alpha_{k+1}) = \tau_{2,p})$$

Now, as Π is finite, then so is every $T^{\Pi}_{\tau_{1,i},k}$ for $i \in \{1, \ldots, p\}$. Assume that $|T^{\Pi}_{\tau_{1,i},k}| = l_i$ and therefore that every $T^{\Pi}_{\tau_{1,i},k}$ is of the form $\{({\tau'}^{1}_{1,i}, \ldots, {\tau'}^{1}_{k,i}), \ldots, ({\tau'}^{l_i}_{1,i}, \ldots, {\tau'}^{l_i}_{k,i})\}$. Thus, our formula can again be rewritten as:

$$\Phi \Vdash_{\Pi} (((p(\alpha_1) = {\tau'}^{1}_{1,1} \wedge \ldots \wedge p(\alpha_k) = {\tau'}^{1}_{k,1}) \vee \ldots \vee (p(\alpha_1) = {\tau'}^{l_1}_{1,1} \wedge \ldots \wedge p(\alpha_k) = {\tau'}^{l_1}_{k,1}))$$
$$\wedge\, p(\alpha_{k+1}) = \tau_{2,1})$$
$$\vee \ldots \vee$$
$$(((p(\alpha_1) = {\tau'}^{1}_{1,1} \wedge \ldots \wedge p(\alpha_k) = {\tau'}^{1}_{k,1}) \vee \ldots \vee (p(\alpha_1) = {\tau'}^{l_p}_{1,p} \wedge \ldots \wedge p(\alpha_k) = {\tau'}^{l_p}_{k,p}))$$
$$\wedge\, p(\alpha_{k+1}) = \tau_{2,p})$$

which by propositional rule P1 is equivalent to

$$\Phi \Vdash_{\Pi} (p(\alpha_1) = {\tau'}^{1}_{1,1} \wedge \ldots \wedge p(\alpha_k) = {\tau'}^{1}_{k,1} \wedge p(\alpha_{k+1}) = \tau_{2,1})$$
$$\vee \ldots \vee$$
$$(p(\alpha_1) = {\tau'}^{l_1}_{1,1} \wedge \ldots \wedge p(\alpha_k) = {\tau'}^{l_1}_{k,1}) \wedge p(\alpha_{k+1}) = \tau_{2,1})$$
$$\vee \ldots \vee$$
$$(p(\alpha_1) = {\tau'}^{1}_{1,1} \wedge \ldots \wedge p(\alpha_k) = {\tau'}^{1}_{k,1} \wedge p(\alpha_{k+1}) = \tau_{2,p})$$
$$\vee \ldots \vee$$
$$(p(\alpha_1) = {\tau'}^{l_p}_{1,p} \wedge \ldots \wedge p(\alpha_k) = {\tau'}^{l_p}_{k,p} \wedge p(\alpha_{k+1}) = \tau_{2,p})$$

We can observe that every conjunctive clause we obtain is in fact a combinations of values from $T^{\Pi}_{v,k+1}$. Furthermore, if $(v_1, \ldots, v_{k+1}) \in T^{\Pi}_{v,k+1}$ then $(v_1 + \ldots + v_k, v_{k+1}) \in T^{\Pi}_{v,2}$ – since $v \in [0,1]$ and we are dealing with addition only, then $v_1 + \ldots + v_k \in \Pi$ as well. Thus, for every combination from $T^{\Pi}_{v,k+1}$ we can also find a conjunctive clause associated with it, and our formula is in fact equivalent to $\Phi \Vdash_{\Pi} \bigvee_{(\tau''_1,\ldots,\tau''_{k+1}) \in T^{\Pi}_{v,k+1}} (p(\alpha_1) = \tau''_1 \wedge \ldots \wedge p(\alpha_{k+1}) = \tau''_{k+1})$. Since all of our transformation were using syntactical equivalences, the right to left direction of our proof can be shown in a similar fashion.

2. Based on derivable rule 3, $\Phi \Vdash_{\Pi} p(\top) = 1$. We can observe that $c_1 \vee \ldots c_j \vdash \top$ and $\top \vdash c_1 \vee \ldots c_j$. Furthermore, every c_i is consistent and for every i, k s.t. $i \neq k$, $c_i \wedge c_k \vdash \bot$. Thus, $\Phi \Vdash_{\Pi} p(c_1 \vee \ldots c_j) = 1$, and by the previous point of this proof, $\Phi \Vdash_{\Pi} \bigvee_{(\tau_1,\ldots,\tau_j) \in T^{\Pi}_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$.

3. For the purpose of this proof, we introduce a shorthand $\varphi \approx \varphi'$ stating that the two formulae have the same syntactical features under commutativity and associativity (so, for example, $(\mathtt{A} \vee \mathtt{B}) \wedge \mathtt{C} \approx (\mathtt{A} \vee \mathtt{B}) \wedge \mathtt{C}$, $(\mathtt{A} \vee \mathtt{B}) \wedge \mathtt{C} \approx (\mathtt{B} \vee \mathtt{A}) \wedge \mathtt{C}$, but $(\mathtt{A} \vee \mathtt{B}) \wedge \mathtt{C} \not\approx (\mathtt{A} \wedge \mathtt{C}) \vee (\mathtt{B} \wedge \mathtt{C})$).

   Let $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \ldots, c_j\}$ be the collection of all argument complete terms for $\mathcal{G}$. First of all, we can observe that for any term $\alpha \in \mathsf{Terms}(\mathcal{G})$, we can find an equivalent formula that is either $\bot$

if α is inconsistent, or which is equivalent to $c'_1 \vee \ldots \vee c'_k$, where $C' = \{c'_1, \ldots, c'_k\} \subseteq \mathsf{AComplete}(\mathcal{G})$ is a nonempty collection of argument complete terms. This form of α is easily found by constructing the (propositional disjunctive normal form) DNF using the truth table method. Let α' be this form of α.

We start by considering $\psi : p(\alpha) = x$. Since $\{\alpha\} \vdash \alpha'$ and $\{\alpha'\} \vdash \alpha$, then by derivable rule 23 it holds that $\Phi \Vdash_\Pi p(\alpha) = x$ iff $\Phi \Vdash_\Pi p(\alpha') = x$. We can now observe that α' is either a propositional contradiction ($\bot$), tautology ($c_1 \vee \ldots \vee c_j$), or neither. Under various values x, the probabilistic atom $p(\alpha') = x$ can become or remain a contradiction, tautology, or neither. We can therefore distinguish the following cases:

(a) $\alpha' : c_1 \vee \ldots \vee c_j$ and $x = 1$

(b) $\alpha' : c_1 \vee \ldots \vee c_j$ and $x \neq 1$

(c) $\alpha' : \bot$ and $x = 0$

(d) $\alpha' : \bot$ and $x \neq 0$

(e) remaining cases, i.e. $\alpha \neq \bot$ and $\alpha \neq c_1 \vee \ldots \vee c_j$

Let us start with the most complicated, last point (i.e. point e). Since all argument complete formulae are consistent and for each two different formulae c'_i, c'_l it holds that $c'_i \wedge c'_l \vdash \bot$, then by using the previous points of this proposition, we can show the following:

$$\Phi \Vdash_\Pi p(\alpha') = x \text{ iff } \Phi \Vdash_\Pi \bigvee_{(\tau'_1,\ldots,\tau'_k)\in T^\Pi_{x,k}} (p(c'_1) = \tau'_1 \wedge \ldots \wedge p(c'_k) = \tau'_k)$$

Let $D = \{d'_1, \ldots, d'_f\} = \mathsf{AComplete}(\mathcal{G}) \setminus C'$ be the collection of argument complete formulae not appearing in α'. If $D = \emptyset$, then all of the argument complete terms appear in our formula, which means that $\alpha' : c_1 \vee \ldots \vee c_j$. Consequently, depending on the value of x, please see the analysis of points a and b which is explained below. If $D \neq \emptyset$, then we can observe that every conjunctive clause in our formula is a partial description of (possibly more than one) probability distribution. We can therefore use the second point of this proposition, which in fact enumerates all possible restricted probability distribution on $\mathcal{G}$, to "complete" our partial descriptions. Using the propositional rules and the second point of this proposition, we can show that

$$\Phi \Vdash_\Pi \bigvee_{(\tau'_1,\ldots,\tau'_k)\in T^\Pi_{x,k}} (p(c'_1) = \tau'_1 \wedge \ldots \wedge p(c'_k) = \tau'_k) \text{ iff}$$

$$\Phi \Vdash_\Pi (\bigvee_{(\tau'_1,\ldots,\tau'_k)\in T^\Pi_{x,k}} (p(c'_1) = \tau'_1 \wedge \ldots \wedge p(c'_k) = \tau'_k)) \wedge (\bigvee_{(\tau_1,\ldots,\tau_j)\in T^\Pi_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j))$$

Let $V = \{(\tau_1, \ldots, \tau_j) \in T^\Pi_{1,j} \mid f((\tau_1, \ldots, \tau_j)) \in T^\Pi_{x,k}\}$, where $f((\tau_1, \ldots, \tau_j))$ returns a sub–tuple $(\tau'_1, \ldots, \tau'_k)$ of $(\tau_1, \ldots, \tau_j)$ s.t. if $c'_g \approx c_h$ then $\tau'_g = \tau_h$, be a collection of tuples in $T^\Pi_{1,j}$ that preserve the assignments from $T^\Pi_{x,k}$. Using the propositional rules, the second formula can be equivalently rewritten as

$$\Phi \Vdash_\Pi \bigvee_{(\tau_1,\ldots,\tau_j)\in V} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_k) = \tau_j)$$

if $V \neq \emptyset$ and as $\Phi \Vdash_\Pi \bot$ otherwise. We have therefore obtained an epistemic formula in (propositional) DNF s.t. it is either $\bot$, or every conjunctive clause is built out of atoms of the form $p(\beta) = x$ where β is an argument complete term and such that for every argument complete term β' there is a probabilistic atom $p(\beta') = x'$ in the clause. Let us refer to this formula as ψ'. Based on the presented procedure and the fact that our system is sound (see Proposition 4.21), we can show that $\mathsf{Sat}(\psi, \Pi) = \mathsf{Sat}(\psi', \Pi)$. Hence, if $\psi' : \bot$, then $\mathsf{Sat}(\psi, \Pi) = \mathsf{Sat}(\psi' \Pi) = \emptyset$, and it is easy to show that ψ' is indeed the DDNF of ψ. Therefore, let us focus on the case where ψ' is not $\bot$, and enumerate the conjunctive clauses as $\psi'_1, \ldots, \psi'_p$. Let $\varphi : \varphi_1 \vee \ldots \vee \varphi_r$ be a DDNF of ψ. We will now show that ψ' and

φ are equivalent under commutativity. By Proposition 4.18, it holds that $\mathsf{Sat}(\psi', \Pi) = \mathsf{Sat}(\varphi, \Pi)$. Hence, by the properties of Sat, $\mathsf{Sat}(\psi'_1, \Pi) \cup \ldots \cup \mathsf{Sat}(\psi'_p, \Pi) = \mathsf{Sat}(\varphi_1, \Pi) \cup \ldots \cup \mathsf{Sat}(\varphi_r, \Pi)$. We can observe that every $\psi'_i \in \{\psi'_1, \ldots, \psi'_p\}$ and $\varphi_s \in \{\varphi_1 \vee \ldots \vee \varphi_r\}$ is in fact an epistemic formula associated with a single unique probability distribution. Consequently, for every ψ'_i we can find a formula φ_s s.t. $\mathsf{Sat}(\psi'_i, \Pi) = \mathsf{Sat}(\varphi_s, \Pi)$ and vice versa. Given the form of these formulae, it therefore has to follow that $\psi'_i \vdash \varphi_s$ and $\varphi_s \vdash \psi'_i$. Hence, ψ' is a DDNF of ψ, and we have shown that for $\psi : p(\alpha) = x$, $\Phi \Vdash_\Pi \psi$ iff $\Phi \Vdash_\Pi \psi'$.

Let us now consider point a, i.e. where $\alpha' : c_1 \vee \ldots \vee c_j$ and $x = 1$. We can observe that by the construction of α', this means that α' (and α) are equivalent to $\top$. By repeating the procedures for the last point, we can show that for α', $\Phi \Vdash_\Pi p(\alpha') = 1$ iff $\Phi \Vdash_\Pi \bigvee_{(\tau_1, \ldots, \tau_j) \in T^\Pi_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$. Given that $p(\top) = 1$ is a tautology (see derivable rules), we can use derivable rule 23 to show that $p(\alpha') = 1$ is a tautology as well. Hence, it is easy to check that the obtained formula is indeed a DDNF of ψ. Please note that the same formula can also be obtained through the use of the propositional rules, basic rule B3 and the second point of this proposition.

Let us now consider point b, i.e. where $\alpha' : c_1 \vee \ldots \vee c_j$ and $x \neq 1$. We can show that $\Phi \Vdash_\Pi p(\alpha') = x$ iff $\Phi \Vdash_\Pi \bigvee_{(\tau_1, \ldots, \tau_j) \in T^\Pi_{x,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$. Since $\Phi \Vdash_\Pi \bigvee_{(\tau_1, \ldots, \tau_j) \in T^\Pi_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$ and $x \neq 1$, we can use the propositional, enumeration and derivable rules in order to show that $\Phi \Vdash_\Pi p(\alpha') = x$ iff $\Phi \Vdash_\Pi \perp$. Another way to arrive at this conclusion is to use the propositional rules to obtain $p(\top) = x$ for $x \neq 1$, and then use the derivable rules to observe that this leads to a contradiction. Given that $\mathsf{Sat}(p(\alpha') = x, \Pi) = \emptyset$ for $x \neq 1$, then $\perp$ is indeed the DDNF of ψ.

Let us now consider point c, i.e. $\alpha' : \perp$ and $x = 0$. By combining basic rules B3 and B4 we can observe that $\Phi \Vdash_\Pi p(\perp) = 0$ iff $\Phi \Vdash_\Pi p(\top) = 1$. We can therefore reuse the previous analysis to show that $\Phi \Vdash_\Pi p(\perp) = 0$ iff $\Phi \Vdash_\Pi \bigvee_{(\tau_1, \ldots, \tau_j) \in T^\Pi_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$. Given that $\mathsf{Sat}(p(\perp) = 0, \Pi) = \mathsf{Dist}(\mathcal{G}, \Pi) = \mathsf{Sat}(\bigvee_{(\tau_1, \ldots, \tau_j) \in T^\Pi_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j))$, it is easy to verify that the obtained formula is indeed the DDNF of ψ.

Finally, consider point d, i.e. where $\alpha' : \perp$ and $x \neq 0$. Based on the derivable rules, we can observe that $\Phi \Vdash_\Pi p(\perp) = 0$ and that for $x \neq 0$, $\Phi \Vdash_\Pi p(\perp) = 0 \wedge p(\perp) = x$ iff $\Phi \Vdash_\Pi \perp$. We therefore observe that $\Phi \Vdash_\Pi p(\alpha) = x$ iff $\Phi \Vdash_\Pi \perp$ in this case. Given that $\mathsf{Sat}(p(\perp) = x) = \emptyset$ for $x \neq 0$, it is easy to see that $\perp$ is indeed the DDNF for ψ.

We can therefore conclude that for all ψ of the form $p(\alpha) = x$, $\Phi \Vdash_\Pi \psi$ iff $\Phi \Vdash_\Pi \varphi$, where φ is the DDNF of ψ.

We now also consider $\psi : \top$ and $\psi : \perp$. If $\psi : \top$, then we can, for example, use basic rule B3 along with the previous parts of the proof to show that $\bigvee_{(\tau_1, \ldots, \tau_j) \in T^\Pi_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$ is the DDNF of ψ. If $\psi : \perp$, then we can observe that it is already in DDNF.

Let us now consider more complex epistemic formulae ψ. We first transform ψ to its (propositional) negation normal form. We then replace every negated epistemic atom with a positive one using the derivable rules 15 to 20. Based on enumeration rule E1, every epistemic atom using $\# \in \{>, <, \geq, \leq, \neq\}$ can be equivalently expressed by a disjunction of atoms using only equality or by $\perp$. Hence, ψ can be transformed into $\psi_=$ that uses only positive equality atoms or $\perp$ and s.t. $\Phi \Vdash_\Pi \psi$ iff $\Phi \Vdash_\Pi \psi_=$. Every term in an epistemic atom can be transformed into $\perp$ or a disjunction of certain argument complete formulae. Furthermore, every epistemic atom containing a term equivalent to $\perp$ can, depending on the value x contained in the atom, be replaced by an epistemic atom $\perp$ or $\top$ through the use of the propositional and basic rules. In a similar fashion, every epistemic atom containing a term equivalent to $\top$ (i.e. one using all possible argument complete formulae) can, depending on the value of x, be replaced by an epistemic atom $\perp$ or $\top$. The resulting epistemic formula can be transformed into a minimal (propositional) conjunctive normal form and if required, the derivable rules and the propositional identity and domination laws[11] can be used to further simplify it. We therefore obtain a formula ψ' s.t. $\psi' : \top$, or $\psi' : \perp$, or $\psi' : \psi'_1 \wedge \ldots \wedge \psi'_a$ where $a \geq 1$ and every

[11] Recall that the identity and domination logical equivalence laws state that $q \wedge \top \equiv q$, $q \vee \perp \equiv q$, $q \vee \top \equiv \top$ and $q \wedge \perp \equiv \perp$.

$\psi'_i : p(\alpha_{i_1}) = x_{i_1} \vee \ldots \vee p(\alpha_{i_n}) = x_{i,n}$ s.t. every α_{i_k} is a disjunction of certain argument complete formulae and is not equivalent to neither $\bot$ nor $\top$. Given the used syntactical rules, we can observe that $\Phi \Vdash_\Pi \psi$ iff $\Phi \Vdash_\Pi \psi'$. If $\psi' = \top$ or $\psi' = \bot$, then we refer the reader to the previous part of this proof concerning how ψ' can be transformed to an appropriate DDNF. We therefore focus on the case where $\psi' : \psi'_1 \wedge \ldots \wedge \psi'_a$.

For every $\psi'_i : p(\alpha_{i_1}) = x_{i_1} \vee \ldots \vee p(\alpha_{i_n}) = x_{i_n}$, every $p(\alpha_{i_k}) = x_{i_k}$ can be transformed into a DDNF formula φ_{i_k} using the previous parts of this proof. Through the use of associativity, every ψ'_i can be equivalently written down as a disjunction of epistemic formulae $\varphi_{i_1} \vee \ldots \vee \varphi_{i_{n_i}}$ where each formula φ_{i_k} is associated with a probability distribution and s.t. $\Phi \Vdash_\Pi \psi'_i$ iff $\Phi \Vdash_\Pi \varphi_{i_1} \vee \ldots \vee \varphi_{i_{n_i}}$. Without the loss of generality, we can assume that every two φ_{i_k} and φ_{i_l} formulae are different - otherwise the idempotent law can be used to remove duplicate formulae. We now therefore have that $\Phi \Vdash_\Pi \psi$ iff $\Phi \Vdash_\Pi (\varphi_{1_1} \vee \ldots \vee \varphi_{1_{n_1}}) \wedge \ldots \wedge (\varphi_{a_1} \vee \ldots \vee \varphi_{a_{n_a}})$. Using the propositional rules and the derivable rule 33, we can show that for any two formulae $\varphi_{i_k}, \varphi_{m_l} \in \{\varphi_{1_1}, \ldots, \varphi_{1_{n_1}}, \ldots, \varphi_{a_1}, \ldots, \varphi_{a_{n_a}}\}$ associated with probability distributions, if $\varphi_{i_k} \not\approx \varphi_{m_l}$ then $\Phi \Vdash_\Pi \varphi_{i_k} \wedge \varphi_{m_l}$ iff $\Phi \Vdash_\Pi \bot$. Using this and the distribution laws, we can bring $(\varphi_{1_1} \vee \ldots \vee \varphi_{1_{n_1}}) \wedge \ldots \wedge (\varphi_{a_1} \vee \ldots \vee \varphi_{a_{n_a}})$ into minimal (propositional) disjunctive normal form γ s.t. it is either $\bot$ or every conjunctive clause is an epistemic formula associated with a distribution. Due to the nature of the syntactic rules we have used, we can observe that $\Phi \Vdash_\Pi \psi$ iff $\Phi \Vdash_\Pi \gamma$. Since our system is sound, it holds that $\Phi \Vdash\!\!\models_\Pi \psi$ iff $\Phi \Vdash\!\!\models_\Pi \gamma$. By using the same argument as in the case of $\psi : p(\alpha) = x$ in the previous parts of this proof, we can show that γ is indeed the DDNF of ψ.

We can therefore conclude that for all $\psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$, $\Phi \Vdash_\Pi \psi$ iff $\Phi \Vdash_\Pi \varphi$, where φ is the DDNF of ψ.

