From Logic Programming to Human Reasoning: How to be Artificially Human

Dissertation

zur Erlangung des akademischen Grades
Doktor rerum naturalium (Dr. rer. nat.)

vorgelegt an der
Technischen Universität Dresden
Fakultät Informatik

eingereicht von
Emmanuelle-Anna Dietz Saldanha, MSc
geb. am 30. Juni 1986 in Saarbrücken

Gutachter: Prof. Dr. habil. Steffen Hölldobler
Technische Universität Dresden
Prof. Dr. Antonis Kakas
University of Cyprus
Prof. Dr. Luís Moniz Pereira
Universidade Nova de Lisboa

Abstract

Results of psychological experiments have shown that humans make assumptions, which are not necessarily valid, that they are influenced by their background knowledge and that they reason non-monotonically. These observations show that classical logic does not seem to be adequate for modeling human reasoning. Instead of assuming that humans do not reason logically at all, we take the view that humans do not reason *classical* logically. Our goal is to model episodes of human reasoning and for this purpose we investigate the so-called Weak Completion Semantics. The Weak Completion Semantics is a Logic Programming approach and considers the least model of the weak completion of logic programs under the three-valued Łukasiewicz logic.

As the Weak Completion Semantics is relatively new and has not yet been extensively investigated, we first motivate why this approach is interesting for modeling human reasoning. After that, we show the formal correspondence to the already established Stable Model Semantics and Well-founded Semantics. Next, we present an extension with an additional *context* operator, that allows us to express negation as failure. Finally, we propose a contextual abductive reasoning approach, in which the context of observations is relevant. Some properties do not hold anymore under this extension.

Besides discussing the well-known psychological experiments Byrne's suppression task and Wason's selection task, we investigate an experiment in spatial reasoning, an experiment in syllogistic reasoning and an experiment that examines the belief-bias effect. We show that the results of these experiments can be adequately modeled under the Weak Completion Semantics. A result which stands out here, is the outcome of modeling the syllogistic reasoning experiment, as we have a higher prediction match with the participants' answers than any of twelve current cognitive theories.

We present an abstract evaluation system for conditionals and discuss well-known examples from the literature. We show that in this system, conditionals can be evaluated in various ways and we put up the hypothesis that humans use a particular evaluation strategy, namely that they prefer abduction to revision. We also discuss how relevance plays a role in the evaluation process of conditionals. For this purpose we propose a semantic definition of relevance and justify why this is preferable to a exclusively syntactic definition. Finally, we show that our system is more general than another system, which has recently been presented in the literature.

Altogether, this thesis shows one possible path on bridging the gap between Cognitive Science and Computational Logic. We investigated findings from psychological experiments and modeled their results within one formal approach, the Weak Completion Semantics. Furthermore, we proposed a general evaluation system for conditionals, for which we suggest a specific evaluation strategy. Yet, the outcome cannot be seen as the ultimate solution but delivers a starting point for new open questions in both areas.

Acknowledgements

Steffen, thank you for offering me the unique opportunity to prepare my thesis. I feel very grateful for having been your PhD student. The topic of the thesis, the development thereof and the end result would not have been imaginable without your support. Your ideas and your many inputs helped me develop and structure the content of the thesis. Thank you for your guidance and your advice allowing me to learn about the intricacies of research and the skills to navigate through the academic jungle of presenting and publishing. Thank you also for introducing me to so many people who supported me throughout this process.

Luís, thank you for having accepted to be my second supervisor and for having given me the possibility to stay at UNL in 2013/2014. I highly appreciated our discussions and your feedback on the thesis. I was very impressed about your extensive and versatile knowledge in so many different areas. It was a pleasure learning from you; especially during your explanations of the connections between different disciplines, as if all of them were just a logical and unified ensemble.

Marco, thank you for sharing your knowledge about Cognitive Science and about cognitive scientists. You shifted my focus to different issues and guided me towards various directions outside of the Computational Logic area, which I would have never thought of. Your expertise gave the work we did together another dimension of relevance. If I ever manage to gain ground in the Cognitive Science community, then it is due to your influence.

Jim, thank you for having given me the possibility to stay at SFU in 2012 and then again in 2014. I very much enjoyed our conversations about Well-founded Semantics and Stable Models Semantics. This allowed me to gain an easy access to a topic, which is not that easy. Jim and Victoria, I am very grateful for your warm hospitality, for all the effort you did to make me feel comfortable, and in particular for the hiking and dinners we had together.

Tony, thank you for having accepted to be the external reviewer of my thesis. It was not easy to deal with the amount of criticism that came from the Logic Programming community in regard to my approach back than at the LPNMR conference 2015. Our conversation and your strong attitude about this topic, helped reassure me that I was on the right track.

I would also like to thank all my former and current colleagues in the group. In particular, Bertram Fronhöfer, Tobias Philipp, and Christoph Wernhard who were always available to help solve problems pertaining to all kinds of formal issues, in particular Logic Programming. Tobias Philipp, thank you for reading my thesis and giving me constructive feedback. This has been extremely helpful. Bertram Fronhöfer, thank you

for providing me with critical comments on the introduction and conclusions of my thesis and on the topic itself. All three of you have given me and my work tremendous support. Further, I would like to thank Sylvia Wünsch, Bettina Weser, Romy Thieme and Susann Gierth. Whenever I needed help for any kind of bureaucratic matter, you were always willing to help me immediately.

Finally, I want to thank the individuals who had to endure all kinds of moods due to stress or excitement of whatever happened because of my work. I would like to thank my mother Claire and my father Wolfgang, and my two sisters, Maria and Stephanie, for always being so supportive. Your strong belief in me and your helpful advices allow me to pursue what is important for me in life. Finally, thanks to my husband Márcio for proof reading my thesis. In fact, thank you for always staying calm and giving me the necessary support for everything that is and that is not related to my work. I am very grateful that we have met and even more that we have such a wonderful life together.

Contents

1. Introduction

In the last century classical logic has played an important role as a normative system for psychologists investigating human reasoning. Psychological research, however, showed that humans systematically deviate from the classical logically correct answers, and therefore classical logic does not seem to be appropriate to model human reasoning. Until now there are no widely accepted theories that express formal representations of human reasoning.

As the area of human reasoning is versatile and complex, we will only deal with one of its aspects, namely the question how can we adequately model episodes of human reasoning with respect to conditionals. Our goal is to develop a methodology that allows us to formalize episodes of human reasoning with respect to conditionals. Before we can formalize reasoning that models human behavior we need to understand how humans draw certain conclusions given some specific information. Conventional formal approaches such as classical logic are not appropriate for this purpose because they deal with the elementary aspects that humans face when reasoning, such as incomplete, inconsistent or updated information in a way completely different from human behavior.