$\square$

Proposition 4.23. *Let Π be a restricted value set. For $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$, and $\psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$, $\Phi \Vdash_\Pi \psi$ iff $\Phi \Vdash\!\!\models_\Pi \psi$.*

Proof. We have shown that our system is sound in Proposition 4.21. We now need to show that the system is also complete, i.e. that if $\Phi \Vdash\!\!\models_\Pi \psi$ then $\Phi \Vdash_\Pi \psi$.

Let $\Phi = \{\phi_1, \ldots, \phi_m\}$ and $\phi : \phi_1 \wedge \ldots \wedge \phi_m$. By using the propositional rules we can easily show that $\Phi \Vdash_\Pi \phi$ and for every formula $\gamma \in \Phi$, $\{\phi\} \Vdash_\Pi \gamma$. Furthermore, it clearly follows from the definition of Sat that $\mathsf{Sat}(\Phi, \Pi) = \mathsf{Sat}(\phi, \Pi)$. Consequently, for the purpose of this proof, it suffices to show that if $\{\phi\} \Vdash\!\!\models_\Pi \psi$ then $\{\phi\} \Vdash_\Pi \psi$. If $\Phi = \emptyset$, then it is easy to see that we can set ϕ to $\top$.

Let $\mathsf{Sat}(\phi, \Pi) = \{P_1, \ldots, P_k\}$ and $\mathsf{Sat}(\psi, \Pi) = \{P'_1, \ldots, P'_l\}$. Let φ^ϕ and φ^ψ be the DDNFs of ϕ and ψ. Based on Proposition 4.18, $\mathsf{Sat}(\psi, \Pi) = \mathsf{Sat}(\varphi^\psi, \Pi)$ and $\mathsf{Sat}(\phi, \Pi) = \mathsf{Sat}(\varphi^\phi, \Pi)$. Therefore, $\{\phi\} \Vdash\!\!\models_\Pi \psi$ iff $\{\varphi^\phi\} \Vdash\!\!\models_\Pi \varphi^\psi$. By definition, $\{\varphi^\phi\} \Vdash\!\!\models_\Pi \varphi^\psi$ iff $\mathsf{Sat}(\varphi^\phi, \Pi) \subseteq \mathsf{Sat}(\varphi^\psi, \Pi)$. If $k = 0$, then the DDNF of ϕ is $\bot$ and therefore through the propositional rule P2, we can show that $\{\phi\} \Vdash_\Pi \psi$ for any ψ. If $k \neq 0$ and $l = 0$, then it cannot be the case that $\mathsf{Sat}(\varphi^\phi, \Pi) \subseteq \mathsf{Sat}(\varphi^\psi, \Pi)$. Therefore, it cannot be the case that $\{\phi\} \Vdash\!\!\models_\Pi \psi$ either and we reach a contradiction. Hence, let $k \neq 0$ and $l \neq 0$. Let therefore φ^ϕ and φ^ψ be of the form $\varphi^\phi : \varphi^{P_1} \vee \varphi^{P_2} \ldots \vee \varphi^{P_k}$ and $\varphi^\psi : \varphi^{P'_1} \vee \varphi^{P'_2} \ldots \vee \varphi^{P'_l}$. Since $\mathsf{Sat}(\varphi^\phi, \Pi) \subseteq \mathsf{Sat}(\varphi^\psi, \Pi)$, then for every P_i there exists P'_j s.t. $P_i = P'_j$ and therefore for every φ^{P_i} there exists an equivalent $\varphi^{P'_j}$. Consequently, by using the propositional proof rule P1 it is easy to show that if $\{\varphi^\phi\} \Vdash\!\!\models_\Pi \varphi^\psi$ then $\{\varphi^\phi\} \Vdash_\Pi \varphi^\psi$. We can now use Proposition 4.22 to show that $\{\phi\} \Vdash_\Pi \psi$. We can therefore conclude that if $\Phi \Vdash\!\!\models_\Pi \psi$ then $\Phi \Vdash_\Pi \psi$ and our system is complete. $\square$

Proposition 4.24. *Let Π be a reasonable restricted value set. For $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$ and $\psi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$, $\Phi \Vdash_\Pi \psi$ iff $\Phi \cup \{\neg\psi\} \Vdash_\Pi \bot$.*

Proof. For the purpose of this proof, we introduce a shorthand $\varphi \approx \varphi'$ stating that the two formulae have the same syntactical features under commutativity and associativity (so, for example, $(\mathtt{A} \vee \mathtt{B}) \wedge \mathtt{C} \approx (\mathtt{A} \vee \mathtt{B}) \wedge \mathtt{C}$, $(\mathtt{A} \vee \mathtt{B}) \wedge \mathtt{C} \approx (\mathtt{B} \vee \mathtt{A}) \wedge \mathtt{C}$, but $(\mathtt{A} \vee \mathtt{B}) \wedge \mathtt{C} \not\approx (\mathtt{A} \wedge \mathtt{C}) \vee (\mathtt{B} \wedge \mathtt{C})$).

We want to show that $\Phi \Vdash_\Pi \psi$ iff $\Phi \cup \{\neg\psi\} \Vdash_\Pi \bot$. Let $\Phi = \{\phi_1, \ldots, \phi_m\}$ and $\phi : \phi_1 \wedge \ldots \wedge \phi_m$. By using the propositional rules we can easily show that $\Phi \Vdash_\Pi \phi$ and for every formula $\gamma \in \Phi$, $\{\phi\} \Vdash_\Pi \gamma$. Furthermore, it clearly follows from the definition of Sat that $\mathsf{Sat}(\Phi, \Pi) = \mathsf{Sat}(\phi, \Pi)$. If $\Phi = \emptyset$, then it is easy to see that we can set ϕ to $\top$. Let now φ^{ϕ} and φ^{ψ} be DDNFs of ϕ and ψ. Based on Propositions 4.23 and 4.22 the definition of DDNF, we can observe that for the purpose of this proof, it suffices to show that $\{\varphi^{\phi}\} \Vdash_{\Pi} \varphi^{\psi}$ iff $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} \bot$. Without the loss of generality, let us assume that we have an ordering over arguments, and if a given DDNF is not $\bot$, then the atoms in every subformula associated with a probability distribution are ordered according to the set of argument complete terms $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \ldots, c_p\}$ of $\mathcal{G}$ and that the terms themselves are also ordered. We can use this assumption due to derivable rule 23 (i.e. two probabilistic atoms with same constants and argument complete terms equivalent under commutativity are themselves equivalent) and propositional rules (commutativity law).

Let us focus on the left to right direction first. Assume $\varphi^{\phi} : \bot$. Then it is easy to see that $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} \bot$ independently of the nature of $\neg\varphi^{\psi}$. We can also observe that if $\varphi^{\psi} : \bot$, then as $\{\varphi^{\phi}\} \Vdash_{\Pi} \varphi^{\psi}$, $\varphi^{\phi} : \bot$ as well. Thus, based on previous remarks, the property still holds.

Let us therefore consider the situation in which none of the DDNFs are $\bot$ and let φ^{ϕ} and φ^{ψ} be of the forms $\varphi^{\phi} : \varphi^{\phi}_1 \lor \ldots \lor \varphi^{\phi}_m$ and $\varphi^{\psi} : \varphi^{\psi}_1 \lor \ldots \lor \varphi^{\psi}_n$ for $m, n \geq 1$. We can observe that due to the nature of DDNF, every φ^{ϕ}_i and φ^{ψ}_j is an epistemic formula associated with a (single) probability distribution. Therefore, if $\{\varphi^{\phi}\} \Vdash_{\Pi} \varphi^{\psi}$, then based on the ordering assumption we have made, it has to be the case that $\{\varphi^{\phi}_1, \ldots, \varphi^{\phi}_m\} \subseteq \{\varphi^{\psi}_1, \ldots, \varphi^{\psi}_n\}$. Thus, for φ^{ϕ}_i there exists a φ^{ψ}_j s.t. $\varphi^{\phi}_i \approx \varphi^{\psi}_j$. Thus, using the propositional and derivable rules, we can show that for such formulae, $\{\varphi^{\phi}_i \land \neg\varphi^{\psi}_j\} \Vdash_{\Pi} \bot$. Using this, we can now prove the following. Using propositional rules, $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} \varphi^{\phi} \land \neg\varphi^{\psi}$ which is $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} (\varphi^{\phi}_1 \lor \ldots \lor \varphi^{\phi}_m) \land \neg(\varphi^{\psi}_1 \lor \ldots \lor \varphi^{\psi}_n)$, which in turns is equivalent to $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} (\varphi^{\phi}_1 \lor \ldots \lor \varphi^{\phi}_m) \land (\neg\varphi^{\psi}_1 \land \ldots \land \neg\varphi^{\psi}_n)$ and to $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} (\varphi^{\phi}_1 \land \neg\varphi^{\psi}_1 \land \ldots \land \neg\varphi^{\psi}_n) \lor \ldots \lor (\varphi^{\phi}_m \land \neg\varphi^{\psi}_1 \land \ldots \land \neg\varphi^{\psi}_n)$. Based on the previous explanations and propositional rules, for every φ^{ϕ}_i, $\{\varphi^{\phi}_i \land \neg\varphi^{\psi}_1 \land \ldots \land \neg\varphi^{\psi}_n\} \vdash \bot$. Hence, using propositional rules once more, $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} \bot \lor \ldots \lor \bot$ and therefore $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} \bot$. This concludes the left to right direction of our proof.

Let us now focus on the right to left direction of our proof, i.e. that if $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} \bot$ then $\{\varphi^{\phi}\} \Vdash_{\Pi} \varphi^{\psi}$. Assume $\varphi^{\phi} : \bot$. Then clearly, $\{\varphi^{\phi}\} \Vdash_{\Pi} \varphi^{\psi}$ based on propositional rules. Now assume $\varphi^{\psi} : \bot$. Then $\neg\varphi^{\psi}$ is equivalent to $\top$ through the use of propositional rules and our property holds easily and it is easy to argue that if $\{\varphi^{\phi} \land \top\} \Vdash_{\Pi} \bot$ then $\{\varphi^{\phi}\} \Vdash_{\Pi} \bot$.

Let us therefore consider the situation in which none of the DDNFs are $\bot$ and let φ^{ϕ} and φ^{ψ} be of the forms $\varphi^{\phi} : \varphi^{\phi}_1 \lor \ldots \lor \varphi^{\phi}_m$ and $\varphi^{\psi} : \varphi^{\psi}_1 \lor \ldots \lor \varphi^{\psi}_n$ for $m, n \geq 1$. Assume that it is not the case that $\{\varphi^{\phi}\} \Vdash_{\Pi} \varphi^{\psi}$. Therefore, using the previous analysis and the properties of DDNFs, we can show that there must exist $\varphi^{\phi}_i \in \{\varphi^{\phi}_1, \ldots, \varphi^{\phi}_m\}$ s.t. $\varphi^{\phi}_i \notin \{\varphi^{\psi}_1, \ldots, \varphi^{\psi}_n\}$ (i.e. there exists a probability distribution satisfying φ^{ϕ} and not satisfying φ^{ψ}, and therefore a formula associated with that probability present in φ^{ϕ} and not present in φ^{ψ}). Therefore, we can reuse the previous reasoning to show that it cannot be the case that $\{\varphi^{\phi}_1 \land \neg\varphi^{\psi}_1 \land \ldots \land \neg\varphi^{\psi}_n) \lor \ldots \lor (\varphi^{\phi}_m \land \neg\varphi^{\psi}_1 \land \ldots \land \neg\varphi^{\psi}_n\} \Vdash_{\Pi} \bot$ and therefore it cannot be the case that $\{\varphi^{\phi} \land \top\} \Vdash_{\Pi} \bot$. We reach a contradiction with our assumptions. Hence, we conclude that if $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} \bot$ then $\{\varphi^{\phi}\} \Vdash_{\Pi} \varphi^{\psi}$. Given the previous results, we have therefore shown that $\{\varphi^{\phi} \land \neg\varphi^{\psi}\} \Vdash_{\Pi} \bot$ iff $\{\varphi^{\phi}\} \Vdash_{\Pi} \varphi^{\psi}$. □

Lemma 4.25. *Let Π be a reasonable restricted value set, and let the restricted constraint language w.r.t. Π be* $\mathsf{BFormulae}(\mathcal{G}, \Pi)$. *There is a set of propositional formulae Ω with $\Lambda \subseteq \Omega$, and there is a function f :* $\mathsf{BFormulae}(\mathcal{G}, \Pi) \to \Omega$ *s.t. for each $\{\phi_1, \ldots, \phi_n\} \subseteq$* $\mathsf{BFormulae}(\mathcal{G}, \Pi)$*, and for each $\psi \in$* $\mathsf{BFormulae}(\mathcal{G}, \Pi)$*,*

$$\{\phi_1, \ldots, \phi_n\} \Vdash_{\Pi} \psi \textit{ iff } \{f(\phi_1), \ldots, f(\phi_n)\} \cup \Lambda \vdash f(\psi)$$

Proof. First, we provide the following translation, and then show the equivalence. Let Π be a restricted value set, and let the basic restricted constraint language w.r.t. Π be $\mathsf{BFormulae}(\mathcal{G}, \Pi)$. For each $\phi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$, let $f(\phi)$ be defined as follows:

- for an atom of the form $p(\alpha)\#x$, $f(p(\alpha)\#x) = a^{\alpha}_{\#x}$
- for a negated formula of the form $\neg\phi$, $f(\neg\phi) = \neg(f(\phi))$

- for a conjunction of the form $\phi \wedge \psi$, $f(\phi \wedge \psi) = f(\phi) \wedge f(\psi)$
- for a disjunction of the form $\phi \vee \psi$, $f(\phi \vee \psi) = f(\phi) \vee f(\psi)$

So Ω contains $f(\phi_i)$ for each $\phi_i \in \{\phi_1, \ldots, \phi_n\}$. In addition, for each atom used in the propositional formulae in Ω, we create a number of propositional formulae corresponding to the proof rules for the epistemic entailment relation. Let $a^{\alpha}_{\#x}$ be an atom used in the propositional formulae in Ω. We form the set of propositional formulae $\Lambda \subseteq \Omega$ as follows.

- Following basic rule B1, we add rule $a^{\alpha}_{\geq 0} \leftrightarrow \top$
- Following basic rule B2, we add rule $a^{\alpha}_{\leq 1} \leftrightarrow \top$
- Following basic rule B3, we add rule $a^{\top}_{=1} \leftrightarrow \top$
- Following basic rule B4, we add rule $a^{\perp}_{=0} \leftrightarrow \top$
- Assume that $\Pi^{0,(-,-,+)}_{=} = \{(x_1, y_1, z_1, v_1), \ldots, (x_m, y_m, z_m, v_m)\}$. Following the probabilistic rule PR1, we add rule $(a^{\alpha\vee\beta}_{=x_1} \wedge a^{\alpha}_{y_1} \wedge a^{\beta}_{=z_1} \wedge a^{\alpha\wedge\beta}_{=v_1}) \vee \ldots \vee (a^{\alpha\vee\beta}_{=x_m} \wedge a^{\alpha}_{y_m} \wedge a^{\beta}_{=z_m} \wedge a^{\alpha\wedge\beta}_{=v_m})$
- Following subject rule S1, we add rule $a^{\alpha}_{>x} \rightarrow a^{\beta}_{>x}$ for each β such that $\{\alpha\} \vdash \beta$
- Following subject rule S2, we add rule $a^{\alpha}_{\geq x} \rightarrow a^{\beta}_{\geq x}$ for each β such that $\{\alpha\} \vdash \beta$
- Following subject rule S3, we add rule $a^{\alpha}_{\leq x} \rightarrow a^{\beta}_{\leq x}$ for each β such that $\{\beta\} \vdash \alpha$
- Following subject rule S4, we add rule $a^{\alpha}_{<x} \rightarrow a^{\beta}_{<x}$ for each β such that $\{\beta\} \vdash \alpha$
- Following enumeration rule E1, we add rules $a^{\alpha}_{\#x} \leftrightarrow \bigvee_{v \in \Pi^{x}_{\#}} a^{\alpha}_{=v}$ for $\# \in \{>, <, \leq, \geq, \neq\}$ if $\Pi^{x}_{\#} \neq \emptyset$ and $a^{\alpha}_{\#x} \leftrightarrow \perp$ otherwise
- Following enumeration rule E2, we add rule $a^{\alpha}_{>x} \leftrightarrow \neg(\bigvee_{v \in \Pi^{x}_{\leq}} a^{\alpha}_{=v})$
- Following enumeration rule E3, we add rule $a^{\alpha}_{\geq x} \leftrightarrow \neg(\bigvee_{v \in \Pi^{x}_{<}} a^{\alpha}_{=v})$ if $\Pi^{x}_{<} \neq \emptyset$ and $a^{\alpha}_{\geq x} \leftrightarrow \neg\perp$ otherwise
- Following enumeration rule E4, we add rule $a^{\alpha}_{<x} \leftrightarrow \neg(\bigvee_{v \in \Pi^{x}_{\geq}} a^{\alpha}_{=v})$
- Following enumeration rule E5, we add rule $a^{\alpha}_{\leq x} \leftrightarrow \neg(\bigvee_{v \in \Pi^{x}_{>}} a^{\alpha}_{=v})$ if $\Pi^{x}_{>} \neq \emptyset$ and $a^{\alpha}_{\leq x} \leftrightarrow \neg\perp$ otherwise

With this translation, we can observe that the propositional proof rules P1 and P2 and the proof rules for classical propositional logic correspond. Therefore, given Λ, it is straightforward to show by induction on the structure of the epistemic formulae in $\phi_1, \ldots, \phi_n$ and ψ that $\{\phi_1, \ldots, \phi_n\} \Vdash_{\Pi} \psi$ iff $\{f(\phi_1), \ldots, f(\phi_n)\} \cup \Lambda \vdash f(\psi)$. □

Lemma 4.26. *Let Ω be a propositional language composed from a set of atoms and the usual definitions for the Boolean connectives. There is a restricted constraint language* $\mathsf{BFormulae}(\mathcal{G}, \Pi)$ *where $\Pi = \{0, 1\}$ and there is a function $g : \Omega \rightarrow$* $\mathsf{BFormulae}(\mathcal{G}, \Pi)$ *s.t. for each set of propositional formulae $\{\alpha_1, \ldots, \alpha_n\} \subseteq \Omega$ and for each propositional formula $\beta \in \Omega$,*

$$\{\alpha_1, \ldots, \alpha_n\} \vdash \beta \textit{ iff } \{g(\alpha_1), \ldots, g(\alpha_n)\} \Vdash_{\Pi} g(\beta)$$

Proof. We assume that $\Pi = \{0, 1\}$. First, we provide the following translation, and then show the equivalence. For each propositional formula α, let $g(\alpha)$ be defined as follows:

- if α is an atom, then $g(\alpha)$ is $p(\alpha) = 1$
- if α is negation of the form $\neg\alpha'$, then $g(\neg\alpha') = \neg g(\alpha')$.

- if α is conjunction of the form $\alpha_1 \wedge \alpha_2$, then $g(\alpha_1 \wedge \alpha_2) = g(\alpha_1) \wedge g(\alpha_2)$.
- if α is disjunction of the form $\alpha_1 \vee \alpha_2$, then $g(\alpha_1 \vee \alpha_2) = g(\alpha_1) \vee g(\alpha_2)$.

We can observe that there is a correspondence between the propositional proof rules P1 and P2 and the proof rules for classical propositional logic under this translation. Hence, it is straightforward to show by induction on the structure of the propositional formulae in $\alpha_1, \ldots, \alpha_n$ and β that $\{\alpha_1, \ldots, \alpha_n\} \vdash \beta$ iff $\{g(\alpha_1), \ldots, g(\alpha_n)\} \Vdash_\Pi g(\beta)$. □

Proposition 4.27. *The restricted constraint language with the epistemic consequence relation is equivalent to the classical propositional language with the classical propositional consequence relation.*

Proof. Follows easily from Lemma 4.25 and 4.26. □

Proposition 4.29. *For valued epistemic atoms $\varphi_1 = f_1\#x$ and $\varphi_2 = f_2\#x$ in $\mathsf{VFormulae}(\mathcal{G}, \Pi)$, the following hold:*

- *if $\varphi_1 \simeq_{re} \varphi_2$ then $\mathsf{Sat}(\varphi_1) = \mathsf{Sat}(\varphi_2)$*
- *if $\varphi_1 \succeq^+_{su} \varphi_2$ and $\# \in \{>, \geq\}$ then $\mathsf{Sat}(\varphi_1) \subseteq \mathsf{Sat}(\varphi_2)$, and if $\# \in \{<, \leq\}$, then $\mathsf{Sat}(\varphi_2) \subseteq \mathsf{Sat}(\varphi_1)$*
- *if $\varphi_1 \succeq^-_{su} \varphi_2$ and $\# \in \{<, \leq\}$ then $\mathsf{Sat}(\varphi_1) \subseteq \mathsf{Sat}(\varphi_2)$, and if $\# \in \{>, \geq\}$, then $\mathsf{Sat}(\varphi_2) \subseteq \mathsf{Sat}(\varphi_1)$*

Proof. Let $f_1 : p(\alpha_1) *_1 p(\alpha_2) *_2 \ldots *_{m-1} p(\alpha_m)$, and $f_2 : p(\beta_1) \star_1 p(\beta_2) \star_2 \ldots \star_{l-1} p(\beta_l)$, where $\alpha_i, \beta_i \in \mathsf{Terms}(\mathcal{G})$ and $*_i, \star_i \in \{+, -\}$, be operational formulae. Let $\varphi_1 = f_1\#x$ and $\varphi_2 = f_2\#x$, where $\# \in \{=, \neq, \geq, \leq, >, <\}$ and $x \in [0, 1]$.

- Given the properties of addition and subtraction, it is easy to see that if f_1 is a valid arithmetical rearrangement of f_2, then $f_1 = f_2$. Consequently, it is easy to show that $f_1\#x$ iff $f_2\#x$ and that any probability distribution in $\mathsf{Sat}(\varphi_1)$ is in $\mathsf{Sat}(\varphi_2)$ and vice versa. Hence, $\mathsf{Sat}(\varphi_1) = \mathsf{Sat}(\varphi_2)$
- Let $p(\alpha_i)$ be the element that became weakened to $p(\alpha'_i)$ (i.e. $\beta_i = \alpha'_i$). Since $\{\alpha_i\} \vdash \alpha'_i$, then for any probability distribution P it holds that $P(\alpha_i) \leq P(\alpha'_i)$. Therefore, for any probability distribution P, $P(\alpha_1) *_1 P(\alpha_2) *_2 \ldots + P(\alpha_i) *_i \ldots *_{m-1} P(\alpha_m) \leq P(\alpha_1) *_1 P(\alpha_2) \ldots + P(\alpha'_i) *_i \ldots *_{m-1} P(\alpha_m)$. Consequently, if $P(\alpha_1) *_1 P(\alpha_2) *_2 \ldots + P(\alpha_i) *_i \ldots *_{m-1} P(\alpha_m)\#x$, where $\# \in \{>, \geq\}$, then $P(\alpha_1) *_1 P(\alpha_2) \ldots + P(\alpha'_i) *_i \ldots *_{m-1} P(\alpha_m)\#x$ as well. Hence, for every $P' \in \mathsf{Sat}(\varphi_1)$, $P' \in \mathsf{Sat}(\varphi_2)$ as well, and it holds that $\mathsf{Sat}(\varphi_1) \subseteq \mathsf{Sat}(\varphi_2)$. Furthermore, if $P(\alpha_1) *_1 P(\alpha_2) *_2 \ldots + P(\alpha'_i) *_i \ldots *_{m-1} P(\alpha_m)\#x$, where $\# \in \{<, \leq\}$, then $P(\alpha_1) *_1 P(\alpha_2) \ldots + P(\alpha_i) *_i \ldots *_{m-1} P(\alpha_m)\#x$ as well. Hence, for every $P' \in \mathsf{Sat}(\varphi_2)$, $P' \in \mathsf{Sat}(\varphi_1)$ as well, and it holds that $\mathsf{Sat}(\varphi_2) \subseteq \mathsf{Sat}(\varphi_1)$.
- Let $p(\alpha_i)$ be the element that became weakened to $p(\alpha'_i)$ (i.e. $\beta_i = \alpha'_i$). Since $\{\alpha_i\} \vdash \alpha'_i$, then for any probability distribution P it holds that $P(\alpha_i) \leq P(\alpha'_i)$. Therefore, for any probability distribution P, $P(\alpha_1) *_1 P(\alpha_2) *_2 \ldots - P(\alpha_i) *_i \ldots *_{m-1} P(\alpha_m) \geq P(\alpha_1) *_1 P(\alpha_2) \ldots - P(\alpha'_i) *_i \ldots *_{m-1} P(\alpha_m)$. Therefore, if $P(\alpha_1) *_1 P(\alpha_2) *_2 \ldots - P(\alpha_i) *_i \ldots *_{m-1} P(\alpha_m)\#x$, where $\# \in \{<, \leq\}$, then $P(\alpha_1) *_1 P(\alpha_2) \ldots - P(\alpha'_i) *_i \ldots *_{m-1} P(\alpha_m)\#x$ as well. Hence, for every $P' \in \mathsf{Sat}(\varphi_1)$, $P' \in \mathsf{Sat}(\varphi_2)$ as well, and it holds that $\mathsf{Sat}(\varphi_1) \subseteq \mathsf{Sat}(\varphi_2)$. Furthermore, if $P(\alpha_1) *_1 P(\alpha_2) \ldots - P(\alpha'_i) *_i \ldots *_{m-1} P(\alpha_m)\#x$, where $\# \in \{>, \geq\}$ then $P(\alpha_1) *_1 P(\alpha_2) *_2 \ldots - P(\alpha_i) *_i \ldots *_{m-1} P(\alpha_m)\#x$ as well. Hence, for every $P' \in \mathsf{Sat}(\varphi_2)$, $P' \in \mathsf{Sat}(\varphi_1)$ as well, and it holds that $\mathsf{Sat}(\varphi_2) \subseteq \mathsf{Sat}(\varphi_1)$.

□

Proposition 4.31. *Let $\# \in \{=, \neq, \geq, \leq, >, <\}$ be the set of inequality relationships, let Π be a reasonable restricted value set, and let $\Pi^x_\# = \{y \in \Pi \mid x\#y\}$ be the subset obtained according to the value x and relationship $\#$. Let $f_1 : p(\alpha_1) *_1 p(\alpha_2) *_2 \ldots *_{k-1} p(\alpha_k)$ and $f_2 : p(\beta_1) \star_1 p(\beta_2) \star_2 \ldots \star_{l-1} p(\beta_l)$, where $k, l \geq 1$, $\alpha_i, \beta_i \in \mathsf{Terms}(\mathcal{G})$ and $\star_j, *_i \in \{+, -\}$ be operational formulae. The following hold, where $\vdash$ is propositional consequence relation, $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$, $\phi, \psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$ and $x \in \Pi$.*

1. $\Phi \Vdash^+_\Pi p(\alpha) \geq 0$

2. $\Phi \Vdash^+_\Pi p(\alpha) \leq 1$

3. $\Phi \Vdash^+_\Pi p(\top) = 1$

4. $\Phi \Vdash^+_\Pi p(\bot) = 0$

5. $\Phi \Vdash^+_\Pi f_1 > x$ *implies* $\Phi \Vdash^+_\Pi f_1 \geq x$

6. $\Phi \Vdash^+_\Pi f_1 < x$ *implies* $\Phi \Vdash^+_\Pi f_1 \leq x$

7. $\Phi \Vdash^+_\Pi f_1 = x$ *implies* $\Phi \Vdash^+_\Pi f_1 \geq x$

8. $\Phi \Vdash^+_\Pi f_1 = x$ *implies* $\Phi \Vdash^+_\Pi f_1 \leq x$

9. $\Phi \Vdash^+_\Pi f_1 \neq x$ *iff* $\Phi \Vdash^+_\Pi f_1 > x \vee f_1 < x$

10. $\Phi \Vdash^+_\Pi f_1 \geq x$ *iff* $\Phi \Vdash^+_\Pi f_1 > x \vee f_1 = x$

11. $\Phi \Vdash^+_\Pi f_1 \leq x$ *iff* $\Phi \Vdash^+_\Pi f_1 < x \vee f_1 = x$

12. $\Phi \Vdash^+_\Pi f_1 > x$ *iff* $\Phi \Vdash^+_\Pi f_1 \geq x \wedge f_1 \neq x$

13. $\Phi \Vdash^+_\Pi f_1 < x$ *iff* $\Phi \Vdash^+_\Pi f_1 \leq x \wedge f_1 \neq x$

14. $\Phi \Vdash^+_\Pi f_1 = x$ *iff* $\Phi \Vdash^+_\Pi f_1 \geq x \wedge f_1 \leq x$

15. $\Phi \Vdash^+_\Pi f_1 > x$ *iff* $\Phi \Vdash^+_\Pi \neg(f_1 \leq x)$

16. $\Phi \Vdash^+_\Pi f_1 < x$ *iff* $\Phi \Vdash^+_\Pi \neg(f_1 \geq x)$

17. $\Phi \Vdash^+_\Pi f_1 \leq x$ *iff* $\Phi \Vdash^+_\Pi \neg(f_1 > x)$

18. $\Phi \Vdash^+_\Pi f_1 \geq x$ *iff* $\Phi \Vdash^+_\Pi \neg(f_1 < x)$

19. $\Phi \Vdash^+_\Pi f_1 = x$ *iff* $\Phi \Vdash^+_\Pi \neg(f_1 \neq x)$

20. $\Phi \Vdash^+_\Pi f_1 \neq x$ *iff* $\Phi \Vdash^+_\Pi \neg(f_1 = x)$

21. $\Phi \Vdash^+_\Pi f_1 = x$ *and* $f_1 \succeq^+_{su} f_2$ *implies* $\Phi \Vdash^+_\Pi f_2 \geq x$

22. $\Phi \Vdash^+_\Pi f_1 = x$ *and* $f_1 \succeq^-_{su} f_2$ *implies* $\Phi \Vdash^+_\Pi f_2 \leq x$

23. $\Phi \Vdash^+_\Pi f_2 = x$ *and* $f_1 \succeq^+_{su} f_2$ *implies* $\Phi \Vdash^+_\Pi f_1 \leq x$

24. $\Phi \Vdash^+_\Pi f_2 = x$ *and* $f_1 \succeq^-_{su} f_2$ *implies* $\Phi \Vdash^+_\Pi f_1 \geq x$

25. *if* $f_1 \succeq^+_{su} f_2$ *and* $f_2 \succeq^+_{su} f_1$, *then* $\Phi \Vdash^+_\Pi f_1 = x$ *iff* $\Phi \Vdash^+_\Pi f_2 = x$

26. *if* $f_1 \succeq^-_{su} f_2$ *and* $f_2 \succeq^-_{su} f_1$, *then* $\Phi \Vdash^+_\Pi f_1 = x$ *iff* $\Phi \Vdash^+_\Pi f_2 = x$

27. $\Phi \Vdash^+_\Pi f_1 > x$ *and* $x \geq y$ *implies* $\Phi \Vdash^+_\Pi f_1 > y$

28. $\Phi \Vdash^+_\Pi f_1 > x$ *and* $x > y$ *implies* $\Phi \Vdash^+_\Pi f_1 > y$

29. $\Phi \Vdash^+_\Pi f_1 \geq x$ *and* $x \geq y$ *implies* $\Phi \Vdash^+_\Pi f_1 \geq y$

30. $\Phi \Vdash^+_\Pi f_1 \geq x$ *and* $x > y$ *implies* $\Phi \Vdash^+_\Pi f_1 > y$

31. $\Phi \Vdash^+_\Pi f_1 < x$ *and* $x \leq y$ *implies* $\Phi \Vdash^+_\Pi f_1 < y$

32. $\Phi \Vdash^+_\Pi f_1 < x$ *and* $x < y$ *implies* $\Phi \Vdash^+_\Pi f_1 < y$

33. $\Phi \Vdash^+_\Pi f_1 \leq x$ *and* $x \leq y$ *implies* $\Phi \Vdash^+_\Pi f_1 \leq y$

34. $\Phi \Vdash^+_\Pi f_1 \leq x$ *and* $x < y$ *implies* $\Phi \Vdash^+_\Pi f_1 < y$

35. $\Phi \Vdash^{+}_{\Pi} f_1 = x \wedge f_1 = y$ *where* $x \neq y$ *iff* $\Phi \Vdash^{+}_{\Pi} \perp$

36. $\Phi \Vdash^{+}_{\Pi} f_1\#x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_2\#x$ *where* $f_1 \simeq_{re} f_2$

37. $\Phi \Vdash^{+}_{\Pi} f_1\#x$ *iff* $\Phi \Vdash^{+}_{\Pi} f_1 \star p(\perp)\#x$, *where* $\star \in \{+,-\}$

38. $\Phi \Vdash^{+}_{\Pi} f_1 > x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 + p(\gamma) > x$

39. $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 + p(\gamma) \geq x$

40. $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 - p(\gamma) \leq x$

41. $\Phi \Vdash^{+}_{\Pi} f_1 < x$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 - p(\gamma) < x$

42. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *and* $\Phi \Vdash^{+}_{\Pi} f_2 = y$ *and* $z = x + y \in \Pi$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 + f_2 = z$

43. $\Phi \Vdash^{+}_{\Pi} f_1 = x$ *and* $\Phi \Vdash^{+}_{\Pi} f_2 = y$ *and* $z = x - y \in \Pi$ *implies* $\Phi \Vdash^{+}_{\Pi} f_1 - p(\beta_1) \,\star_1 \ldots \star_{l-1}\, p(\beta_l) = z$, *where* $☆_i = +$ *if* $\star_i = -$ *and* $☆_i = -$ *if* $\star_i = +$

44. $\Phi \Vdash^{+}_{\Pi} p(\alpha) - p(\alpha) = 0$

45. $\Phi \Vdash^{+}_{\Pi} p(\alpha \vee \beta) = x$ *iff* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) = x$

46. $\Phi \Vdash^{+}_{\Pi} p(\alpha \wedge \beta) = x$ *iff* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \vee \beta) = x$

47. *if* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) < 0$ *or* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) > 1$ *then* $\Phi \Vdash^{+}_{\Pi} \perp$

48. *if* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \vee \beta) < 0$ *or* $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \vee \beta) > 1$ *then* $\Phi \Vdash^{+}_{\Pi} \perp$

Proof. 1. We prove that $\Phi \Vdash^{+}_{\Pi} p(\alpha) \geq 0$. We can observe that $\Phi \Vdash^{+}_{\Pi} \top$ by propositional rule P2. Thus, by combining the propositional rules and the basic rule B1, $\Phi \Vdash^{+}_{\Pi} p(\alpha) \geq 0$.

2. We can prove that $\Phi \Vdash^{+}_{\Pi} p(\alpha) \leq 1$ similarly to previous point.

3. We can prove that $\Phi \Vdash^{+}_{\Pi} p(\top) = 1$ similarly to previous points.

4. We can prove that $\Phi \Vdash^{+}_{\Pi} p(\perp) = 0$ similarly to previous points.

5. We prove that $\Phi \Vdash^{+}_{\Pi} f_1 > x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$. Based on enumeration rule E1, it holds that either $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{>}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ or $\Phi \Vdash^{+}_{\Pi} \perp$ if $\Pi^{x,\mathsf{AOp}(f_1)}_{>} = \varnothing$. If $\Phi \Vdash^{+}_{\Pi} \perp$, then based on propositional rule P1, $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$. Let us focus on the case where $\Pi^{x,\mathsf{AOp}(f_1)}_{>} \neq \varnothing$. We can observe that $\{\bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{>}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)\} \vdash \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{>}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k) \vee \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{=}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ and we can therefore use propositional rule P1 to show that $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{>}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k) \vee \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{=}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$. Hence, by using enumeration rule E1 once more we can show that $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$.

6. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 < x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ similarly to the previous point.

7. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 = x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ similarly to the previous point.

8. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 = x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ similarly to the previous point.

9. We prove that $\Phi \Vdash^{+}_{\Pi} f_1 \neq x$ iff $\Phi \Vdash^{+}_{\Pi} f_1 > x \vee f_1 < x$. Based on rule E1, $\Phi \Vdash^{+}_{\Pi} f_1 \neq x$ iff $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{\neq}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ (we observe that since Π is a reasonable restricted value set, then $\{0,1\} \subseteq \Pi$ and hence $\Pi^{x,\mathsf{AOp}(f_1)}_{\neq} \neq \varnothing$ by Proposition 4.8). It is easy to see that it is equivalent to

- $\Phi \Vdash^+_\Pi \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_>}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k) \vee \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_<}$ $(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ if $\Pi^{x,\mathsf{AOp}(f_1)}_> \neq \emptyset$ and $\Pi^{x,\mathsf{AOp}(f_1)}_< \neq \emptyset$, or
- $\Phi \Vdash^+_\Pi \bot \vee \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_<}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ if $\Pi^{x,\mathsf{AOp}(f_1)}_> = \emptyset$ and $\Pi^{x,\mathsf{AOp}(f_1)}_< \neq \emptyset$, or
- $\Phi \Vdash^+_\Pi \bot \vee \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_>}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ if $\Pi^{x,\mathsf{AOp}(f_1)}_> \neq \emptyset$ and $\Pi^{x,\mathsf{AOp}(f_1)}_< = \emptyset$.

We can observe that it cannot be the case that $\Pi^{x,\mathsf{AOp}(f_1)}_> = \Pi^{x,\mathsf{AOp}(f_1)}_< = \emptyset$, as we are dealing with a reasonable restricted value sets. Nevertheless, based on rule E1, this is equivalent to $\Phi \Vdash^+_\Pi f_1 > x \vee f_1 < x$. Therefore, $\Phi \Vdash^+_\Pi f_1 \neq x$ iff $\Phi \Vdash^+_\Pi f_1 > x \vee f_1 < x$.

10. We can prove that $\Phi \Vdash^+_\Pi f_1 \geq x$ iff $\Phi \Vdash^+_\Pi f_1 > x \vee f_1 = x$ similarly to the previous point.
11. We can prove that $\Phi \Vdash^+_\Pi f_1 \leq x$ iff $\Phi \Vdash^+_\Pi f_1 < x \vee f_1 = x$ similarly to the previous point.
12. $\Phi \Vdash^+_\Pi f_1 < x$ iff $\Phi \Vdash^+_\Pi f_1 \leq x \wedge f_1 \neq x$
13. $\Phi \Vdash^+_\Pi f_1 = x$ iff $\Phi \Vdash^+_\Pi f_1 \geq x \wedge f_1 \leq x$
14. We prove that $\Phi \Vdash^+_\Pi f_1 > x$ iff $\Phi \Vdash^+_\Pi f_1 \geq x \wedge f_1 \neq x$. As shown in the previous parts of this proof, $\Phi \Vdash^+_\Pi f_1 > x$ implies $\Phi \Vdash^+_\Pi f_1 \geq x$. Furthermore, $\{f_1 > x\} \vdash f_1 > x \vee f_1 < x$, hence $\Phi \Vdash^+_\Pi f_1 > x \vee f_1 < x$ by P1. By the previous parts of this proof it thus holds that $\Phi \Vdash^+_\Pi f_1 \neq x$. Consequently, we can use P1 again to show that if $\Phi \Vdash^+_\Pi f_1 > x$ then $\Phi \Vdash^+_\Pi f_1 \geq x \wedge f_1 \neq x$.