Let us consider a famous psychological study from the literature, *Byrne's suppression task* [Byrne, 1989]. This experiment shows that people with no prior exposure to formal logic suppress previously drawn conclusions when additional information becomes available. Interestingly, in some instances the previously drawn conclusions were valid whereas in other instances the next drawn conclusions were invalid with respect to classical logic. Consider the following example: '*If she has an essay to finish, then she will study late in the library*' and '*she has an essay to finish.*' Most participants (96%) conclude: '*she will study late in the library.*' If participants, however, receive an additional conditional: '*if the library stays open, she will study late in the library*' then only 38% of them conclude: '*she will study late in the library.*' This shows that, although the conclusion is still correct with respect to classical logic, the conclusion is suppressed by an additional conditional. This is an excellent example of the human capability to draw *non-monotonic* inferences. The participants received the following three conditionals:

Simple If she has an essay to finish, then she will study late in the library.

Alternative If she has a textbook to read, then she will study late in the library.

Additional If the library stays open, then she will study late in the library.

The participants were divided into three groups: the first group received the simple conditional; the second group received the simple and the alternative conditional, and

Part	Fact	Simple	Alternative	Additional
I	She has an essay to finish (E)	L (96%)	L (96%)	L **(38%)**
	She does not have an essay to finish ($\overline{E}$)	$\overline{L}$ (46%)	$\overline{L}$ **(4%)**	$\overline{L}$ (63%)
II	She will study late in the library (L)	E (53%)	E **(16%)**	E (55%)
	She will not study late in the library ($\overline{L}$)	$\overline{E}$ (69%)	$\overline{E}$ (69%)	$\overline{E}$ **(44%)**

Table 1.1.: The results of Byrne's suppression task.

the third group received the simple and the additional conditional. The task was split into two parts. The first part was as follows: The participants got either the fact '*she has an essay to finish*' (E) or the negation of it, '*she does not have an essay to finish*' ($\overline{E}$). In the second part part the participants got either the fact '*she will go to the library*' (L) or the negation of it, '*she will not study late in the library*' ($\overline{L}$). Based on the given information, they had to draw conclusions. Table 1.1 presents the experimental findings of Byrne for each case, with percentages in brackets. Similar results are shown among others by Dieussaert, Schaeken, Schroyens, and D'Ydewalle [2000].

Psychological results which confirm that humans do not reason according to classical logic could be used as an evidence against the suitability of logic in general for modeling human reasoning. Do humans reason logically in the first place? We are convinced that they do so and instead claim that it is the *classical* logic which is not adequate. Humans might have in mind a particular representation of conditionals and might reason with a logic that is different than the classical one. Some alternatives to classical logic have already been proposed in the area of Computational Logic such as non-monotonic logics, commonsense reasoning or many-valued logics. Furthermore, in the field of Artificial Neural Networks human reasoning processes are attempted to be understood and simulated. These approaches are more expressive than classical logic. Unfortunately, most of them are purely theoretical and have never been applied to real case studies. Instead, artificial examples are constructed, which only show that the theory works within that very specific context. But what is the value of a theory for human reasoning that has never been tested on how humans actually reason?

In Artificial Intelligence, one commonly used requirement is that if computational models are biologically plausible then they should also exhibit behavior similar to that of the biological brain [Herrmann and Ohl, 2009]. In Cognitive Science it is common to evaluate theories by performing reasoning experiments on subjects. For instance, Knauff [1999] and Renz, Rauh, and Knauff [2000] investigate which kind of information humans use when representing and remembering spatial arrangements in Allen's interval calculus.

In the last century, scientists from various fields, for instance in social sciences, have investigated human behavior and reasoning in general. These fields have a great expertise in the area of human reasoning. However, their theories are usually formulated in natural language and are not formalized. Yet in the area of Computer Science the knowledge

about human reasoning is not existing or very limited. Only a few have actually made the effort to investigate the literature from Cognitive Science or Psychology to understand the cognitive theories presented there. An exchange would be a great opportunity for both areas: On the one hand, investigating the current psychological results can help computer scientists to understand what human reasoning is about and why it is too complex for being evaluated by a few toy examples, as has been done in the past. On the other hand, the formal approaches of Computer Science and the flexible techniques of Artificial Intelligence can offer the required tools to formalize the elaborated cognitive theories. These formalizations would possibly facilitate communication and theories could be formally compared to one another.

In order to cope with the goal of this thesis, to develop a methodology that allows us to formalize episodes of human reasoning with respect to conditionals, we need to investigate the findings in both areas, in Cognitive Science and in Computational Logic. We try to bridge the gap between these areas by understanding and formalizing psychological results on the one hand and by evaluating the differences and the suitability of formal techniques on the other hand.

1.1. Intuitive and Reflective Minds

Kowalski [2011] argues that the relationship between logic and thinking lies in that logic deals with formalizing the laws of thoughts. Logic is mainly concerned with normative theories, that is, how people ought to think. In Cognitive Psychology on the other hand, the focus lies on descriptive theories, that is, how people actually think. Their thoughts are not subject to any judgment, but the central questions are about how they think and why they come to certain conclusions. According to Kowalski, Computational Logic is a dual process theory which combines both theories.

Evans [2012] explains and discusses extensively the conflict between logic and belief in human reasoning. He says that rationality is instrumental, in the sense that we act in a certain way to achieve our goals. In his book, he distinguishes between the old and the new mind rationality, or the intuitive and the reflective mind, respectively. The behaviour of the intuitive mind is instrumentally conditioned to reward and punishment. Its rationality is primarily that of the genes and it does not adapt to the given circumstances. Additionally, it is entirely driven by individual past experience and therefore it is unsuitable for reflection. This makes the intuitive mind vulnerable when the environment changes. On the other side, the reflective mind is primarily directed by goals motivated by us as individuals. It is driven by curiosity and it is deliberate. This allows us to think flexibly and solve problems in new and unforeseen settings, which also gives us the ability to react against unacceptable circumstances. As the intuitive mind, it is also instrumental, however it differs with respect to the goals which are pursued and with respect to the mental resources that are necessary. The reflective mind's motivation is influenced by complex emotions, where the goals are directed towards the future, and

which can imagine possibilities, make suppositions and simulate future events. This is done by having explicit knowledge, working memory, meta-representation, the ability to engage in novel thinking and reasoning. Evans calls this dual view on the human mind the *two minds hypothesis*. It is supported by experiments that have shown the *belief-bias effect* [Evans, 2012]: The belief-bias effect is the conflict between the reflective and intuitive minds when reasoning about problems involves real-world beliefs. It is the tendency to accept or reject arguments based on own beliefs or prior knowledge rather than on the reasoning process. The belief-bias effect has been detected in a number of psychological experiments about deductive reasoning. These experiments demonstrated possibly conflicting processes at the logical and psychological level. Chapter 8 investigates a psychological experiment about the belief-bias effect.

The distinction between normative and descriptive theories by Kowalski [2011] and Evan's two minds hypothesis seems to have some similarity. Normative theories correspond roughly to reflective thinking and descriptive theories correspond roughly to intuitive thinking.[1]

1.2. Cognitive Adequacy

Just modeling is not satisfying: Strube [1992] argues that knowledge engineering should also aim at being *cognitively adequate*. Accordingly, when evaluating computational approaches which try to explain human reasoning we insist on assessing their cognitive adequacy. Strube distinguishes between *weak* and *strong cognitive adequacy*: Weak cognitive adequacy requires the system to be ergonomic and user-friendly, whereas strong cognitive adequacy involves an exact model of human knowledge and reasoning mechanisms that follows the relevant human cognitive processes.

The concept of adequacy has originally been defined in a linguistic context to compare and explain language theories and their properties, for which there are two different measures: *conceptual adequacy* and *inferential adequacy*. Conceptual adequacy reflects on how far the language represents the content correctly. Inferential adequacy is about the procedural part when the language is applied to the content [Strube, 1996].

Knauff, Rauh, and Schlieder [1995] and Knauff, Rauh, and Renz [1997] define cognitive adequacy in the setting of qualitative spatial reasoning, where they make a similar distinction: The authors distinguish between *conceptual* cognitive adequacy and *inferential* cognitive adequacy. The degree of conceptual adequacy reflects to what extent a system corresponds to human conceptual knowledge. Inferential adequacy focuses on the procedural part and indicates whether the reasoning process of a system is structured similarly to the way humans reason. This is analogous to the proposal made

[1] Luís Moniz Pereira states yet another view, namely that reactive thinking may be the result of mere intuition, or else the result of a prior compilation of reflective thinking that avoids repeating (reflectively) thinking about it. (personal communication, February 10, 2016)

by Stenning and van Lambalgen [2005, 2008] to model human reasoning by a two step approach: First, human reasoning should be modeled by setting up an *appropriate representation* (conceptual adequacy) and, second, the *reasoning process* should be modeled with respect to this representation (inferential adequacy).

McCarthy [1977] foresaw epistemological issues of artificial intelligence and illustrates this with the paradox of AI-systems which solve simple problems slower than difficult ones. Accordingly, Bibel [1991] states that a method is adequate if 'for any given knowledge base, the model solves simpler problems faster than more difficult ones'. Shastri and Ajjanagadde [1993] present SHRUTI, a connectionist model that efficiently encodes a large amount of facts and rules while performing fast on it by efficiently implementing inferences. Their main motivation is an attempt to resolve the paradox McCarthy stated, namely, the gap between the ability of humans to "draw a variety of inferences effortlessly, spontaneously, and with remarkable efficiency" on the one hand and the "results about the complexity of reasoning reported by researchers in artificial intelligence" on the other hand [Shastri and Ajjanagadde, 1993]. They say that the intuitive mind has to deal with a huge amount of data from the facts and rules point of view. Hence, they state that the complexity of an algorithm that simulates this reasoning process should be in the best case optimal or independent from the amount of data someone has to deal with. Similar to the distinction in Cognitive Science, Shastri and Ajjandagadde implicitly differentiate between conceptual and inferential adequacy. Their rules and facts are encoded in first-order logic. First, they discuss the representation of facts in the program. They consider static and dynamic bindings and short- and long-term facts. After that, they consider the approach of how to dynamically encode rules and propagate dynamic bindings.

By taking Bibel's [1991] approach as a starting point, Beringer and Hölldobler [1993] show that Shastri and Ajjandagadde's approach is not more than reasoning by reductions. They conclude that if it is really the case that the effortlessly and spontaneously reasoning processes of humans can be expressed as Shastri and Ajjandagadde have done it, then these problems are just reasoning by reductions and do not present an AI paradox at all, but are much simpler than the AI community thought. Hölldobler and Thielscher [1994] state that adequacy implies massive parallelism and Herrmann and Reine [1996] discuss adequate learning in neural networks.

1.3. A Novel Cognitive Theory

Let us consider again the suppression task. It is straightforward to see that classical logic cannot model this task adequately. At least some kind of non-monotonicity is needed. As appropriate representation to model the suppression task, Stenning and van Lambalgen [2005, 2008] propose logic programs under completion semantics based on the three-valued logic used by Fitting [1985], which itself is based on the three-valued logic of Kleene [1952]. Unfortunately, some technical claims made by Stenning and

van Lambalgen are wrong. Hölldobler and Kencana Ramli [2009a,b] have shown that the three-valued logic proposed by Stenning and van Lambalgen is inadequate for the suppression task, but that the suppression task can be adequately modeled if the three-valued logic of Łukasiewicz [1920] is used instead. The computational logic approach in [Hölldobler and Kencana Ramli, 2009b, Dietz, Hölldobler, and Ragni, 2012a] models the suppression task by means of logic programs under the so-called *Weak Completion Semantics*, a three-valued variation of Clark's completion. They show that the conclusions drawn with respect to least models correspond to the findings by Byrne [1989] and conclude that the derived logic programs under the three-valued Łukasiewicz semantics are inferentially cognitively adequate for the suppression task.[2] We will discuss this formalization of Byrne's suppression task in Chapter 5.1. Motivated by these findings, we decide to take the Weak Completion Semantics as starting point and as underlying approach for the investigations in this thesis.

1.4. Contributions

The thesis attempts to further link Cognitive Science and Computational Logic. We point to related work in both fields and give a comprehensive overview of the state-of-the-art research. The goal of this thesis, to develop a methodology that allows us to formalize episodes of human reasoning with respect to conditionals, has been carried out as follows: We investigated conditionals in human reasoning and attempted to formalize the results within a Logic Programming approach, the Weak Completion Semantics.