    Let now $\Phi \Vdash^+_\Pi f_1 \geq x \wedge f_1 \neq x$. Thus, by the previous points of this proof, $\Phi \Vdash^+_\Pi (f_1 > x \vee f_1 = x) \wedge (f_1 > x \vee f_1 < x)$. We can use rule E1 and rule P1 to show that $\Phi \Vdash^+_\Pi f_1 > x$. Hence, if $\Phi \Vdash^+_\Pi f_1 \geq x \wedge f_1 \neq x$ then $\Phi \Vdash^+_\Pi f_1 > x$ and this concludes the proof.
15. We can prove that $\Phi \Vdash^+_\Pi f_1 < x$ iff $\Phi \Vdash^+_\Pi f_1 \leq x \wedge f_1 \neq x$ similarly to the previous point.
16. We can prove that $\Phi \Vdash^+_\Pi f_1 = x$ iff $\Phi \Vdash^+_\Pi f_1 \geq x \wedge f_1 \leq x$ similarly to the previous point.
17. We can prove that $\Phi \Vdash^+_\Pi f_1 > x$ iff $\Phi \Vdash^+_\Pi \neg(f_1 \leq x)$ using enumeration rules E1 and E2.
18. We prove that $\Phi \Vdash^+_\Pi f_1 < x$ iff $\Phi \Vdash^+_\Pi \neg(f_1 \geq x)$ using enumeration rules E1 and E4.
19. We prove that $\Phi \Vdash^+_\Pi f_1 \leq x$ iff $\Phi \Vdash^+_\Pi \neg(f_1 > x)$ using enumeration rules E1 and E3.
20. We prove that $\Phi \Vdash^+_\Pi f_1 \geq x$ iff $\Phi \Vdash^+_\Pi \neg(f_1 < x)$ using enumeration rules E1 and E5.
21. We prove that $\Phi \Vdash^+_\Pi f_1 = x$ iff $\Phi \Vdash^+_\Pi \neg(f_1 \neq x)$. Using the propositional rules we can show that $\Phi \Vdash^+_\Pi f_1 = x$ iff $\Phi \Vdash^+_\Pi \neg(\neg(f_1 = x))$. By one of the previous points $\Phi \Vdash^+_\Pi \neg(\neg(f_1 = x))$ iff $\Phi \Vdash^+_\Pi \neg(\neg(f_1 \geq x \wedge f_1 \leq x))$, which by the propositional rules and previous parts of this proof is equivalent to $\Phi \Vdash^+_\Pi \neg(f_1 < x \vee f_1 > x))$ and thus to $\Phi \Vdash^+_\Pi \neg(f_1 \neq x)$.
22. We can prove that $\Phi \Vdash^+_\Pi f_1 \neq x$ iff $\Phi \Vdash^+_\Pi \neg(f_1 = x)$ using the previous point and the propositional rules.
23. We can prove that $\Phi \Vdash^+_\Pi f_1 = x$ and $f_1 \geq^+_{su} f_2$ implies $\Phi \Vdash^+_\Pi f_2 \geq x$ using the previously proved rule $\Phi \Vdash^+_\Pi f_1 \geq x$ iff $\Phi \Vdash^+_\Pi f_1 > x \vee f_1 = x$, subject rule S2 and the propositional rules.
24. We can prove that $\Phi \Vdash^+_\Pi f_1 = x$ and $f_1 \geq^-_{su} f_2$ implies $\Phi \Vdash^+_\Pi f_2 \leq x$ using the previously proved rule $\Phi \Vdash^+_\Pi f_1 \leq x$ iff $\Phi \Vdash^+_\Pi f_1 < x \vee f_1 = x$, subject rule S4 and the propositional rules.
25. We can prove that $\Phi \Vdash^+_\Pi f_2 = x$ and $f_1 \geq^+_{su} f_2$ implies $\Phi \Vdash^+_\Pi f_1 \leq x$ using the previously proved rule $\Phi \Vdash^+_\Pi f_2 \leq x$ iff $\Phi \Vdash^+_\Pi f_2 < x \vee f_2 = x$, subject rule S6 and the propositional rules.

26. We can prove that $\Phi \Vdash^{+}_{\Pi} f_2 = x$ and $f_1 \succeq^{-}_{su} f_2$ implies $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ using the previously proved rule $\Phi \Vdash^{+}_{\Pi} f_2 \geq x$ iff $\Phi \Vdash^{+}_{\Pi} f_2 > x \vee f_2 = x$, subject rule S8 and the propositional rules.

27. We now prove that if $f_1 \succeq^{+}_{su} f_2$ and $f_2 \succeq^{+}_{su} f_1$, then $\Phi \Vdash^{+}_{\Pi} f_1 = x$ iff $\Phi \Vdash^{+}_{\Pi} f_2 = x$. Using the previous points of this proof and the propositional rule P1, we can show that if $f_1 \succeq^{+}_{su} f_2$ and $f_2 \succeq^{+}_{su} f_1$, then $\Phi \Vdash^{+}_{\Pi} f_2 \geq x \wedge f_2 \leq x$. This, based on derivable rule 13, is equivalent to $\Phi \Vdash^{+}_{\Pi} f_2 = x$. The right to left direction can be proved in a similar fashion.

28. We can show that if $f_1 \succeq^{-}_{su} f_2$ and $f_2 \succeq^{-}_{su} f_1$, then $\Phi \Vdash^{+}_{\Pi} f_1 = x$ iff $\Phi \Vdash^{+}_{\Pi} f_2 = x$, similarly as the previous point.

29. We prove that $\Phi \Vdash^{+}_{\Pi} f_1 > x$ and $x \geq y$ implies $\Phi \Vdash^{+}_{\Pi} f_1 > y$. Based on enumeration rule E1, $\Phi \Vdash^{+}_{\Pi} f_1 > x$ iff $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ or $\Phi \Vdash^{+}_{\Pi} \perp$ when $\Pi^{x,\mathsf{AOp}(f_1)}_{>} = \varnothing$. If it is the latter case, then by using P1 we can show that if $\Phi \Vdash^{+}_{\Pi} \perp$ then $\Phi \Vdash^{+}_{\Pi} f_1 > y$. Let us focus on the former case. Since $\Pi^{x,\mathsf{AOp}(f_1)}_{>} \subseteq \Pi^{y,\mathsf{AOp}(f_1)}_{>}$, it is easy to show that $\{\bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)\} \vdash \bigvee_{(v_1,\ldots,v_k)\in\Pi^{y,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$. Consequently, it holds that $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{y,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ and therefore $\Phi \Vdash^{+}_{\Pi} f_1 > y$.

30. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 > x$ and $x > y$ implies $\Phi \Vdash^{+}_{\Pi} f_1 > y$ similarly to the previous point. We note that $\Pi^{x,\mathsf{AOp}(f_1)}_{>} \subset \Pi^{y,\mathsf{AOp}(f_1)}_{>}$.

31. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ and $x \geq y$ implies $\Phi \Vdash^{+}_{\Pi} f_1 \geq y$ similarly to the previous points.

32. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ and $x > y$ implies $\Phi \Vdash^{+}_{\Pi} f_1 > y$. We note that $\Pi^{x,\mathsf{AOp}(f_1)}_{\geq} \subset \Pi^{y,\mathsf{AOp}(f_1)}_{>}$.

33. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 < x$ and $x \leq y$ implies $\Phi \Vdash^{+}_{\Pi} f_1 < y$ similarly to the previous points.

34. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 < x$ and $x < y$ implies $\Phi \Vdash^{+}_{\Pi} f_1 < y$ similarly to the previous points.

35. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ and $x \leq y$ implies $\Phi \Vdash^{+}_{\Pi} f_1 \leq y$ similarly to the previous points.

36. We can prove that $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ and $x < y$ implies $\Phi \Vdash^{+}_{\Pi} f_1 < y$ similarly to the previous points.

37. We now prove that $\Phi \Vdash_{\Pi} f_1 = x \wedge f_1 = y$ where $x \neq y$ iff $\Phi \Vdash_{\Pi} \perp$. Through the use of enumeration rule E1, we can observe that $\Phi \Vdash_{\Pi} f_1 = x \wedge f_1 = y$ iff $\Phi \Vdash_{\Pi} (\bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{=}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)) \wedge (\bigvee_{(v'_1,\ldots,v'_k)\in\Pi^{y,\mathsf{AOp}(f_1)}_{=}} (p(\alpha_1) = v'_1 \wedge p(\alpha_2) = v'_2 \wedge \ldots \wedge p(\alpha_k) = v'_k))$ (we note that since Π is a reasonable restricted value set, $\{0,1\} \subseteq \Pi$ and neither $\Pi^{x,\mathsf{AOp}(f_1)}_{=}$ nor $\Pi^{y,\mathsf{AOp}(f_1)}_{=}$ are empty by Proposition 4.8). We can observe that since $x \neq y$, $Pi^{x,\mathsf{AOp}(f_1)}_{=} \cap \Pi^{y,\mathsf{AOp}(f_1)}_{=} = \varnothing$, and that for any conjunctive clause $(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ associated with x and for any conjunctive clause $(p(\alpha_1) = v'_1 \wedge p(\alpha_2) = v'_2 \wedge \ldots \wedge p(\alpha_k) = v'_k)$, there is $p(\alpha_i)$ s.t. $v_i \neq v'_i$. We can easily adapt the proof of Proposition 4.20 created for the basic system in order to show that if $v_i \neq v'_i$, then $p(\alpha_i) = v_i \wedge p(\alpha_i) = v'_i$ is equivalent to $\perp$. Thus, the conjunction of any x–clause and y–clause will , through the use of propositional domination laws, is equivalent to $\perp$. Hence, we can use the distributive, domination and identity laws[12] to show that $\Phi \Vdash_{\Pi} (\bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{=}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)) \wedge (\bigvee_{(v'_1,\ldots,v'_k)\in\Pi^{y,\mathsf{AOp}(f_1)}_{=}} (p(\alpha_1) = v'_1 \wedge p(\alpha_2) = v'_2 \wedge \ldots \wedge p(\alpha_k) = v'_k))$ iff $\Phi \Vdash^{+}_{\Pi} \perp$. Therefore, $\Phi \Vdash_{\Pi} f_1 = x \wedge f_1 = y$ where $x \neq y$ iff $\Phi \Vdash_{\Pi} \perp$.

[12] Recall that the identity and domination logical equivalence laws state that $q \wedge \top \equiv q$, $q \vee \perp \equiv q$, $q \vee \top \equiv \top$ and $q \wedge \perp \equiv \perp$.

38. We now prove that $\Phi \Vdash^{+}_{\Pi} f_1\#x$ iff $\Phi \Vdash^{+}_{\Pi} f_1'\#x$ where $f_1 \simeq_{re} f_1'$. Assume that $\Pi^{x,\mathsf{AOp}(f_1)}_{\#} \neq \varnothing$ and that $\Pi^{x,\mathsf{AOp}(f_1')}_{\#} \neq \varnothing$. Using enumeration rule E1, it holds that $\Phi \Vdash^{+}_{\Pi} f_1\#x$ iff $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ and $\Phi \Vdash^{+}_{\Pi} f_1'\#x$ iff $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1')}_{\#}}(p(\alpha_1') = v_1 \wedge p(\alpha_2') = v_2 \wedge \ldots \wedge p(\alpha_k') = v_k)$. Since $f_1 \simeq_{re} f_1'$, then we can show that $\{\bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)\} \vdash \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1')}_{\#}}(p(\alpha_1') = v_1 \wedge p(\alpha_2') = v_2 \wedge \ldots \wedge p(\alpha_k') = v_k)$ and $\{\bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1')}_{\#}}(p(\alpha_1') = v_1 \wedge p(\alpha_2') = v_2 \wedge \ldots \wedge p(\alpha_k') = v_k)\} \vdash \Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ based on the commutative law. Hence, based on P1 and E1, $\Phi \Vdash^{+}_{\Pi} f_1\#x$ iff $\Phi \Vdash^{+}_{\Pi} f_1'\#x$ where $f_1 \simeq_{re} f_1'$. Assume that $\Pi^{x,\mathsf{AOp}(f_1)}_{\#} = \varnothing$ and that $\Pi^{x,\mathsf{AOp}(f_1')}_{\#} = \varnothing$. Then $\Phi \Vdash^{+}_{\Pi} f_1\#x$ iff $\Phi \Vdash^{+}_{\Pi} \bot$ and $\Phi \Vdash^{+}_{\Pi} f_1'\#x$ iff $\Phi \Vdash^{+}_{\Pi} \bot$. Thus, $\Phi \Vdash^{+}_{\Pi} f_1\#x$ iff $\Phi \Vdash^{+}_{\Pi} f_1'\#x$. Finally, given the fact that $f_1 \simeq_{re} f_1'$, we can show that $\Pi^{x,\mathsf{AOp}(f_1)}_{\#} = \varnothing$ iff $\Pi^{x,\mathsf{AOp}(f_1')}_{\#} = \varnothing$ by considering the cases from Proposition 4.8. If $\mathsf{AOp}(f_1) = 0$, then f_1 and f_2 have to be the same operational formulae. Furthermore, if $\mathsf{AOp}(f_1)$ contains a $+$ $(-)$ then so does $\mathsf{AOp}(f_1')$ and vice versa. Finally, since Π is reasonable, then $\{0,1\} \subseteq \Pi$. This concludes our proof.

39. We now show that $\Phi \Vdash^{+}_{\Pi} f_1\#x$ iff $\Phi \Vdash^{+}_{\Pi} f_1 \star p(\bot)\#x$, where $\star \in \{+,-\}$. Let us start with the left to right direction. Based on enumeration rule E1, $\Phi \Vdash f_1\#x$ iff $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ or $\Phi \Vdash^{+}_{\Pi} \bot$ if $\Pi^{x,\mathsf{AOp}(f_1)}_{\#} = \varnothing$. Assume $\Pi^{x,\mathsf{AOp}(f_1)}_{\#} \neq \varnothing$. Based on the fourth derivable rule $\Phi \Vdash p(\bot) = 0$. Hence, through the use of propositional rules, $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ iff $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k)\in\Pi^{x,\mathsf{AOp}(f_1)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k \wedge p(\bot) = 0)$. This is equivalently expressed with $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k,0)\in\Pi^{x,(*_1,\ldots,*_{k-1},\star)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k \wedge p(\bot) = 0)$. Since any conjunctive clause s.t. $p(\bot) = v$ and $v \neq y$ is equivalent to $\bot$ through the propositional and derivable rules, then through the use of propositional rules again we can show that $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k,v_{k+1})\in\Pi^{x,(*_1,\ldots,*_{k-1},\star)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k \wedge p(\bot) = v_{k+1})$. This, given the enumeration rule E1, is equivalent to $f_1 \star p(\bot)\#x$, where $\star \in \{+,-\}$. The right to left direction follows in a similar fashion.

We can now consider the case where $\Pi^{x,\mathsf{AOp}(f_1)}_{\#} = \varnothing$. Thus, $\Phi \Vdash^{+}_{\Pi} f_1\#x$ iff $\Phi \Vdash^{+}_{\Pi} \bot$. Consequently, through the use of propositional rules, if $\Phi \Vdash^{+}_{\Pi} \bot$, then $\Phi \Vdash^{+}_{\Pi} f_1 \star p(\bot)\#x$, where $\star \in \{+,-\}$. Let us therefore focus on the right to left direction. If $\Pi^{x,(*_1,\ldots,*_{k-1},\star)}_{\#} = \varnothing$ as well, then through the use of enumeration rule E1 $\Phi \Vdash^{+}_{\Pi} f_1 \star p(\bot)\#x$ implies $\Phi \Vdash^{+}_{\Pi} \bot$, which leads to $\Phi \Vdash^{+}_{\Pi} f_1\#x$. Hence, assume $\Pi^{x,(*_1,\ldots,*_{k-1},\star)}_{\#} \neq \varnothing$. Thus, through the use of E1, $\Phi \Vdash^{+}_{\Pi} f_1 \star p(\bot)\#x$ iff $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k,v_{k+1})\in\Pi^{x,(*_1,\ldots,*_{k-1},\star)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k \wedge p(\bot) = v_{k+1})$. Since $\Pi^{x,\mathsf{AOp}(f_1)}_{\#} = \varnothing$, then there is no sequence of values $(v_1,\ldots,v_k)$ s.t. $v_1 *_1 \ldots *_{k-1} v_k\#x$. Clearly, this means that there is no sequence of values $(v_1,\ldots,v_k,0)$ s.t. $v_1 *_1 \ldots *_{k-1} v_k \star 0\#x$. This means that for any $(v_1,\ldots,v_k,v_{k+1}) \in \Pi^{x,(*_1,\ldots,*_{k-1},\star)}_{\#}$, $v_{k+1} \neq 0$. Hence, any conjunctive clause in the previous formula contains a basic atom $p(\bot) = v_{k+1}$ which through the use of basic rule B4, derivable rule 35 and propositional rules can be shown to be equivalent to $\bot$. Hence, if $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,\ldots,v_k,v_{k+1})\in\Pi^{x,(*_1,\ldots,*_{k-1},\star)}_{\#}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k \wedge p(\bot) = v_{k+1})$ then $\Phi \Vdash^{+}_{\Pi} \bot$ and therefore $\Phi \Vdash^{+}_{\Pi} f_1\#x$, which concludes our proof.

40. We now show that $\Phi \Vdash^{+}_{\Pi} f_1 > x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 + p(\gamma) > x$. Based on the previous parts of this proof, $\Phi \Vdash^{+}_{\Pi} f_1 > x$ iff $\Phi \Vdash^{+}_{\Pi} f_1 + p(\bot) > x$. Since $\{\bot\} \vdash \gamma$ for any γ, then $f_1 + p(\bot) \geq^{+}_{su} f_1 + p(\gamma)$. Hence, based on subject rule S1, $\Phi \Vdash^{+}_{\Pi} f_1 + p(\bot) > x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 + p(\gamma) > x$ and therefore

$\Phi \Vdash^{+}_{\Pi} f_1 > x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 + p(\gamma) > x$.

41. We can show that $\Phi \Vdash^{+}_{\Pi} f_1 \geq x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 + p(\gamma) \geq x$ similarly as in the previous point.

42. We can show that $\Phi \Vdash^{+}_{\Pi} f_1 \leq x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 - p(\gamma) \leq x$ similarly as in the previous point.

43. We can show that $\Phi \Vdash^{+}_{\Pi} f_1 < x$ implies $\Phi \Vdash^{+}_{\Pi} f_1 - p(\gamma) < x$ similarly as in the previous point.

44. We show that if $\Phi \Vdash^{+}_{\Pi} f_1 = x$ and $\Phi \Vdash^{+}_{\Pi} f_2 = y$ and $z = x + y \in \Pi$, then $\Phi \Vdash^{+}_{\Pi} f_1 + f_2 = z$. We observe that based on Proposition 4.8, $\Pi_{=}^{x,\mathsf{AOp}(f_1)} \neq \varnothing$ and $\Pi_{=}^{y,\mathsf{AOp}(f_2)} \neq \varnothing$. Since $\Phi \Vdash^{+}_{\Pi} f_1 = x$ and $\Phi \Vdash^{+}_{\Pi} f_2 = x$, then through the use of enumeration rule E1 and propositional rule P1, $\Phi \Vdash^{+}_{\Pi} (\bigvee_{(v_1,\ldots,v_k)\in\Pi_{=}^{x,\mathsf{AOp}(f_1)}}(p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)) \wedge (\bigvee_{(v'_1,\ldots,v'_l)\in\Pi_{=}^{y,\mathsf{AOp}(f_2)}}(p(\beta_1) = v'_1 \wedge p(\beta_2) = v'_2 \wedge \ldots \wedge p(\beta_l) = v'_l))$. Through the application of propositional rule P1 (in particular, through distributive laws), we see that $\Phi \Vdash^{+}_{\Pi} \phi_1 \vee \ldots \vee \phi_{k*l}$, where every ϕ_i is of the form $p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k \wedge p(\beta_1) = v'_1 \wedge p(\beta_2) = v'_2 \wedge \ldots \wedge p(\beta_l) = v'_l$ for some $(v_1,\ldots,v_k) \in \Pi_{=}^{x,\mathsf{AOp}(f_1)}$ and $(v'_1,\ldots,v'_l) \in \Pi_{=}^{y,\mathsf{AOp}(f_2)}$. We can observe that since $z = x + y$ is in Π, then for all of the aforementioned sequences $(v_1,\ldots,v_k)$ and $(v'_1,\ldots,v'_l)$, $(v_1,\ldots,v_k,v'_1,\ldots,v'_l) \in \Pi_{=}^{z,\mathsf{AOp}(f_1+f_2)}$. Consequently, we can show that $\{\phi_1 \vee \ldots \vee \phi_{k*l}\} \vdash \bigvee_{(\tau_1,\ldots,\tau_{k+l})\in\Pi_{=}^{z,\mathsf{AOp}(f_1+f_2)}}(p(\alpha_1) = \tau_1 \wedge p(\alpha_2) = \tau_2 \wedge \ldots \wedge p(\alpha_k) = \tau_k \wedge p(\beta_1) = \tau_{k+1} \wedge p(\beta_2) = \tau_{k+2} \wedge \ldots \wedge p(\beta_l) = \tau_{k+l})$. Hence, through the use of propositional rule P1 and again enumeration rule E1, we can show that $\Phi \Vdash^{+}_{\Pi} f_1 + f_2 = z$.

45. The fact that $\Phi \Vdash^{+}_{\Pi} f_1 = x$ and $\Phi \Vdash^{+}_{\Pi} f_2 = y$ and $z = x - y \in \Pi$ implies $\Phi \Vdash^{+}_{\Pi} f_1 - p(\beta_1) \star_1 p(\beta_2) \star_2 \ldots \star_{l-1} p(\beta_l) = z$, where $\star_i = +$ if $\star_i = -$ and $\star_i = -$ if $\star_i = +$, can be shown similarly to the previous point.

46. We now show that $\Phi \Vdash^{+}_{\Pi} p(\alpha) - p(\alpha) = 0$. Based on the first derivable rule, $\Phi \Vdash^{+}_{\Pi} p(\alpha) \geq 0$, which through the use of the enumeration rule and the properties of Π we can show to be equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{v\in\Pi} p(\alpha) = v$. Through the use of the propositional rule P1, this is also equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{v\in\Pi} p(\alpha) = v \wedge p(\alpha) = v$. We can then use derivable rule 35 and propositional rule P1 (in particular, identity laws) to show that this formula is equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,v_2)\in\Pi_{=}^{0,(-)}} p(\alpha) = v_1 \wedge p(\alpha) = v_2$, which by enumeration rule E1 is simply $\Phi \Vdash^{+}_{\Pi} p(\alpha) - p(\alpha) = 0$.