Background and Correspondence to other (three-valued) Semantics

- Background

  We provide the first introduction to a novel cognitive theory, the Weak Completion Semantics and an implementation thereof in Prolog. Additionally, we investigate how two-valued abduction can be extended to three-valued abduction and show their correspondence. Moreover, we give a new characterization of integrity constraints under the Weak Completion Semantics.

- Correspondence to other Semantics [Dietz, Hölldobler, and Wernhard, 2014]

  – We consider several well-established approaches in Logic Programming such as the Well-founded Semantics and the Stable Model Semantics and show the formal correspondence to the Weak Completion Semantics. Two main differences can be identified: One difference lies on the treatment of undefined atoms in programs, where the Well-founded Semantics and the Stable Model

[2] Wernhard [2011, 2012] discusses the application of different logic programming semantics to model human reasoning tasks according to the approach by Stenning and van Lambalgen and the roles of three-valuedness in this context, within a different technical framework based on circumscription.

Semantics assume the closed-world assumption, whereas the Weak Completion Semantics assumes the open-world assumption. The second difference is about tight programs: Under certain circumstances, atoms that are involved in a positive cycle in a program, are false under the Well-founded Semantics, whereas they stay unknown under the Weak Completion Semantics.

  - We further establish an overview of related semantics and show the relations to the well-known two-valued semantics, such as the (Well)-supported Model Semantics and Clark's Completion Semantics.

- Psychological Investigation on Cycles [Dietz, Hölldobler, and Ragni, 2013]

  As the Weak Completion Semantics and the Well-founded Semantics deal differently with positive cycles in logic programs, we carried out a psychological study about positive cyclic conditionals. It seems that the participants understood positive cyclic conditionals of length 1 differently than positive cyclic conditionals of length 2 or 3. In the first case, they understood the conditionals as facts, whereas in the second case they understood the conditionals as actual conditionals. Preliminary results show that the participants' understanding of positive cyclic conditionals seems to be in favor of the way how the Weak Completion Semantics treats positive cycles in logic programs.

Contextual Reasoning

- Contextual Programs [Dietz Saldanha, Hölldobler, and Pereira, 2017b]

  - We extend the programs under the Weak Completion Semantics with an additional truth-functional operator, ctxt, and introduce so-called contextual programs and provide an implementation thereof in Prolog. The ctxt operator can be seen as a mapping of three-valuedness to two-valuedness and allows us to express negation as failure under the Weak Completion Semantics.

  - We reconsider former formal results of the Weak Completion Semantics and show that for contextual programs the $\Phi_{\mathcal{P}}$ operator is not monotonic anymore. Further, the $\Phi_{\mathcal{P}}$ operator does not necessarily have a least fixed point for this class of programs. However, we can guarantee that a least fixed point exists for the class of acyclic contextual programs.

- Contextual Abduction [Pereira, Dietz, and Hölldobler, 2014a,b, Dietz Saldanha, Hölldobler, and Pereira, 2017b]

  - We present a contextual abductive approach, that allows us to express a preference among explanations, where the context in which information is observed plays a central role.

  - Further, this approach allows us to define more fine-grained relations between observations. We specify when an observation is a contextual (contestable) side-effect of another observation or whether they are both (jointly supported) contextual consequences of one another.

Modeling Human Reasoning Tasks

- Byrne's Suppression Task and the Well-founded Semantics [Dietz, Hölldobler, and Ragni, 2012a, Dietz, Hölldobler, and Wernhard, 2014]

  We reconsider Byrne's suppression task and discuss open questions with respect to the formalization. Furthermore we show how the respective logic programs need to be adapted in order to adequately model the task within the Well-founded Semantics.

- Wason's Selection Task [Dietz, Hölldobler, and Ragni, 2013]

  We model Wason's selection task by taking Kowalski's interpretation of the differences between the social and the abstract case as starting point for our formalization.

- Spatial Reasoning [Dietz, Hölldobler, and Höps, 2015a]

  We model a spatial reasoning task by taking the ideas of the Preferred Model Theory as starting point. For this purpose a logic program representation of the first free fit technique is provided: The idea is that a new to be included object will be either placed directly next to the already existing one, provided that there is space left, or otherwise this new object will be placed in the next available space. As psychological experiments have shown, this technique seems to be cognitively adequate. Our results on the spatial reasoning task complies with the results of these psychological experiments.

- Modeling Quantified Statements [Dietz, Hölldobler, and Ragni, 2015d, Costa, Dietz, Hölldobler, and Ragni, 2016]

  - We develop five principles for the representation of quantified statements motivated by Logic Programming techniques and findings from Cognitive Science. We then propose a representation of the four possible quantified statements as logic programs.

  - We predict the answers of 64 syllogistic premises and compare them to the results of psychological experiments. The Weak Completion Semantics has a matching of 85%, which is quite a good result, considering that the best of 12 other recent cognitive theories, only has a matching of 84%.

  - The achievement of this contribution stands out here. For the first time we can evaluate the performance of the Weak Completion Semantics and compare our results to the results of other state-of-the-art approaches. This achievement

emphasizes that the Weak Completion Semantics has to be taken seriously as a competitive cognitive theory.

- Belief-Bias Effect [Dietz, 2017]
  - We develop an adequate modeling approach for Evans, Barston and Pollard's syllogistic reasoning task, taking the previous developed representations of quantified statements as starting point. We investigate the belief-bias effect and argue that the belief bias can occur either in the representational part or the reasoning part when modeling human reasoning.
  - For the representational part, we model the belief bias by means of abnormality predicates. For the reasoning part, we propose a new principle, in which we suggest that humans search for alternative models, when in the current one, no conclusion seems possible. For this purpose, we apply abduction, and show that by explaining the available information about the presented syllogistic premises, the belief bias can be adequately modeled.

The logic program representation of all human reasoning tasks in this thesis and their results under the Weak Completion Semantics have been implemented in Prolog or Java. To the best of our knowledge, no one has shown a formalization of so many different human reasoning episodes within a single approach. Furthermore, we are not aware of any formalization, that has attempted to model Evans, Barston and Pollard's syllogistic reasoning task or the belief-bias effect. We show for all tasks which steps need to be taken and motivated them by experimental findings from psychology. Summing up, the key findings are that the weak completion of a program should be favored over the completion of a program, that skeptical abduction should be favored over credulous abduction and that explanations should be minimal.