47. We show that $\Phi \Vdash^{+}_{\Pi} p(\alpha \vee \beta) = x$ iff $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) = x$. Let us consider the left to right direction and assume that $\Phi \Vdash^{+}_{\Pi} p(\alpha \vee \beta) = x$. Based on the probabilistic rule PR1, $\Phi \Vdash^{+}_{\Pi} p(\alpha \vee \beta) - p(\alpha) - p(\beta) + p(\alpha \wedge \beta) = 0$, which using enumeration rule E1 can be written as $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,v_2,v_3,v_4)\in\Pi_{=}^{0,(-,-,+)}}(p(\alpha \vee \beta) = v_1 \wedge p(\alpha) = v_2 \wedge p(\beta) = v_3 \wedge p(\alpha \wedge \beta) = v_4)$ (we observe that based on Proposition 4.8, $\Pi_{=}^{0,(-,-,+)} \neq \varnothing$). Through the use of propositional rule P1, $\Phi \Vdash^{+}_{\Pi} p(\alpha\vee\beta) = x \wedge \bigvee_{(v_1,v_2,v_3,v_4)\in\Pi_{=}^{0,(-,-,+)}}(p(\alpha\vee\beta) = v_1 \wedge p(\alpha) = v_2 \wedge p(\beta) = v_3 \wedge p(\alpha\wedge\beta) = v_4)$. Through the use of propositional rule (in particular, distributive and identity laws) and derivable rule 35, we can observe that the above formula is equivalent to equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{(x,v_2,v_3,v_4)\in\Pi_{=}^{0,(-,-,+)}}(p(\alpha \vee \beta) = x \wedge p(\alpha) = v_2 \wedge p(\beta) = v_3 \wedge p(\alpha \wedge \beta) = v_4)$. This can be further shown to be equivalent to $\Phi \Vdash^{+}_{\Pi} p(\alpha \vee \beta) = x \wedge \bigvee_{(v_1,v_2,v_3)\in\Pi_{=}^{x,(+,-)}}(p(\alpha) = v_1 \wedge p(\beta) = v_2 \wedge p(\alpha \wedge \beta) = v_3)$ and therefore to $\Phi \Vdash^{+}_{\Pi} p(\alpha \vee \beta) = x \wedge (p(\alpha) + p(\beta) - p(\alpha \wedge \beta) = x)$ through the use of enumeration rule E1. Hence, by P1, $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) = x$. The right to left direction can be proved in a similar fashion.

48. We can show that $\Phi \Vdash^{+}_{\Pi} p(\alpha \wedge \beta) = x$ iff $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \vee \beta) = x$ in the same way as the previous point of this proof.

49. We now show that if $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) < 0$ or $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) > 1$ then $\Phi \Vdash^{+}_{\Pi} \perp$. Let us consider the first case. Based on derivable rule 1, $\Phi \Vdash^{+}_{\Pi} p(\alpha \wedge \beta) \geq 0$, which through the use of enumeration rule E1 and the definition of the combination sets we can show to be equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{v\in\Pi} p(\alpha \wedge \beta) = v$ (we observe that based on Proposition 4.8, $\Pi_{\geq}^{0} \neq \varnothing$). Hence,

using derivable rule 46, $\Phi \Vdash^{+}_{\Pi} \bigvee_{v \in \Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) = v$. Through the use of enumeration and propositional rules we can show that this implies that $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) \geq 0$. Since $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) < 0$ by assumption, which by derivable rule 16 means that $\Phi \Vdash^{+}_{\Pi} \neg(p(\alpha) + p(\beta) - p(\alpha \wedge \beta) \geq 0)$, we can use the propositional rule P1 to show that $\Phi \Vdash^{+}_{\Pi} \bot$.

Let us now consider the second case. Based on derivable rule 1, $\Phi \Vdash^{+}_{\Pi} p(\alpha \wedge \beta) \leq 1$, which through the use of enumeration rule E1 and the definition of the combination sets we can show to be equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{v \in \Pi} p(\alpha \wedge \beta) = v$. We can repeat the previous reasoning to show that this implies that $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) \leq 1$. Since $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \wedge \beta) > 1$ by assumption, which by derivable rule 16 means that $\Phi \Vdash^{+}_{\Pi} \neg(p(\alpha) + p(\beta) - p(\alpha \wedge \beta) \leq 1)$, we can use the propositional rule P1 to show that $\Phi \Vdash^{+}_{\Pi} \bot$.

50. We can show that if $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \vee \beta) < 0$ or $\Phi \Vdash^{+}_{\Pi} p(\alpha) + p(\beta) - p(\alpha \vee \beta) > 1$ then $\Phi \Vdash^{+}_{\Pi} \bot$ similarly to the previous point of this proof.

□

Proposition 4.32. *Let Π be a restricted value set. For $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$, and $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$, if $\Phi \Vdash^{+}_{\Pi} \psi$ then $\Phi \Vdash\!\!\models_{\Pi} \psi$.*

Proof. We can show that each proof rule is sound. We first consider the **basic rules**:

- Consider proof rule 1. We need to show that $\Phi \Vdash\!\!\models_{\Pi} p(\alpha) \geq 0$ iff $\Phi \Vdash\!\!\models_{\Pi} \top$. We can observe that $\mathsf{Sat}(\top, \Pi) = \mathsf{Dist}(\Pi)$. Furthermore, by definition, $\mathsf{Sat}(p(\alpha) \geq 0, \Pi) = \{P' \in \mathsf{Dist}(\mathcal{G}) \mid P'(\alpha) \geq 0\} \cap \mathsf{Dist}(\Pi)$. It is easy to see that $\{P' \in \mathsf{Dist}(\mathcal{G}) \mid P'(\alpha) \geq 0\} = \mathsf{Dist}(\mathcal{G})$ for any α. Since $\mathsf{Dist}(\Pi) \subseteq \mathsf{Dist}(\mathcal{G})$, $\mathsf{Sat}(p(\alpha) \geq 0, \Pi) = \mathsf{Dist}(\Pi) = \mathsf{Sat}(\top, \Pi)$. Thus, we can show that $\Phi \Vdash\!\!\models_{\Pi} p(\alpha) \geq 0$ iff $\Phi \Vdash\!\!\models_{\Pi} \top$.
- Proof rules 2 to 4 can be proved in a similar fashion.

We now consider the **enumeration rules**:

- Consider proof rule 1. We need to show that $\Phi \Vdash\!\!\models_{\Pi} f_1 \# x$ iff $\Phi \Vdash\!\!\models_{\Pi} \bigvee_{(v_1,\ldots,v_k) \in \Pi^{x,\mathsf{AOp}(f_1)}_{\#}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$ if $\Pi^{x,\mathsf{AOp}(f_1)}_{\#} \neq \varnothing$ and $\Phi \Vdash\!\!\models_{\Pi} \bot$ otherwise. We first consider $\#$ being $>$ and start with the case where $\Pi^{x,\mathsf{AOp}(f_1)}_{>} = \varnothing$. Based on Proposition 4.8 and the fact that $\{0, 1\} \subseteq \Pi$ (note that Π is a reasonable restricted value set), it therefore holds that $x = 1$ and either $\mathsf{AOp}(f_1) = ()$ or for no $*_i$, $*_i = +$. If $\mathsf{AOp}(f_1) = ()$, then $f_1 : p(\alpha_1) > 1$ and it is easy to see that $\mathsf{Sat}(p(\alpha_1) > 1, \Pi) = \varnothing = \mathsf{Sat}(\bot, \Pi)$. If for every $*_i$, $*_i = -$, then based on the fact that probabilities belong to the unit interval, $p(\alpha_1) - p(\alpha_2) - \ldots - p(\alpha_k) \geq p(\alpha_1)$. Hence, if $\mathsf{Sat}(p(\alpha_1) > 1, \Pi) = \varnothing$, then $\mathsf{Sat}(f_1 > 1, \Pi) = \varnothing$ as well. Thus, if $\Pi^{x,\mathsf{AOp}(f_1)}_{>} = \varnothing$, $\Phi \Vdash\!\!\models_{\Pi} f_1 \# x$ iff $\Phi \Vdash\!\!\models_{\Pi} \bot$.

  Now consider the case where $\Pi^{x,\mathsf{AOp}(f_1)}_{\#} \neq \varnothing$. For every $P' \in \mathsf{Sat}(f_1 > x, \Pi)$, $P'(\alpha_1) *_1 P'(\alpha_2) *_2 \ldots *_{k-1} P'(\alpha_k) > x$. Consequently, $(P'(\alpha_1), \ldots, P'(\alpha_k)) \in \Pi^{x,\mathsf{AOp}(f_1)}_{>}$. We can therefore show that $\mathsf{Sat}(f_1 > x, \Pi) \subseteq \mathsf{Sat}(\bigvee_{(v_1,\ldots,v_k) \in \Pi^{x,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k), \Pi)$. Let now $P' \in \mathsf{Sat}(\bigvee_{(v_1,\ldots,v_k) \in \Pi^{x,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k), \Pi)$. Based on the properties of Sat, it means that there is $(v_1, \ldots, v_k) \in \Pi^{x,\mathsf{AOp}(f_1)}_{>}$ s.t. $P' \in \mathsf{Sat}((p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k), \Pi)$. Since $v_1 *_1 v_2 *_2 \ldots *_{k-1} v_k > x$, then $P'(\alpha_1) *_1 P'(\alpha_2) *_2 \ldots *_{k-1} P'(\alpha_k) > x$. Hence, $P' \in \mathsf{Sat}(f_1 > x, \Pi)$, and we can show that $\mathsf{Sat}(\bigvee_{(v_1,\ldots,v_k) \in \Pi^{x,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k), \Pi) \subseteq \mathsf{Sat}(f_1 > x, \Pi)$. Given the previous result, $\mathsf{Sat}(f_1 > x, \Pi) = \mathsf{Sat}(\bigvee_{(v_1,\ldots,v_k) \in \Pi^{x,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k), \Pi)$ and therefore $\Phi \Vdash\!\!\models_{\Pi} f_1 > x$ iff $\Phi \Vdash\!\!\models_{\Pi} \bigvee_{(v_1,\ldots,v_k) \in \Pi^{x,\mathsf{AOp}(f_1)}_{>}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)$. The results for other operators can be obtained in a similar fashion.
- Consider proof rule 2. We need to show that $\Phi \Vdash\!\!\models_{\Pi} f_1 > x$ iff $\Phi \Vdash\!\!\models_{\Pi} \neg(\bigvee_{(v_1,\ldots,v_k) \in \Pi^{x,\mathsf{AOp}(f_1)}_{\leq}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k))$. For every $P' \in \mathsf{Sat}(f_1 > x, \Pi)$, it holds that $P'(\alpha_1) *_1 P'(\alpha_2) *_2$

$\ldots *_{k-1} P'(\alpha_k) > x$. Consequently, we can observe that $(P'(\alpha_1), \ldots, P'(\alpha_k)) \in \Pi_{>}^{x,\mathsf{AOp}(f_1)}$ and $(P'(\alpha_1), \ldots, P'(\alpha_k)) \notin \Pi_{\leq}^{x,\mathsf{AOp}(f_1)}$. Hence, it holds that

$$\mathsf{Sat}(f_1 > x, \Pi) \subseteq \mathsf{Sat}(\top, \Pi) \setminus \mathsf{Sat}\Big(\bigvee_{(v_1,\ldots,v_k)\in\Pi_{\leq}^{x,\mathsf{AOp}(f_1)}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k), \Pi\Big) =$$
$$\mathsf{Sat}\Big(\neg\Big(\bigvee_{(v_1,\ldots,v_k)\in\Pi_{\leq}^{x,\mathsf{AOp}(f_1)}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)\Big), \Pi\Big)$$

Let now $P' \in \mathsf{Sat}(\neg(\bigvee_{(v_1,\ldots,v_k)\in\Pi_{\leq}^{x,\mathsf{AOp}(f_1)}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)), \Pi)$. This means that $P' \in \mathsf{Sat}(\top, \Pi) \setminus \mathsf{Sat}(\bigvee_{(v_1,\ldots,v_k)\in\Pi_{\leq}^{x,\mathsf{AOp}(f_1)}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k), \Pi)$. Hence, for every P', $P'(\alpha_1) *_1 P'(\alpha_2) *_2 \ldots *_{k-1} P'(\alpha_k) > x$ and therefore $P' \in \mathsf{Sat}(f_1 > x, \Pi)$. Given the previous result, this means that $\mathsf{Sat}(f_1 > x, \Pi) = \mathsf{Sat}(\neg(\bigvee_{(v_1,\ldots,v_k)\in\Pi_{\leq}^{x,\mathsf{AOp}(f_1)}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k)), \Pi)$. Hence, $\Phi \Vdash_{\Pi} f_1 > x$ iff $\Phi \Vdash_{\Pi} \neg(\bigvee_{(v_1,\ldots,v_k)\in\Pi_{\leq}^{x,\mathsf{AOp}(f_1)}} (p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge \ldots \wedge p(\alpha_k) = v_k))$.

- Soundness for the proof rules 3 to 5 is obtained in the same way as for rule 2. For rule 3, we observe that if $\Pi_{<}^{x,\mathsf{AOp}(f_1)} = \emptyset$, then based on the properties of Π, $x = 0$ and either $\mathsf{AOp}(f_1) = ()$ or for no $*_i$, $*_i = -$. If $\mathsf{AOp}(f_1) = ()$, then $f_1 : p(\alpha_1) \geq 0$ is a tautology based on the basic rules (and is therefore equivalent to $\neg\bot$). If for every $*_i$, $*_i = +$, then we can observe that $p(\alpha_1) \leq p(\alpha_1) + \ldots + p(\alpha_k)$, hence $\mathsf{Sat}(p(\alpha_1) + \ldots + p(\alpha_k) \geq 0, \Pi) = \mathsf{Sat}(\top, \Pi) = \mathsf{Sat}(\neg\bot)$. Similar observations can be made for rule 5.

The soundness of the **subject rules** can be easily proved by using Proposition 4.29.

We now consider the **probabilistic rule**. We can observe that for any probability distribution P and terms $\alpha, \beta \in \mathsf{Terms}(\mathcal{G})$, $P(\alpha \vee \beta) = P(\alpha) + P(\beta) - P(\alpha \wedge \beta)$. This can be easily checked by analyzing the definition of the probability of a term (Definition 3.2). Thus, it holds that $\mathsf{Sat}(p(\alpha \vee \beta) - p(\alpha) - p(\beta) + p(\alpha \wedge \beta) = 0, \Pi) = \mathsf{Dist}(\mathcal{G}, \Pi)$. Since for any set of probabilistic formulae Φ, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Dist}(\mathcal{G}, \Pi)$, it holds that $\Phi \Vdash_{\Pi} p(\alpha \vee \beta) - p(\alpha) - p(\beta) + p(\alpha \wedge \beta) = 0$.

We now consider the **propositional rules**.

- Assume $\Phi \Vdash_{\Pi} \phi_1$ and .... and $\Phi \Vdash_{\Pi} \phi_n$ and $\{\phi_1, \ldots, \phi_n\} \vdash \psi$. So for each $i \in \{1, \ldots, n\}$, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Sat}(\phi_i, \Pi)$. Furthermore, based on the definition of Sat, if $\{\phi_1, \ldots, \phi_n\} \vdash \psi$, then it holds that $\mathsf{Sat}(\{\phi_1, \ldots, \phi_n\}) \subseteq \mathsf{Sat}(\psi)$. Hence, $\mathsf{Sat}(\{\phi_1, \ldots, \phi_n\}, \Pi) \subseteq \mathsf{Sat}(\psi, \Pi)$ as well. Therefore, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Sat}(\psi, \Pi)$. Hence, $\Phi \Vdash_{\Pi} \psi$.

- Assume $\Phi \vdash \varphi$. Consequently, $\mathsf{Sat}(\Phi) \subseteq \mathsf{Sat}(\varphi)$. Hence, $\mathsf{Sat}(\Phi, \Pi) \subseteq \mathsf{Sat}(\varphi, \Pi)$ as well. Therefore, $\Phi \Vdash_{\Pi} \varphi$.

We have therefore shown that every rule in our system is sound. Hence, the system is sound as well. □

Proposition 4.33. *Let $\# \in \{=, \neq, \geq, \leq, >, <\}$ be the set of inequality relationships, let Π be a reasonable restricted value set, and let $\Pi_{\#}^{x} = \{y \in \Pi \mid x\#y\}$ be the subset obtained according to the value x and relationship $\#$. Let $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \ldots, c_j\}$ be the collection of all argument complete propositional formulae for $\mathcal{G}$ and $T_{v,k}^{\Pi} = \Pi_{=}^{v,(+,\ldots,+)}$ s.t. the length of $(+, \ldots, +)$ is $k-1$ be the collection of k–tuples of values from Π that sum up to $v \in \Pi$. The following hold, where $\vdash$ is propositional consequence relation, $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$, $\phi, \psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$ and $x \in \Pi$. The following hold:*

1. *for consistent formulae $\alpha_1, \ldots \alpha_m \in \mathsf{Terms}(\mathcal{G})$, if for all $1 \leq i, j \leq m$ s.t. $i \neq j$ it holds that $\alpha_i \wedge \alpha_j \vdash \bot$, then $\Phi \Vdash_{\Pi}^{+} p(\alpha_1 \vee \ldots \vee \alpha_m) = x$ iff $\Phi \Vdash_{\Pi}^{+} \bigvee_{(\tau_1,\ldots,\tau_m)\in T_{x,m}^{\Pi}} (p(\alpha_1) = \tau_1 \wedge \ldots \wedge p(\alpha_m) = \tau_m)$*

2. *$\Phi \Vdash_{\Pi}^{+} \bigvee_{(\tau_1,\ldots,\tau_j)\in T_{1,j}^{\Pi}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$*

3. *$\Phi \Vdash_{\Pi}^{+} \psi$ iff $\Phi \Vdash_{\Pi}^{+} \varphi$, where φ is the distribution disjunctive normal form of ψ*

Proof. 1. We prove this property by induction. We show the cases for $m = 1$ and, for clarity, $m = 2$, and then prove that if the property holds for $m = k$, then it holds for $m = k + 1$ as well.

- Let us start with $m = 1$. Since $T^{\Pi}_{v,1} = \Pi^{v,()}_{=}$, the rule states that $\Phi \Vdash^{+}_{\Pi} p(\alpha_1) = v$ iff $\Phi \Vdash^{+}_{\Pi} p(\alpha_1) = v$, which is clearly true.
- Let $m = 2$. We start with the left to right direction. By derivable rule 45, it holds that $\Phi \Vdash^{+}_{\Pi} p(\alpha_1 \vee \alpha_2) = x$ iff $\Phi \Vdash^{+}_{\Pi} p(\alpha_1 \wedge \alpha_2) - p(\alpha_1) - p(\alpha_2) = x$. By enumeration rule E1, this is equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,v_2,v_3)\in\Pi^{x,(+,-)}_{=}} (\wedge p(\alpha_1) = v_1 \wedge p(\alpha_2) = v_2 \wedge p(\alpha_1 \wedge \alpha_2) = v_3)$ (we note that based on Proposition 4.8 and the properties of Π, $\Pi^{x,(+,-)}_{=} \neq \varnothing$). From this point on we can proceed as in the proof of Proposition 4.22 in order to show that since $\alpha_1 \wedge \alpha_2 \vdash \bot$ by assumption, then this is equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{(\tau_1,\tau_2)\in T^{\Pi}_{x,2}} (p(\alpha_1) = \tau_1 \wedge p(\alpha_2) = \tau_2)$ and that our property holds.
- Assume that for $m = k$ our property is true, i.e. if our assumptions hold, then $\Phi \Vdash^{+}_{\Pi} p(\alpha_1 \vee \ldots \vee \alpha_k) = x$ iff $\Phi \Vdash^{+}_{\Pi} \bigvee_{(\tau_1,\ldots,\tau_k)\in T^{\Pi}_{x,k}} (p(\alpha_1) = \tau_1 \wedge \ldots \wedge p(\alpha_k) = \tau_k)$. We need to show that for $m = k + 1$, if our assumptions hold, then $\Phi \Vdash^{+}_{\Pi} p(\alpha_1 \vee \ldots \vee \alpha_{k+1}) = x$ iff $\Phi \Vdash^{+}_{\Pi} \bigvee_{(\tau_1,\ldots,\tau_{k+1})\in T^{\Pi}_{x,k+1}} (p(\alpha_1) = \tau_1 \wedge \ldots \wedge p(\alpha_{k+1}) = \tau_{k+1})$. We focus on the left to right direction first. By derivable rule 45, it holds that $\Phi \Vdash^{+}_{\Pi} p(\alpha_1 \vee \ldots \vee \alpha_{k+1}) = x$ iff $\Phi \Vdash^{+}_{\Pi} p(\alpha_1 \vee \ldots \vee \alpha_k) + p(\alpha_{k+1}) - p((\alpha_1 \vee \ldots \vee \alpha_k) \wedge \alpha_{k+1}) = x$. By enumeration rule E1, this is equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{(v_1,v_2,v_3)\in\Pi^{x,(+,-)}_{=}} (p(\alpha_1 \vee \ldots \vee \alpha_k) = v_1 \wedge p(\alpha_{k+1}) = v_2 \wedge p((\alpha_1 \vee \ldots \vee \alpha_k) \wedge \alpha_{k+1}) = v_3)$ (we note that based on Proposition 4.8 and the properties of Π, $\Pi^{x,(+,-)}_{=} \neq \varnothing$). . From this point on we can proceed as in the proof of Proposition 4.22 to show that since for every α_i, α_j s.t. $i \neq j$, $\alpha_i \wedge \alpha_j \vdash \bot$, then our formula is equivalent to $\Phi \Vdash^{+}_{\Pi} \bigvee_{(\tau_1,\ldots,\tau_{k+1})\in T^{\Pi}_{v,k+1}} (p(\alpha_1) = \tau_1 \wedge \ldots \wedge p(\alpha_{k+1}) = \tau_{k+1})$ and that our property indeed holds.