On Conditionals

- Evaluation System for Conditionals [Dietz, Hölldobler, and Pereira, 2015b]
  - We develop a novel system for the evaluation of conditionals in human reasoning. For this purpose, we introduce a simple revision operator and together with abduction, we show with the help of a few examples from the literature that we can basically model any outcome on how a conditional should be evaluated. The outcome depends mainly on the order in which we consider the conditions of a conditional. Moreover, in difference to the literature, where a conditional is either evaluated to true or false, the proposed system allows the possibility that conditionals can be evaluated to unknown.
  - We additionally discuss the issue of relevance within conditionals. We identify differences between weak relevance and strong relevance and suggest a semantical definition of the concept of relevance.

  - We conjecture that humans prefer abduction over revision, which seems to go along with what is assumed in the Cognitive Science literature.

- Correspondence to Schulz's Approach [Dietz and Hölldobler, 2015]

  We further show the correspondence of our system to another approach and show that our approach is more general.

1.5. Structure

The reason why the central parts of the thesis are divided into three parts might not be self-explanatory: Even though all three parts deal with human reasoning and in particular all parts take the Weak Completion Semantics as underlying approach, they have different subgoals.

The first part consists of three chapters. Chapter 2 introduces the preliminaries, including the Weak Completion Semantics, illustrated by examples of the defined notions and notations. Chapter 3 shows a formal correspondence between the Weak Completion Semantics and other semantics, in particular with respect to the Well-founded Semantics. We additionally show the relation between two- and three-valued approaches in general. Motivated by the limitations of the Weak Completion Semantics when modeling the famous Tweety example, Chapter 4 proposes to extend the Weak Completion Semantics with a new truth-functional operator and develops a contextual abductive approach. The goal of this first part is to allow the reader an easy access to the Weak Completion Semantics. We clarify where to categorize the Weak Completion Semantics in relation to the other already existing approaches. Furthermore, the last chapter of this part in which the Weak Completion Semantics is extended, shows why former formal properties of the Weak Completion Semantics do not hold anymore.

The second part is about modeling well-known human reasoning tasks within the Weak Completion Semantics. This part consists of the formalizations of Byrne's suppression task and Wason's selection task in Chapter 5, reasoning with spatial relations in Chapter 6, reasoning with quantified statements in Chapter 7 and reasoning with the belief-bias effect in Chapter 8. We apply various Logic Programming techniques such as abduction and integrity constraints for the formalization of these tasks. The goal of this part is to show how findings from Cognitive Science can be adequately modeled under the Weak Completion Semantics.

The goal of the third part emerges from the results of the first two parts: A general system for the evaluation of conditionals. We propose an abstract reduction system in Chapter 9 and motivate a preferred derivation with the help of examples from the literature. Finally, Chapter 10 shows a formal correspondence to another Logic Programming approach for the evaluation of conditionals.

1.6. Reading Paths

Not every reader might be equally interested in every chapter and section presented in this thesis. We suggest the following reading paths, depending on the readers' focus:

- For a general non-technical overview about the topic, the following chapters and sections may be considered:

$$\begin{array}{c} 1 \longrightarrow \\ 3.1 \longrightarrow 3.4 \longrightarrow 3.5 \longrightarrow 4.1 \longrightarrow 4.6 \longrightarrow \\ 6.1 \longrightarrow 6.2 \longrightarrow 6.6 \longrightarrow 7.1 \longrightarrow 7.2 \longrightarrow 7.5 \longrightarrow 8.1 \longrightarrow 8.2 \longrightarrow 8.3 \longrightarrow 8.5 \longrightarrow \\ 9.1 \longrightarrow 9.4 \longrightarrow 9.7 \longrightarrow \\ 11 \longrightarrow 12 \end{array}$$

- For a general introduction on the Weak Completion Semantics and the correspondence to related semantics, the following sections may be considered:

$$2.1 \longrightarrow 2.2 \longrightarrow 2.3 \longrightarrow 3.2 \longrightarrow 3.3$$

- The reader who is interested on how integrity constraints, abduction and further extensions work under the Weak Completion Semantics, may additionally consider the following chapters and sections:

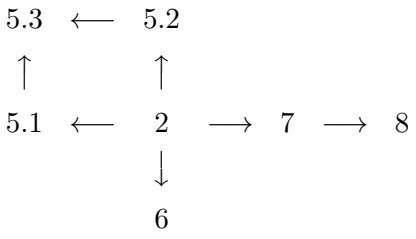
$$\begin{array}{ccccccc} & & & & 4 & & \\ & & & & \uparrow & & \\ 2.4 & \longrightarrow & 2.5 & \longrightarrow & 9 & \longrightarrow & 10 \end{array}$$

  where

  - Section 2.4 and 2.5 are about how integrity constraints and abduction work under the Weak Completion Semantics,
  - Chapter 4 presents a contextual abductive reasoning approach, and
  - Chapter 9 presents an abstract reduction system for conditionals that is compared to another approach in Chapter 10.

- The reader who is interested in the psychological reasoning tasks and their formal

representations, may consider the following chapters and sections:



where

- Chapter 2 is an elaborate introduction about the Weak Completion Semantics including abduction and integrity constraints,
- Section 5.1 is about the suppression task,
- Section 5.2 is about the selection task,
- Section 5.3 concludes with a few open questions about the Weak Completion Semantics and a summary,
- Chapter 6 is about a spatial reasoning task,
- Chapter 7 is about reasoning with quantified statements, and
- Chapter 8 is about a syllogistic reasoning task and the belief-bias effect.

- Readers who are interested in the technical aspects of the thesis, while avoiding the human reasoning tasks, can consider the following chapters and sections:

$$2 \longrightarrow 3.2 \longrightarrow 3.3 \longrightarrow 4 \longrightarrow 9 \longrightarrow 10$$

where

- Chapter 2 is an elaborate introduction about the Weak Completion Semantics including abduction and integrity constraints,
- Section 3.2 and Section 3.3 show the correspondence of related semantics to the Weak Completion Semantics,
- Chapter 4 presents a contextual abductive reasoning approach, and
- Chapter 9 presents an abstract reduction system for conditionals, where in
- Chapter 10, the correspondence of this system to another approach for conditionals is shown.

Part I.