2. The fact that $\Phi \Vdash^{+}_{\Pi} \bigvee_{(\tau_1,\ldots,\tau_j)\in T^{\Pi}_{1,j}} (p(c_1) = \tau_1 \wedge \ldots \wedge p(c_j) = \tau_j)$ can be shown similarly as in Proposition 4.22.

3. We now show that $\Phi \Vdash^{+}_{\Pi} \psi$ iff $\Phi \Vdash^{+}_{\Pi} \varphi$, where φ is the distribution disjunctive normal form of ψ. Let $\mathsf{AComplete}(\mathcal{G}) = \{c_1, \ldots, c_j\}$ be the collection of all argument complete terms for $\mathcal{G}$. First of all, we can observe that for any term $\alpha \in \mathsf{Terms}(\mathcal{G})$, we can find an equivalent formula that is either $\bot$ if α is inconsistent, or which is equivalent to $c'_1 \vee \ldots \vee c'_k$, where $C' = \{c'_1, \ldots, c'_k\} \subseteq \mathsf{AComplete}(\mathcal{G})$ is a nonempty collection of argument complete terms. This form of α is easily found by constructing (propositional) DNF using the truth table method. Let α' be this form of α. Through the use of derivable rule 25, it is easy to show that $\Phi \Vdash^{+}_{\Pi} p(\alpha) = x$ iff $\Phi \Vdash^{+}_{\Pi} p(\alpha') = x$. It therefore suffices to focus on α' in the remainder of this proof.

   Let us consider $\psi : p(\alpha') = x$. Through the use of the previous parts of this proposition and the propositional rules, showing that $\Phi \Vdash^{+}_{\Pi} \psi$ iff $\Phi \Vdash^{+}_{\Pi} \varphi$ follows similarly to the proof of Proposition 4.22. Similar holds for cases where ψ is $\top$ or $\bot$. Let us therefore consider more complex epistemic formulae ψ. We first bring ψ to its (propositional) negation normal form. We then replace every negated epistemic atom with a positive one using the derivable rules 15 to 20. Based on enumeration rule E1, every epistemic atom using $\# \in \{>, <, \geq, \leq, \neq\}$ can be equivalently expressed a disjunction of atoms using only equality or $\bot$. Hence, ψ can be transformed into $\psi_{=}$ that uses only positive equality atoms or $\bot$ and s.t. $\Phi \Vdash^{+}_{\Pi} \psi$ iff $\Phi \Vdash^{+}_{\Pi} \psi_{=}$. Every term in an epistemic atom can be transformed into $\bot$ or a disjunction of certain argument complete formulae. Furthermore, every epistemic atom containing a term equivalent to $\bot$ can, depending on the value x contained in the atom, be replaced by an epistemic atom $\bot$ or $\top$ through the use of the propositional and basic rules. In a similar fashion, every epistemic atom containing a term equivalent to $\top$ (i.e. one using all possible argument complete formulae) can, depending on the value of x, be replaced by an epistemic atom $\bot$ or $\top$. The resulting epistemic formula can be transformed into a minimal (propositional) conjunctive normal form and if required, the derivable rules and the propositional identity and domination laws[13] can be used to

[13] Recall that the identity and domination logical equivalence laws state that $q \wedge \top \equiv q$, $q \vee \bot \equiv q$, $q \vee \top \equiv \top$ and $q \wedge \bot \equiv \bot$.

further simplify it. We therefore obtain a formula ψ' s.t. $\psi' : \top$, or $\psi' : \bot$, or $\psi' : \psi'_1 \wedge \ldots \wedge \psi'_a$ where $a \leq 1$ and every $\psi'_i : p(\alpha_{i_1}) = x_{i_1} \vee \ldots \vee p(\alpha_{i_n}) = x_{i,n}$ s.t. every α_{i_k} is a disjunction of certain argument complete formulae and is not equivalent to neither $\bot$ nor $\top$. From this point on, we can proceed in the same fashion as in Proposition 4.22 (we only replace the use of rule 32 by rule 35) in order to transform every atom in ψ' into DDNF and the resulting formula into a disjunction of basic epistemic formulae associated with epistemic distributions. Using the fact that our system is sound, we can show that this formula is indeed a DDNF of ψ. We can therefore conclude that for all $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$, $\Phi \Vdash^{+}_{\Pi} \psi$ iff $\Phi \Vdash^{+}_{\Pi} \varphi$, where φ is the DDNF of ψ.

□

Proposition 4.34. *Let* Π *be a restricted value set. For* $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$*, and* $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$*,* $\Phi \Vdash^{+}_{\Pi} \psi$ *iff* $\Phi \Vdash\!\!=_{\Pi} \psi$.

Proof. We have shown that our system is sound in Proposition 4.32. We now need to show that the system is also complete, i.e. that if $\Phi \Vdash\!\!=_{\Pi} \psi$ then $\Phi \Vdash^{+}_{\Pi} \psi$.

Let $\Phi = \{\phi_1, \ldots, \phi_m\}$ and $\phi : \phi_1 \wedge \ldots \wedge \phi_m$. By using the propositional rules we can easily show that $\Phi \Vdash^{+}_{\Pi} \phi$ and for every formula $\gamma \in \Phi$, $\{\phi\} \Vdash^{+}_{\Pi} \gamma$. Furthermore, it clearly follows from the definition of Sat that $\mathsf{Sat}(\Phi, \Pi) = \mathsf{Sat}(\phi, \Pi)$. Consequently, for the purpose of this proof, it suffices to show that if $\{\phi\} \Vdash\!\!=_{\Pi} \psi$ then $\{\phi\} \Vdash^{+}_{\Pi} \psi$. If $\Phi = \emptyset$, then it is easy to see that we can set ϕ to $\top$.

Let $\mathsf{Sat}(\phi, \Pi) = \{P_1, \ldots, P_k\}$ and $\mathsf{Sat}(\psi, \Pi) = \{P'_1, \ldots, P'_l\}$. Let φ^{ϕ} and φ^{ψ} be the DDNFs of ϕ and ψ. Based on Proposition 4.18, $\mathsf{Sat}(\psi, \Pi) = \mathsf{Sat}(\varphi^{\psi}, \Pi)$ and $\mathsf{Sat}(\phi, \Pi) = \mathsf{Sat}(\varphi^{\phi}, \Pi)$. Therefore, $\{\phi\} \Vdash\!\!=_{\Pi} \psi$ iff $\{\varphi^{\phi}\} \Vdash\!\!=_{\Pi} \varphi^{\psi}$. By definition, $\{\varphi^{\phi}\} \Vdash\!\!=_{\Pi} \varphi^{\psi}$ iff $\mathsf{Sat}(\varphi^{\phi}, \Pi) \subseteq \mathsf{Sat}(\varphi^{\psi}, \Pi)$. If $k = 0$, then the DDNF of ϕ is $\bot$ and therefore through the propositional rule P2, we can show that $\{\phi\} \Vdash^{+}_{\Pi} \psi$ for any ψ. If $k \neq 0$ and $l = 0$, then it cannot be the case that $\mathsf{Sat}(\varphi^{\phi}, \Pi) \subseteq \mathsf{Sat}(\varphi^{\psi}, \Pi)$. Therefore, it cannot be the case that $\{\phi\} \Vdash\!\!=_{\Pi} \psi$ either and we reach a contradiction. Hence, let $k \neq 0$ and $l \neq 0$. Let φ^{ϕ} and φ^{ψ} be of the forms $\varphi^{\phi} : \varphi^{P_1} \vee \varphi^{P_2} \ldots \vee \varphi^{P_k}$ and $\varphi^{\psi} : \varphi^{P'_1} \vee \varphi^{P'_2} \ldots \vee \varphi^{P'_l}$. Since $\mathsf{Sat}(\varphi^{\phi}, \Pi) \subseteq \mathsf{Sat}(\varphi^{\psi}, \Pi)$, then for every P_i there exists P'_j s.t. $P_i = P'_j$ and therefore for every φ^{P_i} there exists an equivalent $\varphi^{P'_j}$. Consequently, by using the propositional proof rule P1 it is easy to show that if $\{\varphi^{\phi}\} \Vdash\!\!=_{\Pi} \varphi^{\psi}$ then $\{\varphi^{\phi}\} \Vdash^{+}_{\Pi} \varphi^{\psi}$. We can now use Proposition 4.33 to show that $\{\phi\} \Vdash^{+}_{\Pi} \psi$. We can therefore conclude that if $\Phi \Vdash\!\!=_{\Pi} \psi$ then $\Phi \Vdash^{+}_{\Pi} \psi$ and our system is complete. □

Proposition 4.35. *Let* Π *be a reasonable restricted value set. For* $\Phi \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$ *and* $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$*,* $\Phi \Vdash^{+}_{\Pi} \psi$ *iff* $\Phi \cup \{\neg\psi\} \Vdash^{+}_{\Pi} \bot$.

Proof. Using Propositions 4.34 and 4.33 the definition of DDNF, we can prove this property similarly as in Proposition 4.24. □

Proposition 4.36. *Let* Π *be a reasonable restricted value set,* $\Phi \subseteq \mathsf{BFormulae}(\mathcal{G}, \Pi)$ *be a set of basic restricted formulae and* $\varphi \in \mathsf{BFormulae}(\mathcal{G}, \Pi)$ *a basic restricted formulae. If* $\Phi \Vdash_{\Pi} \varphi$ *then* $\Phi \Vdash^{+}_{\Pi} \varphi$.

Proof. Let us consider the rules of the basic system. We can observe that the basic and propositional rules are identical to the ones in the valued system. Furthermore, we can observe that for $k = 0$, $\Pi^{x,()}_{\#} = \{(v) \mid v \in \Pi^{x}_{\#}$. Hence, the valued enumeration rules applied to basic epistemic formulae coincide with the formulae produced by the basic enumeration rules. We can also show that the valued subject rules using only basic formulae coincide with the basic subject rules. In particular, rules S1 and S2 become the same in both systems, valued rules S5 and S6 reduce to basic rules S4 and S3 respectively, and the remaining properties are never used since for no two basic formulae ψ and ϕ, $\psi \succeq^{-}_{su} \phi$. The probabilistic rules from the basic system coincides with the probabilistic rule of the valued system once it is transformed using the enumeration rule E1. □

Proposition 4.37. *Let* Π *be a reasonable restricted value set, and let the restricted valued language w.r.t.* Π *be* $\mathsf{VFormulae}(\mathcal{G}, \Pi)$*. There is a function* $f : \mathsf{VFormulae}(\mathcal{G}, \Pi) \rightarrow \mathsf{BFormulae}(\mathcal{G}, \Pi)$ *s.t. for each* $\{\phi_1, \ldots, \phi_n\} \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$*, and for each* $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$*,* $\{\phi_1, \ldots, \phi_n\} \Vdash^{+}_{\Pi} \psi$ *iff* $\{f(\phi_1), \ldots, f(\phi_n)\} \Vdash_{\Pi} f(\psi)$.

Proof. Every restricted valued epistemic formula has an equivalent restricted basic formula – in particular, its DDNF. Let f assign to every formula ψ its DDNF φ^{ψ}. It is easy to see that $\{\phi_1, \ldots, \phi_n\} \Vdash^{+}_{\Pi} \psi$ iff $\{\varphi^{\phi_1 \wedge \ldots \wedge \phi_n}\} \Vdash^{+}_{\Pi} \varphi^{\psi}$, where φ^{ϕ_i} and φ^{ψ} are DDNFs of ϕ_i and ψ respectively. Furthermore, since the valued proof system is sound and complete, $\{\varphi^{\phi_1}, \ldots, \varphi^{\phi_n}\} \Vdash^{+}_{\Pi} \varphi^{\psi}$ iff $\mathsf{Sat}(\{\varphi^{\phi_1}, \ldots, \varphi^{\phi_n}\}, \Pi) \subseteq \mathsf{Sat}(\varphi^{\psi}, \Pi)$. Given the soundness and completeness of the basic proof system and the fact that every DDNF is a basic formula, $\mathsf{Sat}(\{\varphi^{\phi_1}, \ldots, \varphi^{\phi_n}\}, \Pi) \subseteq \mathsf{Sat}(\varphi^{\psi}, \Pi)$ iff $\{\varphi^{\phi_1}, \ldots, \varphi^{\phi_n}\} \Vdash_{\Pi} \varphi^{\psi}$. Therefore, $\{\phi_1, \ldots, \phi_n\} \Vdash^{+}_{\Pi} \psi$ iff $\{f(\phi_1), \ldots, f(\phi_n)\} \Vdash^{+}_{\Pi} f(\psi)$, which concludes our proof. □

Proposition 4.38. *The restricted basic language with the restricted epistemic consequence relation is equivalent to the restricted valued language with the restricted++ epistemic consequence relation.*

Proof. Follows easily from Proposition 4.36 and 4.37. □

Lemma 4.39. *Let Π be a reasonable restricted value set, and let the restricted constraint language w.r.t. Π be* $\mathsf{VFormulae}(\mathcal{G}, \Pi)$*. There is a set of propositional formulae Ω with $\Lambda \subseteq \Omega$, and there is a function f :* $\mathsf{VFormulae}(\mathcal{G}, \Pi) \to \Omega$ *s.t. for each* $\{\phi_1, \ldots, \phi_n\} \subseteq \mathsf{VFormulae}(\mathcal{G}, \Pi)$*, and for each* $\psi \in \mathsf{VFormulae}(\mathcal{G}, \Pi)$*,*

$$\{\phi_1, \ldots, \phi_n\} \Vdash^{+}_{\Pi} \psi \textit{ iff } \{f(\phi_1), \ldots, f(\phi_n)\} \cup \Lambda \vdash f(\psi)$$

Proof. Follows from Propositions 4.27 and 4.38. □

Proposition 4.40. *The restricted constraint language with the restricted++ epistemic consequence relation is equivalent to the classical propositional language with the classical propositional consequence relation.*

Proof. Follows easily from Proposition 3.13 and Lemmas 4.39 and 4.26. □

Proposition 5.5. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph and $X' = (\mathcal{G}, \mathcal{L}, \mathcal{C}')$ its single form. Then* $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$*.*

Proof. Let $\mathcal{C} = \{\phi_1, \ldots, \phi_m\}$. If $m = 0$, then $\mathcal{C}' = \mathcal{C}$ and it holds easily that $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$. Otherwise, we observe that by definition, $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\phi_1) \cap \ldots \cap \mathsf{Sat}(\phi_m)$. Furthermore, $\mathsf{Sat}(\phi_1) \cap \ldots \cap \mathsf{Sat}(\phi_m) = \mathsf{Sat}(\phi_1 \wedge \ldots \wedge \phi_m) = \mathsf{Sat}(\mathcal{C}')$. Hence, $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$. □

Proposition 5.7. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be an epistemic graph and* $\mathsf{Clausal}(X) = (\mathcal{G}, \mathcal{L}, \mathcal{C}')$ *its clausal form. Then* $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$*.*

Proof. Let $\varphi \in \mathcal{C}$ be a constraint and ψ its d-CNF. We can observe that $\mathsf{Sat}(\varphi) = \mathsf{Sat}(\psi)$. Assume ψ is of the form $\psi : \psi_1 \wedge \ldots \wedge \psi_k$. Therefore, based on the properties of Sat, $\mathsf{Sat}(\varphi) = \mathsf{Sat}(\psi) = \mathsf{Sat}(\psi_1 \wedge \ldots \wedge \psi_k) = \mathsf{Sat}(\psi_1) \cap \ldots \cap \mathsf{Sat}(\psi_k)$. With this, it is easy to show that $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$. □

Proposition 5.17. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph,* $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ *be an argument and* $F = \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ *be a set of arguments. The following hold:*

- *If* $\mathtt{A}$ *is default covered in X, then it is partially and fully covered w.r.t. any set of arguments* $G \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$*, but not necessarily vice versa*
- *If* $\mathtt{A}$ *is fully covered in X w.r.t. F, then it is partially covered in X w.r.t. F, but not necessarily vice versa*

Proof. The properties hold straightforwardly from the definitions of default, full and partial coverage. For counterexamples showing that the relations hold only one way, please consult Examples 53, 55 and 56. □

Proposition 5.18. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ be a consistent epistemic graph,* $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ *be an argument and* $F = \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ *be a set of arguments. Let Π be a reasonable restricted value set and $1 \leq k < |\Pi|$. The following hold:*

- $\mathtt{A}$ *is 1–covered w.r.t. Π in X iff it is restricted default covered in X*

- *If* $\mathtt{A}$ *is k–covered w.r.t.* Π *in* X*, then it is restricted arbitrary fully covered in* X*, but not necessarily vice versa*

- *For* $k < |\Pi| - 1$*, if* $\mathtt{A}$ *is k–covered w.r.t.* Π *in* X *then it is (k+1)–covered w.r.t.* Π *in* X

Proof.

- By definition, if $\mathtt{A}$ is 1–covered w.r.t. Π, then there exists $x \in \Pi$ s.t. $\mathcal{C} \Vdash^{+}_{\Pi} p(\mathtt{A}) \neq x$, which clearly coincides with the restricted version of default coverage. Thus, $\mathtt{A}$ is 1–covered w.r.t. Π in X iff it is restricted default covered in X

- We need to show that if there exists a set of distinct values $\{x_1, \ldots, x_k\} \subseteq \Pi$ s.t. $\mathcal{C} \Vdash^{+}_{\Pi} p(\mathtt{A}) \neq x_1 \vee \ldots \vee p(\mathtt{A}) \neq x_k$, then there exists a set of arguments $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ s.t. for every constraint combination $\mathcal{CC}^F$ on values in Π s.t. $\mathcal{CC}^F \cup \mathcal{C} \nVdash^{+}_{\Pi} \bot$, there exists a value $x \in \Pi$ s.t. $\mathcal{CC}^F \cup \mathcal{C} \Vdash^{+}_{\Pi} p(\mathtt{A}) \neq x$. Let us assume that $F = \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ and let $\mathcal{CC}^F$ be a constraint combination s.t. $\mathcal{CC}^F \cup \mathcal{C} \nVdash^{+}_{\Pi} \bot$. Therefore, $\mathsf{Sat}(\mathcal{CC}^F \cup \mathcal{C}) = \{P_1, \ldots, P_m\} \neq \emptyset$. Let $Y = \{y \mid \exists P_i \text{ s.t. } P_i(\mathtt{A}) = y\}$ be the set of probabilities assigned to $\mathtt{A}$ by the satisfying distributions. We can show that since $\mathcal{C} \Vdash^{+}_{\Pi} p(\mathtt{A}) \neq x_1 \vee \ldots \vee p(\mathtt{A}) \neq x_k$, then $\mathcal{C} \cup \mathcal{CC}^F \Vdash^{+}_{\Pi} p(\mathtt{A}) \neq x_1 \vee \ldots \vee p(\mathtt{A}) \neq x_k$. Hence, there exists a value x_j in our set s.t. $x_j \notin Y$ and we can show that for the assumed constraint combination, $\mathcal{C} \cup \mathcal{CC}^F \Vdash^{+}_{\Pi} p(\mathtt{A}) \neq x_j$. Given that we can find such an x for any constraint combination consistent with our set of constraints, it holds that k–coverage implies restricted arbitrary full coverage.

  In order to see that restricted arbitrary full coverage does not imply k–coverage, consult Example 58.