Weak Completion Semantics

2. Preliminaries

We start with introducing the general notation and terminology that will be used throughout the thesis, which is based on [Lloyd, 1984, Hölldobler, 2009]. Section 2.2 introduces three-valued semantics. In Section 2.3, we discuss different least fixed point operators for the computation of least models. After that, we introduce integrity constraints and explain how they can be understood under three-valued semantics in Section 2.4 and extend two-valued abduction to three-valued abduction in Section 2.5.[1]

We assume that the reader is familiar with logic and logic programming. We consider an *alphabet* that consists of finite disjoint sets of *constants* and *predicate symbols*, the *truth-value constants true* $\top$, *false* $\bot$ and *unknown* U, a separate infinite set of *variables*, the *quantifier symbols* $\forall$ and $\exists$ and the usual connectives *negation* $\neg$, *disjunction* $\vee$, *conjunction* $\wedge$, *implication* $\leftarrow$, *equivalence* $\leftrightarrow$ and punctuation symbols "(", ",", ")". If a letter or the first letter of a word is written with an upper case, it is a variable; otherwise it is a constant or a predicate symbol.

The set of *terms* consists only of constants and variables. A *ground term* is a constant. *Formulas* are constructed in the usual way from the predicate symbols and terms, the truth-value constants, the quantifiers and the connectives. An atomic formula is called an *atom*. If A is an atom, then A and $\neg A$ are *literals*, the *positive literal* and the *negative literal*, respectively. A *ground formula* is a formula not containing free variables. A *language* $\mathcal{L}$ given by an alphabet consists of the set of all formulas constructed from the symbols of this alphabet.

2.1. Logic Programs

A *logic program* $\mathcal{P}$ is a finite set of clauses.

$$A \leftarrow L_1 \wedge \ldots\ldots \wedge L_n \tag{2.1}$$
$$A \leftarrow \top \tag{2.2}$$
$$A \leftarrow \bot \tag{2.3}$$

A is an atom and the L_i with $1 \leq i \leq n$ are literals. The atom A is called *head* of the clause and the subformula to the right of the implication symbol is called *body* of

[1] Section 2.4 has first been published in [Pereira, Dietz, and Hölldobler, 2014a,b]. Some parts of Section 2.5 have not been published and are contributions of this thesis.

the clause. Let us first consider clauses of the form (2.1): It is a *definite clause* if it only contains atoms in its body. A program that contains only definite clauses is a *definite program*. As we restrict terms to be either constants or variables only, we consider so-called data logic programs. If the clause contains variables, then they are implicitly universally quantified within the scope of the entire clause. For instance, the clause $p(X) \leftarrow q(X)$ represents the formula $(\forall X)(p(X) \leftarrow q(X))$. Therefore, a clause never contains free variables. Accordingly, a *ground clause* is a clause that does not contain variables. Clauses of the form (2.2) and (2.3) are called *facts* and *assumptions*, respectively. The notion of falsehood appears to be counterintuitive at first sight, but programs will be interpreted under their weak completion where this implication sign is replaced by an equivalence sign. We restrict the head of the facts and assumptions to be ground atoms, that means, they do not contain variables.

We introduce the following notation in order to refer to the positive and negative part of a body: If formula F is of the form $A_1 \wedge \cdots \wedge A_n \wedge \neg B_1 \wedge \cdots \wedge \neg B_m$ where A_i with $1 \leq i \leq n$ are atoms and $\neg B_j$ with $1 \leq j \leq m$ are negated atoms, then $\mathsf{pos}(F) = A_1 \wedge \cdots \wedge A_n$ and $\mathsf{neg}(F) = \neg B_1 \wedge \cdots \wedge \neg B_m$. An empty conjunction is semantically equivalent to true, therefore, if F does not contain any literal, $\mathsf{pos}(F) = \mathsf{neg}(F) = \top$. Accordingly, $\mathsf{pos}(\top) = \mathsf{neg}(\top) = \top$. To let pos and neg also be applicable to bodies of assumptions, we define additionally $\mathsf{pos}(\bot) = \mathsf{neg}(\bot) = \top$.

A *normal program* – in the standard sense used in the literature on logic programming – is a program that does not contain assumptions, that is, a program whose clauses are all of the form (2.1) or (2.2). If $\mathcal{P}$ is a program, then $\mathcal{P}^+$ denotes the normal program obtained from $\mathcal{P}$ by deleting all assumptions. Obviously, for normal programs $\mathcal{P}$ it holds that $\mathcal{P} = \mathcal{P}^+$.

Here, a *propositional program* $\mathcal{P}$ corresponds to a ground program. If $\mathcal{P}$ is not propositional, then $g\mathcal{P}$ denotes ground $\mathcal{P}$, which means that $\mathcal{P}$ contains exactly all the ground clauses with respect to the alphabet. As the set of constants is finite and $\mathcal{P}$ is finite, $g\mathcal{P}$ is finite as well. As we are in particular interested in ground programs, the following definitions will always be referring to $g\mathcal{P}$.

We assume a fixed non-empty and finite set of ground atoms, denoted by $\mathcal{A}\mathsf{t}$. If $\mathcal{P}$ is a program, then $\mathsf{atoms}(\mathcal{P})$ denotes the set of all atoms occurring in $g\mathcal{P}$. If not stated otherwise, we assume that $\mathcal{A}\mathsf{t} = \mathsf{atoms}(\mathcal{P})$.

An atom A is *defined* in $g\mathcal{P}$ if and only if $g\mathcal{P}$ contains a clause whose head is A; otherwise A is said to be *undefined*. The set of all atoms that are defined in $g\mathcal{P}$ is denoted by $\mathsf{defined}(\mathcal{P})$. The set of all atoms that are undefined in $g\mathcal{P}$, that is $\mathcal{A}\mathsf{t} \setminus \mathsf{defined}(\mathcal{P})$, is denoted by $\mathsf{undef}(\mathcal{P})$. The *definition of* $\mathcal{L}$ *in* $g\mathcal{P}$ is

$$\mathsf{def}(\mathcal{L}, \mathcal{P}) \quad = \quad \{A \leftarrow \mathit{body} \in g\mathcal{P} \mid A \in \mathcal{L} \text{ or } \neg A \in \mathcal{L}\},$$

where $\mathcal{L}$ is a set of (ground) literals. $\mathcal{L}$ is said to be *consistent* if and only if it does not contain a pair of *complementary literals*. The complementary literal of a literal L, is a

Example 2.1. Consider the program $\mathcal{P}$ consisting of three clauses:

$$\begin{array}{lcl} p(X) & \leftarrow & q(X) \wedge \neg r(X) \wedge s(X). \\ q(a) & \leftarrow & \top. \\ r(a) & \leftarrow & \bot. \end{array}$$