- Let $\{x_1, \ldots, x_k\} \subseteq \Pi$ be a set of distinct values and $x_{k+1} \in \Pi$ a value not belonging to it. It holds trivially that if $\mathcal{C} \Vdash^{+}_{\Pi} p(\mathtt{A}) \neq x_1 \vee \ldots \vee p(\mathtt{A}) \neq x_k$, then $\mathcal{C} \Vdash^{+}_{\Pi} p(\mathtt{A}) \neq x_1 \vee \ldots \vee p(\mathtt{A}) \neq x_k \vee p(\mathtt{A}) \neq x_{k+1}$.

□

Proposition 5.19. *Let* $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ *and* $X' = (\mathcal{G}', \mathcal{L}', \mathcal{C}')$ *be consistent epistemic graphs s.t.* $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$*. An argument* $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ *is default (partially, fully) covered in* X *(and w.r.t.* $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$*) iff it is default (partially, fully) covered in* X' *(w.r.t.* F*).*

Proof. We can observe that if $\mathsf{Sat}(\mathcal{C}) = \mathsf{Sat}(\mathcal{C}')$, then $\mathsf{Nodes}(\mathcal{G}) = \mathsf{Nodes}(\mathcal{G}')$. Furthermore, from this it also holds that $\mathcal{C} \Vdash \varphi$ iff $\mathcal{C}' \Vdash \varphi$ for a given formula φ. Hence, if $\mathcal{C} \Vdash p(\mathtt{A}) \neq x$ for an argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ and value $x \in [0,1]$, then $\mathcal{C}' \Vdash p(\mathtt{A}) \neq x$ and vice versa. Therefore, an argument $\mathtt{A}$ is default covered in X iff it is default covered in X'. Remaining forms of coverage can be shown in a similar fashion.

□

Proposition 5.20. *Let* Π *be a reasonable restricted value set. Let* $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ *and* $X' = (\mathcal{G}', \mathcal{L}', \mathcal{C}')$ *be consistent epistemic graphs s.t.* $\mathsf{Sat}(\mathcal{C}, \Pi) = \mathsf{Sat}(\mathcal{C}', \Pi)$*. An argument* $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ *is restricted default (partially, fully,k–) covered in* X *iff it is restricted default (partially, fully,k–) covered in* X'*.*

Proof. Can be proved in a similar fashion as Proposition 5.19.

□

Proposition 5.23. *Let* $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ *be a consistent epistemic graph. Let* $(\mathtt{A}, \mathtt{B})$ *be a relation in* $\mathsf{Arcs}(\mathcal{G})$*,* $Z \subseteq \mathsf{Closure}(\mathcal{C})$ *a consistent set of epistemic constraints,* $F \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{B}\}$ *and* $G = F \setminus \{\mathtt{A}\}$ *sets of arguments.*

- *If* $(\mathtt{A}, \mathtt{B})$ *is strongly effective w.r.t.* F*, then it is effective w.r.t.* F*, but not necessarily vice versa*

- *If* $(\mathtt{A}, \mathtt{B})$ *is strongly semi–effective w.r.t.* (Z, F)*, then it is semi–effective w.r.t.* (Z, F)*, but not necessarily and vice versa*

- *If* $(\mathtt{A}, \mathtt{B})$ *is effective w.r.t.* F*, then it is semi–effective w.r.t.* $(\mathcal{C}, F)$ *and vice versa*

- *If* $(\mathtt{A}, \mathtt{B})$ *is strongly effective w.r.t.* F*, then it is strongly semi–effective w.r.t.* $(\mathcal{C}, F)$ *and vice versa*

- *If* $Z \neq \mathcal{C}$ *and* $(\mathtt{A}, \mathtt{B})$ *is semi–effective w.r.t.* (Z, F)*, then it is not necessarily effective w.r.t.* F

- *If $Z \neq \mathcal{C}$ and $(\mathtt{A}, \mathtt{B})$ is strongly semi–effective w.r.t. (Z, F), then it is not necessarily strongly effective w.r.t. F*

Proof.

- Follows straightforwardly from the definition. To see that the relation is one way, consult Example 59.
- Follows straightforwardly form the definition. To see that the relation is one way, consult Example 61.
- Follows straightforwardly from the definition.
- Follows straightforwardly form the definition.
- Consult Example 61.
- Consider the set of constraints $\mathcal{C} = \{p(\mathtt{A}) > 0.5, p(\mathtt{A}) > 0.5 \to p(\mathtt{B}) < 0.5\}$ and assume that the $(\mathtt{A}, \mathtt{B})$ relation is present in the graph. It is easy to see that it is neither effective nor strongly effective w.r.t. $F = \{\mathtt{A}\}$. However, given $Z = \{p(\mathtt{A}) > 0.5 \to p(\mathtt{B}) < 0.5\}$, we can show that $(\mathtt{A}, \mathtt{B})$ is strongly semi–effective w.r.t. (Z, F).

$\square$

Proposition 5.27. *Let $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$ and $X' = (\mathcal{G}, \mathcal{L}, \mathcal{C}')$ be inconsistent epistemic graphs. An argument $\mathtt{A} \in \mathsf{Nodes}(\mathcal{G})$ that is default (partially, fully) covered in X is not necessarily default (arbitrary partially, arbitrary fully) covered in X' and vice versa.*

Proof. We can compare the epistemic graph from Example 62 with an epistemic graph s.t. the argument graph is the same, but the set of constraints is $\mathcal{C}' = \{p(\mathtt{A}) = 0 \wedge \neg p(\mathtt{A}) = 0\}$. In the original graph, we will find at least one set of maximal consistent constraints containing φ_7, and thus have that, for example, argument B is default (and therefore, also fully and partially) covered. This will never be the case w.r.t. the set of constraints $\mathcal{C}'$. $\square$

Theorem 6.4. *Let $\mathcal{C}$ be a set of valued epistemic atoms using only $\leq, \geq, =$ as comparison operators. Then $\mathsf{Sat}(\mathcal{C})$ is closed and convex.*

Proof. Let $\varphi = p(\alpha_1) *_1 \ldots *_n p(\alpha_n) \leq x$ be a valued epistemic atom with $*_1, \ldots, *_n \in \{+, -\}$ (the cases $\geq$ and $=$ are analogous). Let $P_1, P_2 \in \mathsf{Sat}(\{\varphi\})$, $\delta \in [0, 1]$. We need to show that $P^{\delta}_{P_1,P_2} \in \mathsf{Sat}(\{\varphi\})$ as well. First, note that for every $\alpha \in \mathsf{Terms}(\mathcal{G})$ we have $P^{\delta}_{P_1,P_2}(\alpha) = \delta P_1(\alpha) + (1-\delta)P_2(\alpha)$. Then

$$
\begin{aligned}
& P^{\delta}_{P_1,P_2}(\alpha_1) *_1 \ldots *_n P^{\delta}_{P_1,P_2}(\alpha_n) \\
= & \delta P_1(\alpha_1) + (1-\delta)P_2(\alpha_1) *_1 \ldots *_n \delta P_1(\alpha_n) + (1-\delta)P_2(\alpha_n) \\
= & \delta(P_1(\alpha_1) *_1 \ldots *_n P_1(\alpha_n)) + (1-\delta)(P_2(\alpha_1) *_1 \ldots *_n P_2(\alpha_n)) \\
\leq & \delta x + (1-\delta)x = x
\end{aligned}
$$

showing $P^{\delta}_{P_1,P_2} \in \mathsf{Sat}(\{\varphi\})$. This generalises directly to multiple constraints as the intersection of convex sets is also convex. $\square$

Theorem 6.5. *PCSAT is NP-complete.*

Proof. For NP-membership, consider the following non-deterministic algorithm:

1. Let $\alpha_1, \ldots, \alpha_n$ be the valued epistemic atoms appearing in the valued epistemic formulae in $\mathcal{C}$.
2. Guess a subset $\{\beta_1, \ldots, \beta_m\} \subseteq \{\alpha_1, \ldots, \alpha_n\}$ of valued epistemic atoms (these atoms are the ones supposed to be true); w.l.o.g. say $\{\beta_1, \ldots, \beta_m\} = \{\alpha_1, \ldots, \alpha_m\}$.
3. Guess a polynomial number of subsets $\Gamma_1, \ldots, \Gamma_l \subseteq \mathsf{Nodes}(\mathcal{G})$ and set $P(\Gamma') = 0$ for all other subsets $\Gamma' \subseteq \mathsf{Nodes}(\mathcal{G})$.

4. Solve the set of constraints $\{\alpha_1, \ldots, \alpha_m\}$ using linear programming techniques and obtain a solution for $P(\Gamma_1), \ldots, P(\Gamma_l)$.
5. Verify that $\mathcal{C}$ is true if $\alpha_1, \ldots, \alpha_m$ are true.

First, observe that this algorithm runs in polynomial time, in particular, steps 3 and 4 are each polynomial (linear programming is in P). In steps 2-4 we solve a classical probabilistic satisfiability problem, i. e. we check whether there is a belief distribution that satisfies the atoms $\alpha_1, \ldots, \alpha_m$. As these form a linear program and due to the small-model-property of linear programs [27], there is a solution where only a polynomial number of variables receive a non-zero value. These variables (here: subsets of arguments) are guessed and verified afterwards. Finally, it has to be verified that satisfying this subset of atoms is sufficient to satisfy all valued epistemic formulae (which are built over these atoms).

NP-hardness follows from the fact that probabilistic satisfiability is a special case. □

Theorem 6.6. *resPCSAT is NP-complete.*

Proof. The proof is analogous to the proof of Theorem 6.5 except that in step 3 we also guess values $x_1, \ldots, x_l \in \Pi$ and define $P(\Gamma_i) = x_i$ for all $i = 1, \ldots, l$. In Step 4 we only have to check whether the obtained system of equations is valid (it contains no variables any more). Moreover, note that resPCSAT is still a generalisation of probabilistic satisfiability. □

Theorem 6.8. *PCSAT-NVAL is coNP-complete.*

Proof. Checking whether $\mathtt{A}$ can only be assigned a single value x by the satisfying distributions can be done by verifying that the $\mathcal{C} \cup \{p(\mathtt{A}) \neq x\}$ is not satisfiable, which puts our problem in coNP. Hardness can be easily shown from the PCSAT-VAL problem. Thus, we cano show PCSAT-NVAL to be coNP-complete. □

Theorem 9.1. *EXIST-SAT, EXIST-SAT-AMAX, EXIST-SAT-AMIN, EXIST-SAT-RMAX, EXIST-SAT-RMIN, EXIST-SAT-UMAX, EXIST-SAT-UMIN, EXIST-SAT-IMAX and EXIST-SAT-IMIN are NP-complete.*

Proof. It is easy to see that EXIST-SAT is equivalent to PCSAT. For a given graph $X = (\mathcal{G}, \mathcal{L}, \mathcal{C})$, EXIST-SAT returns TRUE if and only if PCSAT returns TRUE for $\mathcal{C}$. We can therefore easily show the problem to be NP-complete.

Let us now consider the EXIST-SAT-AMAX problem. We can observe that asking whether there exists an acceptance maximizing satisfying probability distribution is not different from asking whether there exists a satisfying one. For a given probability distribution P, let $Bel(P) = \{\mathtt{A} \mid P(\mathtt{A}) > 0.5\}$ be the set of believed arguments. Since we have finitely many arguments, the collection of all possible $Bel(P)$ sets is finite and as long as it is not empty, a maximal w.r.t. $\subseteq$ set can always be found. Thus, if we have a satisfying distribution, then we have an acceptance maximizing satisfying one, and if there is an acceptance maximizing satisfying one, there has to be a satisfying one in the first place. Thus, EXIST-SAT-AMAX returns TRUE if and only if EXIST-SAT does, and we can show the problem to be NP-complete. Similar analysis can be carried out for the EXIST-SAT-AMIN, EXIST-SAT-RMAX, EXIST-SAT-RMIN, EXIST-SAT-UMAX, EXIST-SAT-UMIN, EXIST-SAT-IMAX and EXIST-SAT-IMIN problems. □

Theorem 9.2. *UVER-SAT is in P. UVER-SAT-AMAX, UVER-SAT-RMAX, UVER-SAT-UMIN and UVER-SAT-IMAX are coNP-complete. UVER-SAT-AMIN, UVER-SAT-RMIN, UVER-SAT-UMAX and UVER-SAT-IMIN are in coNP.*

Proof. Verifying that the probabilistic distributions described by $ParP$ are satisfying distributions of an uni–argument epistemic graph can be achieved in polynomial time. Thus, UVER-SAT is in P.

Let us now consider UVER-SAT-AMAX. Verifying that the probabilistic distributions described by $ParP$ are satisfying distributions of an uni–argument epistemic graph can be achieved in polynomial time. Let $Y = \{y_1, \ldots, y_n\} = \{\mathtt{A} \mid ParP(\mathtt{A}) > 0.5\}$ be the set of believed arguments and $Z = \{z_1, \ldots, z_m\} = \mathsf{Nodes}(\mathcal{G}) \setminus Y$ the remaining arguments. We create the constraint $\varphi : p(y_1) > 0.5 \wedge \ldots \wedge p(y_n) > 0.5 \wedge$

$(p(z_1) > 0.5 \vee \ldots \vee p(z_m) > 0.5)$. We can observe that $ParP$ does not describe acceptance maximizing distributions iff $\mathsf{Sat}(\mathcal{C} \cup \{\varphi\}) \neq \emptyset$. Thus, UVER-SAT-AMAX is in coNP.

For hardness, let us consider the problem of verification for preferred semantics in Dung's abstract argumentation frameworks. Let $(\mathcal{G}', \mathcal{L}')$ be a Dung's graph. For every argument $\mathtt{C} \in \mathsf{Nodes}(\mathcal{G})$ with attackers $\mathtt{B}_1, \ldots, \mathtt{B}_n$ we create the constraints

$$(p(\mathtt{B}_1) < 0.5 \wedge \ldots \wedge p(\mathtt{B}_n) < 0.5) \leftrightarrow p(\mathtt{C}) > 0.5)$$
$$(p(\mathtt{B}_1) > 0.5 \vee \ldots \vee p(\mathtt{B}_n) > 0.5) \leftrightarrow p(\mathtt{C}) < 0.5)$$

In case a given argument C is unattacked, we create the constraint $p(\mathtt{C}) > 0.5$. Let $\mathcal{C}'$ be the set of all of the aforementioned constraints. Based on the results [22, 71], the ternary satisfying distributions of $\mathcal{C}'$ describe the complete labelings (and extensions) of our Dung's graph, and the ternary satisfying acceptance maximal distributions of $(\mathcal{G}', \mathcal{L}', \mathcal{C}')$ correspond to the preferred labelings of this graph. Let $Ext \subseteq \mathsf{Nodes}(\mathcal{G}')$ and $Ext^+ = \{\mathtt{C} \mid \exists \mathtt{D} \in Ext \text{ s.t. } (\mathtt{C}, \mathtt{D}) \in \mathsf{Arcs}(\mathcal{G}')\}$. Given the relation between preferred labelings and extensions, it holds that Ext is a preferred extensions of $(\mathcal{G}', \mathcal{L}')$ if and only if a probability distribution P s.t. $P(\mathtt{A}) = 1$ for $\mathtt{A} \in Ext$, $P(\mathtt{B}) = 0.5$ for $\mathtt{B} \in \mathsf{Nodes}(\mathcal{G}) \setminus (Ext \cup Ext^+)$ and $P(\mathtt{C}) = 0$ for $\mathtt{C} \in Ext^+$, is a ternary satisfying acceptance maximal distribution of $(\mathcal{G}', \mathcal{L}', \mathcal{C}')$. We note that the above restrictions in fact form an argument–based partial description of a distribution. Verifying that Ext is a preferred extension is coNP-complete [35] and the aforementioned transformations and checks can be done in polynomial time. This, in combination with the previously shown coNP membership of UVER-SAT-AMAX, proves that UVER-SAT-AMAX is coNP-complete.

Let us now consider UVER-SAT-RMAX. Verifying that the probabilistic distributions described by $ParP$ are satisfying distributions of an uni–argument epistemic graph can be achieved in polynomial time. Let $Y = \{y_1, \ldots, y_n\} = \{\mathtt{A} \mid P(\mathtt{A}) < 0.5\}$ be the set of disbelieved arguments and $Z = \{z_1, \ldots, z_m\} = \mathsf{Nodes}(\mathcal{G}) \setminus Y$ the remaining arguments. We create the constraint $\varphi : p(y_1) < 0.5 \wedge \ldots \wedge p(y_n) < 0.5 \wedge (p(z_1) < 0.5 \vee \ldots \vee p(z_m) < 0.5)$. We can observe that $ParP$ does not describe rejection maximizing distributions iff $\mathsf{Sat}(\mathcal{C} \cup \{\varphi\}) \neq \emptyset$. Thus, UVER-SAT-RMAX is in coNP. Hardness can be showed in a similar fashion as for UVER-SAT-AMAX given the results in [22, 71] (i.e. we use the fact that preferred labelings are out maximizing complete labelings). Thus, UVER-SAT-RMAX is coNP-complete.

Let us now consider UVER-SAT-IMAX. Verifying that the probabilistic distributions described by $ParP$ are satisfying distributions of an uni–argument epistemic graph can be achieved in polynomial time. Let $Y = \{y_1, \ldots, y_n\} = \{\mathtt{A} \mid P(\mathtt{A}) < 0.5\}$ be the set of disbelieved arguments, $W = \{w_1, \ldots, w_k\} = \{\mathtt{A} \mid P(\mathtt{A}) > 0.5\}$ the set of believed arguments, and $Z = \{z_1, \ldots, z_m\} = \mathsf{Nodes}(\mathcal{G}) \setminus (Y \cup W)$ the remaining arguments. We create the constraint $\varphi : p(y_1) < 0.5 \wedge \ldots \wedge p(y_n) < 0.5 \wedge p(w_1) > 0.5 \wedge \ldots \wedge p(w_k) > 0.5(p(z_1) \neq 0.5 \vee \ldots \vee p(z_m) \neq 0.5)$. We can observe that $ParP$ does not describe information maximizing distributions iff $\mathsf{Sat}(\mathcal{C} \cup \{\varphi\}) \neq \emptyset$. Thus, UVER-SAT-IMAX is in coNP. Hardness can be showed in a similar fashion as for UVER-SAT-AMAX given the results in [22, 71] (i.e. we use the fact that preferred labelings are in and out maximizing complete labelings). Thus, UVER-SAT-IMAX is coNP-complete.

Let us now consider UVER-SAT-UMIN. Verifying that the probabilistic distributions described by $ParP$ are satisfying distributions of an uni–argument epistemic graph can be achieved in polynomial time. Let $Y = \{y_1, \ldots, y_n\} = \{\mathtt{A} \mid P(\mathtt{A}) = 0.5\}$ be the set of undecided arguments and $Z = \{z_1, \ldots, z_m\} = \mathsf{Nodes}(\mathcal{G}) \setminus Y$ the remaining arguments. We create the constraint $\varphi : p(z_1) \neq 0.5 \wedge \ldots \wedge p(z_m) \neq 0.5 \wedge (p(y_1) \neq 0.5 \vee \ldots \vee p(y_n) \neq 0.5)$. We can observe that $ParP$ does not describe undecided minimizing distributions iff $\mathsf{Sat}(\mathcal{C} \cup \{\varphi\}) \neq \emptyset$. Thus, UVER-SAT-UMIN is in coNP. Hardness can be showed in a similar fashion as for UVER-SAT-AMAX given the results in [34, 22, 71] (i.e. we use the fact that semi–stable labelings are und minimizing complete labelings). Thus, UVER-SAT-UMIN is coNP-complete.

Let us now consider UVER-SAT-AMIN. Verifying that the probabilistic distributions described by $ParP$ are satisfying distributions of an uni–argument epistemic graph can be achieved in polynomial time. Let $Y = \{y_1, \ldots, y_n\} = \{\mathtt{A} \mid P(\mathtt{A}) > 0.5\}$ be the set of believed arguments and $Z = \{z_1, \ldots, z_m\} = \mathsf{Nodes}(\mathcal{G}) \setminus Y$ the remaining arguments. We create the constraint $\varphi : p(z_1) \leq 0.5 \wedge \ldots \wedge p(z_m) \leq 0.5 \wedge (p(y_1) \leq 0.5 \vee \ldots \vee p(y_n) \leq 0.5)$. We can observe that $ParP$ does not describe acceptance minimizing distributions iff $\mathsf{Sat}(\mathcal{C} \cup \{\varphi\}) \neq \emptyset$. Thus, UVER-SAT-AMIN is in coNP.

Let us now consider UVER-SAT-RMIN. Verifying that the probabilistic distributions described by $ParP$ are satisfying distributions of an uni–argument epistemic graph can be achieved in polynomial time.