The second clause is a fact and the third clause is an assumption. Applying pos and neg to the body of the first clause gives the following result:

$$\mathsf{pos}(q(X) \wedge \neg r(X) \wedge s(X)) \quad = \quad q(X) \wedge s(X)$$

and

$$\mathsf{neg}(q(X) \wedge \neg r(X) \wedge s(X)) \quad = \quad \neg r(X).$$

The set of constants is

$$\mathcal{C} \quad = \quad \mathsf{constants}(\mathcal{P}) \quad = \quad \{a\}.$$

The ground program, $g\mathcal{P}$, consists of the following three clauses:

$$\begin{array}{lcl} p(a) & \leftarrow & q(a) \wedge \neg r(a) \wedge s(a). \\ q(a) & \leftarrow & \top. \\ r(a) & \leftarrow & \bot. \end{array}$$

The set of atoms, of defined atoms and of undefined atoms, are

$$\begin{array}{rcrcl} \mathcal{A}\mathsf{t} & = & \mathsf{atoms}(\mathcal{P}) & = & \{p(a), q(a), r(a), s(a)\}, \\ & & \mathsf{defined}(\mathcal{P}) & = & \{p(a), q(a), r(a)\}, \\ & & \mathsf{undef}(\mathcal{P}) & = & \{s(a)\}. \end{array}$$

The definition of $\mathcal{L} = \{q(a), r(a)\}$ in $\mathcal{P}$ is

$$\mathsf{def}(\mathcal{L}, \mathcal{P}) \quad = \quad \{q(a) \leftarrow \top, r(a) \leftarrow \bot\}.$$

The corresponding normal program $g\mathcal{P}^+$ consists of the following two clauses:

$$\begin{array}{lcl} p(a) & \leftarrow & q(a) \wedge \neg r(a) \wedge s(a). \\ q(a) & \leftarrow & \top. \end{array}$$

literal corresponding to the negation of L, i.e. $\neg L$. When writing sets of literals we will omit curly brackets if the set has only one element.

We assume a fixed set of constants, denoted by $\mathcal{C}$, which is non-empty and finite. If $\mathcal{P}$ is a program, then $\mathsf{constants}(\mathcal{P})$ denotes the set of all constants occurring in $\mathcal{P}$. If not stated otherwise, we assume that $\mathcal{C} = \mathsf{constants}(\mathcal{P})$. Example 2.1 clarifies the just introduced definitions. We assume that each non-propositional program contains at least one constant symbol. The language $\mathcal{L}$ underlying a program $\mathcal{P}$ contains precisely the predicate and constant symbols occurring in $\mathcal{P}$, and no others.

When mechanisms of non-monotonic reasoning are applied to model human reasoning, it seems essential that only certain atoms are subjected to the closed-world assumption, while others are considered to follow the open-world assumption. Under the closed-world assumption all atoms are expected to be false if not stated otherwise.

Consider the following transformation for a given program $\mathcal{P}$:

1. For all $A \in \mathsf{atoms}(\mathcal{P})$ replace $\mathsf{def}(A, \mathcal{P}) = \{A \leftarrow \mathit{body}_1,\ A \leftarrow \mathit{body}_2,$ $\ldots,\ A \leftarrow \mathit{body}_n\}$, where $n \geq 1$, by $A \leftarrow \mathit{body}_1 \vee \mathit{body}_2 \vee \ldots \vee \mathit{body}_n$.

2. For all $A \in \mathsf{undef}(\mathcal{P})$ add $A \leftarrow \bot$.

3. Replace all occurrences of $\leftarrow$ by $\leftrightarrow$.

The resulting set of equivalences is the well-known Clark's *completion* of $\mathcal{P}$, denoted by $\mathsf{c}\,\mathcal{P}$ [Clark, 1978]. If step 2 is omitted, then the resulting set is the *weak completion* of $\mathcal{P}$, denoted by $\mathsf{wc}\,\mathcal{P}$ [Hölldobler and Kencana Ramli, 2009b]. As we will see later, the weak completion of a program allows both, closed-world assumption and open-world assumption, to coexist within a logic program. Consider Example 2.2 for which we show the completion and the weak completion of a program. In the following, we are interested in the weak completion of programs.

2.2. Three-valued Semantics

Under *two-valued semantics*, a *two-valued interpretation* I of a program $\mathcal{P}$ is a mapping of $\mathsf{atoms}(\mathcal{P})$ to $\{\top, \bot\}$. $I(F) = \top$ denotes that interpretation I maps formula F to $\top$ according to the corresponding logic. A *two-valued model* $\mathcal{M}$ of $\mathcal{P}$ is a two-valued interpretation where for which each clause C occurring in $\mathcal{P}$ it holds that $\mathcal{M}(C) = \top$.

We extend two-valued semantics to three-valued semantics, where the corresponding truth values are $\top$, $\bot$ and U, which mean *true*, *false* and *unknown*, respectively. A *three-valued interpretation* I is a mapping from $\mathsf{atoms}(\mathcal{P})$ to the set of truth values $\{\top, \bot, \mathrm{U}\}$.

Example 2.2. Consider the program $\mathcal{P}$ consisting of the following clauses:

$$\begin{array}{lcl} p(X) & \leftarrow & q(X). \\ p(X) & \leftarrow & r(X). \\ q(a) & \leftarrow & \bot. \end{array}$$

Then $g\mathcal{P}$ consists of the following clauses:

$$\begin{array}{lcl} p(a) & \leftarrow & q(a). \\ p(a) & \leftarrow & r(a). \\ q(a) & \leftarrow & \bot. \end{array}$$

The set of atoms, of defined atoms and of undefined atoms, are

$$\begin{array}{rcl} \mathcal{A}\mathsf{t} = \mathsf{atoms}(\mathcal{P}) & = & \{p(a), q(a), r(a)\}, \\ \mathsf{defined}(\mathcal{P}) & = & \{p(a), q(a)\}, \\ \mathsf{undef}(\mathcal{P}) & = & \mathcal{A}\mathsf{t} \setminus \{p(a), q(a)\} \quad = \quad \{r(a)\}. \end{array}$$

The completion of $\mathcal{P}$, $\mathsf{c}\, g\mathcal{P}$, consists of the following equivalences:

$$\begin{array}{lcl} p(a) & \leftrightarrow & q(a) \ \vee \ r(a). \\ q(a) & \leftrightarrow & \bot. \\ r(a) & \leftrightarrow & \bot. \end{array}$$

and the weak completion of $\mathcal{P}$, $\mathsf{wc}\, g\mathcal{P}$, consists of the following equivalences:

$$\begin{array}{lcl} p(a) & \leftrightarrow & q(a) \ \vee \ r(a). \\ q(a) & \leftrightarrow & \bot. \end{array}$$

The truth value of a given formula under a given interpretation is determined according to the corresponding logic. A three-valued interpretation is represented as a pair $I = \langle I^\top, I^\bot \rangle I = \langle I^\top, I^\bot \rangle$ of two disjoint sets of ground atoms, where

$$I^\top = \{A \mid I(A) = \top\} \text{ and } I^\bot = \{A \mid I(A) = \bot\}.$$

Atoms which do not occur in $I^\top \cup I^\bot$ are mapped to U.