Let $Y = \{y_1, \dots, y_n\} = \{\mathtt{A} \mid P(\mathtt{A}) < 0.5\}$ be the set of disbelieved arguments and $Z = \{z_1, \dots, z_m\} = \mathsf{Nodes}(\mathcal{G}) \setminus Y$ the remaining arguments. We create the constraint $\varphi : p(z_1) \geq 0.5 \wedge \dots \wedge p(z_m) \geq 0.5 \wedge (p(y_1) \geq 0.5 \vee \dots \vee p(y_n) \geq 0.5)$. We can observe that $ParP$ does not describe rejection minimizing distributions iff $\mathsf{Sat}(\mathcal{C} \cup \{\varphi\}) \neq \emptyset$. Thus, UVER-SAT-RMIN is in coNP.

Let us now consider UVER-SAT-IMIN. Verifying that the probabilistic distributions described by $ParP$ are satisfying distributions of an uni–argument epistemic graph can be achieved in polynomial time. Let $Y = \{y_1, \dots, y_n\} = \{\mathtt{A} \mid P(\mathtt{A}) < 0.5\}$ be the set of disbelieved arguments, $W = \{w_1, \dots, w_k\} = \{\mathtt{A} \mid P(\mathtt{A}) > 0.5\}$ the set of believed arguments, and $Z = \{z_1, \dots, z_m\} = \mathsf{Nodes}(\mathcal{G}) \setminus (Y \cup W)$ the remaining arguments. We create the constraint $\varphi : p(z_1) = 0.5 \wedge \dots \wedge p(z_m) = 0.5 \wedge p(w_1) \geq 0.5 \wedge \dots \wedge p(w_k) \geq 0.5 \wedge p(y_1) \leq 0.5 \wedge \dots \wedge p(y_n) =\leq 0.5 \wedge (p(y_1) = 0.5 \vee \dots \vee p(y_n) = 0.5 \vee p(w_1) = 0.5 \vee \dots \vee p(w_k) = 0.5)$. We can observe that $ParP$ does not describe information minimizing distributions iff $\mathsf{Sat}(\mathcal{C} \cup \{\varphi\}) \neq \emptyset$. Thus, UVER-SAT-IMIN is in coNP.

Let us now consider UVER-SAT-UMAX. Verifying that the probabilistic distributions described by $ParP$ are satisfying distributions of an uni–argument epistemic graph can be achieved in polynomial time. Let $Y = \{y_1, \dots, y_n\} = \{\mathtt{A} \mid P(\mathtt{A}) = 0.5\}$ be the set of undecided arguments and $Z = \{z_1, \dots, z_m\} = \mathsf{Nodes}(\mathcal{G}) \setminus Y$ the remaining arguments. We create the constraint $\varphi : p(y_1) = 0.5 \wedge \dots \wedge p(y_n) = 0.5 \wedge (p(z_1) = 0.5 \vee \dots \vee p(z_m) = 0.5)$. We can observe that $ParP$ does not describe undecided maximizing distributions iff $\mathsf{Sat}(\mathcal{C} \cup \{\varphi\}) \neq \emptyset$. Thus, UVER-SAT-UMAX is in coNP. □

Theorem 9.3. *CRED-SAT, CRED-SAT-AMAX and CRED-SAT-IMAX are NP-complete. CRED-SAT-AMIN, CRED-SAT-RMAX, CRED-SAT-RMIN, CRED-SAT-UMAX CRED-SAT-IMIN are in Σ_2^p. CRED-SAT-UMIN is Σ_2^p-complete. SKEPT-SAT is coNP-complete. SKEPT-SAT-AMIN, SKEPT-SAT-RMIN, SKEPT-SAT-UMAX and SKEPT-SAT-IMIN are in Π_2^p. Finally, SKEPT-SAT-AMAX, SKEPT-SAT-RMAX, SKEPT-SAT-UMIN and SKEPT-SAT-IMAX are Π_2^p-complete.*

Proof. Let us first consider CRED-SAT. We can observe that the answer is yes if and only if $\mathcal{C} \cup \{p(\mathtt{A}) > 0.5\}$ is satisfiable, thus we can show the problem to be in NP. For hardness, let us consider the problem of credulous acceptance for admissible semantics in Dung's abstract argumentation frameworks. Let $(\mathcal{G}', \mathcal{L}')$ be a Dung's graph. For every argument $\mathtt{C} \in \mathsf{Nodes}(\mathcal{G})$ with attackers $\mathtt{B}_1, \dots, \mathtt{B}_n$ we create the constraints

$$p(\mathtt{C}) > 0.5 \rightarrow (p(\mathtt{B}_1) < 0.5 \wedge \dots \wedge p(\mathtt{B}_n) < 0.5)$$
$$p(\mathtt{C}) < 0.5 \rightarrow (p(\mathtt{B}_1) > 0.5 \vee \dots \vee p(\mathtt{B}_n) > 0.5)$$

In case a given argument $\mathtt{C}$ is unattacked, we create the constraint $p(\mathtt{C}) \geq 0.5$. Let $\mathcal{C}'$ be the set of all of the aforementioned constraints. Based on the results [71], the ternary satisfying distributions of $\mathcal{C}'$ describe the admissible labelings (and therefore extensions) of our Dung's graph. Let $Ext \subseteq \mathsf{Nodes}(\mathcal{G}')$ and $Ext^+ = \{\mathtt{C} \mid \exists \mathtt{D} \in Ext \text{ s.t. } (\mathtt{C}, \mathtt{D}) \in \mathsf{Arcs}(\mathcal{G}')\}$. Given the relation between preferred labelings and extensions, it holds that Ext is an admissible extension of $(\mathcal{G}', \mathcal{L}')$ if and only if a probability distribution P s.t. $P(\mathtt{A}) = 1$ for $\mathtt{A} \in Ext$, $P(\mathtt{B}) = 0.5$ for $\mathtt{B} \in \mathsf{Nodes}(\mathcal{G}) \setminus (Ext \cup Ext^+)$ and $P(\mathtt{C}) = 0$ for $\mathtt{C} \in Ext^+$, is a ternary satisfying distribution of $(\mathcal{G}', \mathcal{L}', \mathcal{C}')$. Thus, an argument is credulously accepted in an admissible extension of $(\mathcal{G}', \mathcal{L}')$ iff it is assigned a probability 1 by a ternary satisfying distribution of $(\mathcal{G}', \mathcal{L}', \mathcal{C}')$. Credulous acceptance under the admissible semantics is NP-complete [35] and the aforementioned transformations and checks can be done in polynomial time. This, in combination with the previously shown NP-membership of CRED-SAT, proves that CRED-SAT is NP-complete.

Similarly like the problem of credulous acceptance for admissible semantics is the same as the problem for credulous acceptance for preferred semantics in Dung's graph, CRED-SAT-AMAX is equivalent to CRED-SAT: if $P(\mathtt{A}) > 0.5$ for some $P \in \mathsf{Sat}(\mathcal{C})$ then there is a acceptance maximizing $P' \in \mathsf{Sat}(\mathcal{C})$ with $\{\mathtt{A} \mid P(\mathtt{A}) > 0.5\} \subseteq \{\mathtt{A} \mid P'(\mathtt{A}) > 0.5\}$ and therefore $P'(\mathtt{A}) > 0.5$. We can therefore show that CRED-SAT-AMAX is NP-complete. The same can be argued for CRED-SAT-IMAX.

Let us now consider CRED-SAT-AMIN. The Σ_2^p-membership of this problem can be seen from the following algorithm:

1. Guess two disjoint sets of arguments $\{\mathtt{A}_1, \dots, \mathtt{A}_n\} \subseteq \mathsf{Nodes}(\mathcal{G})$ s.t. for some i, $\mathtt{A}_i = \mathtt{A}$, and $\{\mathtt{B}_1, \dots, \mathtt{B}_m\}$, and let $\{\mathtt{C}_1, \dots, \mathtt{C}_k\}$ be the remaining arguments.

2. Create the constraint $\varphi : p(\mathtt{A}_1) > 0.5 \wedge \ldots \wedge p(\mathtt{A}_n) > 0.5 \wedge p(\mathtt{B}_1) < 0.5 \wedge \ldots \wedge p(\mathtt{B}_m) < 0.5 \wedge p(\mathtt{C}_1) = 0.5 \wedge \ldots \wedge p(\mathtt{C}_k) = 0.5$

3. Create the constraint $\psi : (p(\mathtt{A}_1) \leq 0.5 \vee \ldots \vee p(\mathtt{A}_n) \leq 0.5) \wedge p(\mathtt{B}_1) \leq 0.5 \wedge \ldots \wedge p(\mathtt{B}_m) \leq 0.5 \wedge p(\mathtt{C}_1) \leq 0.5 \wedge \ldots \wedge p(\mathtt{C}_k) \leq 0.5$

4. Using the NP-oracle verify that $\mathcal{C} \cup \{\varphi\}$ is satisfiable.

5. With a second call to the NP-oracle verify that $\mathcal{C} \cup \{\psi\}$ is *not* satisfiable.

We can observe that steps 4 and 5 establish that there is an acceptance minimizing probability distribution that exactly accepts $\mathtt{A}_1, \ldots, \mathtt{A}_n$, and based on the construction, accepts $\mathtt{A}$. The algorithm runs in Σ_2^p and thus CRED-SAT-AMIN is in Σ_2^p.

Let us now consider CRED-SAT-RMAX. The Σ_2^p-membership of this problem can be seen from the following algorithm:

1. Guess two disjoint sets of arguments $\{\mathtt{A}_1, \ldots, \mathtt{A}_n\} \subseteq \mathsf{Nodes}(\mathcal{G})$ s.t. for some i, $\mathtt{A}_i = \mathtt{A}$, and $\{\mathtt{B}_1, \ldots, \mathtt{B}_m\}$, and let $\{\mathtt{C}_1, \ldots, \mathtt{C}_k\}$ be the remaining arguments.

2. Create the constraint $\varphi : p(\mathtt{A}_1) > 0.5 \wedge \ldots \wedge p(\mathtt{A}_n) > 0.5 \wedge p(\mathtt{B}_1) < 0.5 \wedge \ldots \wedge p(\mathtt{B}_m) < 0.5 \wedge p(\mathtt{C}_1) = 0.5 \wedge \ldots \wedge p(\mathtt{C}_k) = 0.5$

3. Create the constraint $\psi : (p(\mathtt{A}_1) < 0.5 \vee \ldots \vee p(\mathtt{A}_n) < 0.5 \vee p(\mathtt{C}_1) < 0.5 \vee \ldots \vee p(\mathtt{C}_k) < 0.5) \wedge p(\mathtt{B}_1) < 0.5 \wedge \ldots \wedge p(\mathtt{B}_m) < 0.5$

4. Using the NP-oracle verify that $\mathcal{C} \cup \{\varphi\}$ is satisfiable.

5. With a second call to the NP-oracle verify that $\mathcal{C} \cup \{\psi\}$ is *not* satisfiable.

We can observe that steps 4 and 5 establish that there is a rejection maximizing probability distribution that exactly accepts $\mathtt{A}_1, \ldots, \mathtt{A}_n$, and based on the construction, accepts $\mathtt{A}$. The algorithm runs in Σ_2^p and thus CRED-SAT-RMAX is in Σ_2^p.

Creating membership algorithms for the remaining credulous problems follows the same pattern as in the case of CRED-SAT-AMIN and CRED-SAT-RMAX. It is only the nature of the ψ constraint that changes, but every time it is a straightforward adaptation of the constraint in the membership proof of the relevant verification problem (see proof of Theorem 9.2). Consequently, we can show all CRED-SAT-RMIN, CRED-SAT-UMAX, CRED-SAT-UMIN and CRED-SAT-IMIN to be in Σ_2^p. We will therefore focus on hardness.

Hardness for CRED-SAT-UMIN can be shown through credulous acceptance under semi–stable semantics in Dung's graph, which is Σ_2^p-complete [36, 22] (see also previous parts of this proof and the proof of Theorem 9.2) Thus, CRED-SAT-UMIN is Σ_2^p-complete.

SKEPT-SAT is equivalent to asking whether $\mathcal{C} \cup \{p(\mathtt{A}) \leq 0.5\}$ is *not* satisfiable and is therefore in coNP. To show hardness, we consider the problem of skeptical acceptance under the stable semantics for Dung's graphs. Let $(\mathcal{G}', \mathcal{L}')$ be a Dung's graph. For every argument $\mathtt{C} \in \mathsf{Nodes}(\mathcal{G})$ with attackers $\mathtt{B}_1, \ldots, \mathtt{B}_n$ we create the constraints

$$(p(\mathtt{B}_1) < 0.5 \wedge \ldots \wedge p(\mathtt{B}_n) < 0.5) \leftrightarrow p(\mathtt{C}) > 0.5)$$
$$(p(\mathtt{B}_1) > 0.5 \vee \ldots \vee p(\mathtt{B}_n) > 0.5) \leftrightarrow p(\mathtt{C}) < 0.5)$$

In case a given argument $\mathtt{C}$ is unattacked, we create the constraint $p(\mathtt{C}) > 0.5$. Let $\mathcal{C}'$ be the set of all of the aforementioned constraints. Based on the results [22, 71], we can show that the non–neutral ternary satisfying distributions of $\mathcal{C}'$ describe the stable labelings (and extensions) of our Dung's graph. We can therefore proceed as in the previous part of this proof, and since skeptical acceptance under the stable semantics is coNP-complete [35], we can show that SKEPT-SAT is coNP-complete.

Let us now consider SKEPT-SAT-AMAX. The Π_2^p-membership of this problem can be seen from the following algorithm, which is a Σ_2^p-algorithm solving the complement problem of deciding that $\mathtt{A}$ is *not* accepted in all acceptance maximizing $P \in \mathsf{Sat}(\mathcal{C})$.

1. Guess two disjoint sets of arguments $\{\mathtt{A}_1, \ldots, \mathtt{A}_n\} \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ and $\{\mathtt{B}_1, \ldots, \mathtt{B}_m\}$ and let $\{\mathtt{C}_1, \ldots, \mathtt{C}_k\}$ be the remaining arguments.
2. Create the constraint $\varphi : p(\mathtt{A}_1) > 0.5 \wedge \ldots \wedge p(\mathtt{A}_n) > 0.5 \wedge p(\mathtt{B}_1) < 0.5 \wedge \ldots \wedge p(\mathtt{B}_m) < 0.5 \wedge p(\mathtt{C}_1) = 0.5 \wedge \ldots \wedge p(\mathtt{C}_k) = 0.5$
3. Create the constraint $\psi : p(\mathtt{A}_1) > 0.5 \wedge \ldots \wedge p(\mathtt{A}_n) > 0.5 \wedge (p(\mathtt{B}_1) > 0.5 \vee \ldots \vee p(\mathtt{B}_m) > 0.5 \vee p(\mathtt{C}_1) > 0.5 \vee \ldots \vee p(\mathtt{C}_k) > 0.5)$
4. Using the NP-oracle verify that $\mathcal{C} \cup \{\varphi\}$ is satisfiable.
5. With a second call to the NP-oracle verify that $\mathcal{C} \cup \{\psi\}$ is *not* satisfiable.

We can observe that steps 4 and 5 establish that there is an acceptance maximizing probability distribution that exactly accepts $\mathtt{A}_1, \ldots, \mathtt{A}_n$, and based on the construction, does not accept $\mathtt{A}$. The algorithm runs in Σ_2^p, and as it is a co–problem, we can show SKEPT-SAT-AMAX to be in Π_2^p.

For hardness, consider the problem of skeptical acceptance wrt. preferred semantics in Dung's graphs, which has been shown to be Π_2^p-complete [33]. We can construct an epistemic graph $(\mathcal{G}', \mathcal{L}', \mathcal{C}')$ associated with a Dung's graph $(\mathcal{G}', \mathcal{L}')$ s.t. the preferred extensions of that graph correspond to the ternary acceptance maximizing satisfying distribution of the epistemic graph (see also proof of Theorem 9.2). For every argument $\mathtt{C} \in \mathsf{Nodes}(\mathcal{G}')$ with attackers $\mathtt{B}_1, \ldots, \mathtt{B}_n$ we add the constraints

$$p(\mathtt{C}) > 0.5 \leftrightarrow (p(\mathtt{B}_1) < 0.5 \wedge \ldots \wedge p(\mathtt{B}_n) < 0.5)$$
$$p(\mathtt{C}) < 0.5 \leftrightarrow (p(\mathtt{B}_1) > 0.5 \vee \ldots \vee p(\mathtt{B}_n) > 0.5)$$

to $\mathcal{C}'$ (if $\mathtt{C}$ has no attackers, we only add the constraint $p(\mathtt{C}) > 0.5$). Clearly, $\mathtt{A}$ is contained in all preferred extensions of $\mathcal{G}$ if and only if $P(\mathtt{A}) > 0.5$ for all ternary acceptance maximizing $P \in \mathsf{Sat}(\mathcal{C})$, showing Π_2^p-hardness of SKEPT-SAT-AMAX. Hence, our problem is Π_2^p-complete.

Let us now consider SKEPT-SAT-AMIN. The Π_2^p-membership of this problem can be seen from the following algorithm, which is a Σ_2^p-algorithm solving the complement problem of deciding that $\mathtt{A}$ is *not* accepted in all acceptance minimizing $P \in \mathsf{Sat}(\mathcal{C})$.

1. Guess two disjoint sets of arguments $\{\mathtt{A}_1, \ldots, \mathtt{A}_n\} \subseteq \mathsf{Nodes}(\mathcal{G}) \setminus \{\mathtt{A}\}$ and $\{\mathtt{B}_1, \ldots, \mathtt{B}_m\}$ and let $\{\mathtt{C}_1, \ldots, \mathtt{C}_k\}$ be the remaining arguments.
2. Create the constraint $\varphi : p(\mathtt{A}_1) > 0.5 \wedge \ldots \wedge p(\mathtt{A}_n) > 0.5 \wedge p(\mathtt{B}_1) < 0.5 \wedge \ldots \wedge p(\mathtt{B}_m) < 0.5 \wedge p(\mathtt{C}_1) = 0.5 \wedge \ldots \wedge p(\mathtt{C}_k) = 0.5$
3. Create the constraint $\psi : (p(\mathtt{A}_1) \leq 0.5 \vee \ldots \vee p(\mathtt{A}_n) \leq 0.5) \wedge p(\mathtt{B}_1) \leq 0.5 \wedge \ldots \wedge p(\mathtt{B}_m) \leq 0.5 \wedge p(\mathtt{C}_1) \leq 0.5 \wedge \ldots \wedge p(\mathtt{C}_k) \leq 0.5$
4. Using the NP-oracle verify that $\mathcal{C} \cup \{\varphi\}$ is satisfiable.
5. With a second call to the NP-oracle verify that $\mathcal{C} \cup \{\psi\}$ is *not* satisfiable.

We can observe that steps 4 and 5 establish that there is an acceptance minimizing probability distribution that exactly accepts $\mathtt{A}_1, \ldots, \mathtt{A}_n$, and based on the construction, does not accept $\mathtt{A}$. The algorithm runs in Σ_2^p, and as it is a co–problem, we can show SKEPT-SAT-AMIN to be in Π_2^p.

Creating membership algorithms for the remaining skeptical problems follows the same pattern as in the case of SKEPT-SAT-AMAX and SKEPT-SAT-AMIN. It is only the nature of the ψ constraint that changes, but every time it is a straightforward adaptation of the constraint in the membership proof of the relevant verification problem (see proof of Theorem 9.2). Consequently, we can show all SKEPT-SAT-RMAX, SKEPT-SAT-RMIN, SKEPT-SAT-UMAX, SKEPT-SAT-UMIN, SKEPT-SAT-IMAX and SKEPT-SAT-IMIN to be in Π_2^p. We will therefore focus on hardness.

Concerning hardness of SKEPT-SAT-RMAX and SKEPT-SAT-IMAX, we can consider the problem of skeptical acceptance wrt. preferred semantics in Dung's graphs (see previous parts of this proof and the proof of Theorem 9.2). Hence, these problems are Π_2^p–hard, and given the membership results, also Π_2^p-complete.

Hardness for SKEPT-SAT-UMIN can be shown through skeptical acceptance under semi–stable semantics in Dung's graph, which is Π_2^p-complete [36, 22] (see also previous parts of this proof and the proof of Theorem 9.2) Thus, SKEPT-SAT-UMIN is Π_2^p-complete.

$\square$

Theorem 6.11. *DEFCOV is in coDP.*

Proof. An argument $\mathtt{A}$ is default covered in an epistemic graph iff there is a value $x \in [0,1]$ s.t. $\mathcal{C} \Vdash p(\mathtt{A}) \neq x$. Based on the definition of $\Vdash$, this means that an argument is default covered iff $\{x \mid P \in \mathsf{Sat}(\mathcal{C}), P(\mathtt{A}) = x\} \neq [0,1]$, i.e. there is at least one value not produced by any satisfying distribution. We can therefore observe that this problem is complementary to PCSAT-INT. This puts DEFCOV in coDP.

$\square$