There are two common ways to order three-valued interpretations, which, following Ruiz and Minker [1995], we call *truth ordering* ($\preceq_t$) and *knowledge ordering* ($\preceq_k$). Given two interpretations, I and J,

$$I \preceq_t J \qquad \text{if and only if} \qquad I^\top \subseteq J^\top \qquad \text{and} \qquad I^\bot \supseteq J^\bot.$$

The positive interpretation $I^\top$ is minimized and the negative interpretation $I^\bot$ is maximized. This is different for the knowledge ordering. Again, given that I and J are two interpretations,

$$I \preceq_k J \qquad \text{if and only if} \qquad I^\top \subseteq J^\top \qquad \text{and} \qquad I^\bot \subseteq J^\bot.$$

Here both, the positive interpretation $I^\top$ and the negative interpretation $I^\bot$, are minimized. The intersection and the union of two interpretations $I = \langle I^\top, I^\bot \rangle$ and $J = \langle J^\top, J^\bot \rangle$ are defined as

$$I \cap J \;=\; \langle I^\top \cap J^\top, I^\bot \cap J^\bot \rangle$$

and

$$I \cup J \;=\; \langle I^\top \cup J^\top, I^\bot \cup J^\bot \rangle,$$

respectively. A *three-valued model* $\mathcal{M}$ of $\mathcal{P}$ is a three-valued interpretation where for each clause $A \leftarrow \mathit{body}$ occurring in $\mathcal{P}$ it holds that $\mathcal{M}(A \leftarrow \mathit{body}) = \top$. Analogously, a *three-valued model* $\mathcal{M}$ of the weak completion of $\mathcal{P}$ is a three-valued interpretation where for each equivalence $A \leftrightarrow \mathit{body}_1 \vee \mathit{body}_2 \vee \ldots \vee \mathit{body}_n$, $n \geq 1$, occurring in $\mathsf{wc}\,\mathcal{P}$ it holds that $\mathcal{M}(A \leftrightarrow \mathit{body}_1 \vee \mathit{body}_2 \vee \ldots \vee \mathit{body}_n) = \top$. Three-valued models that are minimal with respect to the truth ordering or knowledge ordering are called *truth-minimal* or *knowledge-minimal models*, respectively. If there only exists one minimal model, then this model is called the least model. Likewise, three-valued models which are least with respect to the truth ordering or knowledge ordering are called *truth-least* or *knowledge-least models*. In the sequel, we implicitly assume all interpretations and models to be three-valued, if not explicitly stated otherwise. The distinction between both orderings are made clear in Example 2.3.

Since the first three-valued logic has been invented by Łukasiewicz [1920], various different interpretations of the three-valued connectives were proposed. Table 2.1 gives some common truth tables for negation, conjunction and disjunction. For implication and equivalence it shows different versions: Kleene [1952] introduced the implication

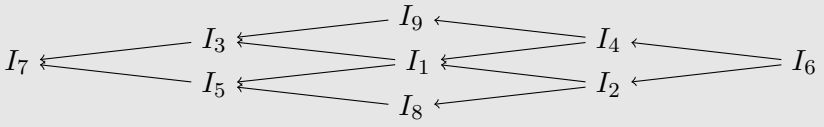
Example 2.3. Consider program $\mathcal{P}$ with the following two clauses:

$$\begin{array}{lcl} p(a) & \leftarrow & q(a). \\ q(a) & \leftarrow & \bot. \end{array}$$

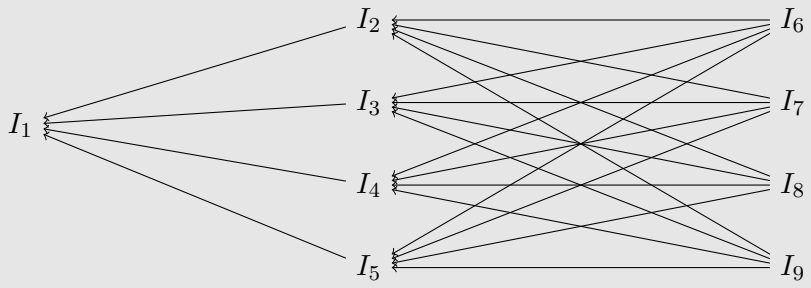
$\mathsf{atoms}(\mathcal{P})$ is $\{p(a), q(a)\}$ and accordingly, under three-valued logics, there are nine possible interpretations:

$$\begin{array}{lcllcllcl} I_1 & = & \langle \emptyset, \emptyset \rangle & I_2 & = & \langle \{p(a)\}, \emptyset \rangle & I_3 & = & \langle \emptyset, \{p(a)\} \rangle \\ & & & I_4 & = & \langle \{q(a)\}, \emptyset \rangle & I_5 & = & \langle \emptyset, \{q(a)\} \rangle \\ & & & I_6 & = & \langle \{p(a), q(a)\}, \emptyset \rangle & I_7 & = & \langle \emptyset, \{p(a), q(a)\} \rangle \\ & & & I_8 & = & \langle \{p(a)\}, \{q(a)\} \rangle & I_9 & = & \langle \{q(a)\}, \{p(a)\} \rangle \end{array}$$

Only I_1, I_2, I_5, I_6, I_7 and I_8 are models of $\mathcal{P}$. The following graph shows the truth ordering of the nine interpretations:



where $I_7 \leftarrow I_3$ means that $I_7 \preceq_t I_3$. The next graph shows the knowledge ordering of the nine interpretations:



Here, $I_1 \leftarrow I_2$ means that $I_7 \preceq_k I_3$.

F	$\neg F$
$\top$	$\bot$
$\bot$	$\top$
U	U

$\leftarrow_{\text{Ł}}$	$\top$	U	$\bot$
$\top$	$\top$	$\top$	$\top$
U	U	$\top$	$\top$
$\bot$	$\bot$	U	$\top$

$\leftrightarrow_{\text{Ł}}$	$\top$	U	$\bot$
$\top$	$\top$	U	$\bot$
U	U	$\top$	U
$\bot$	$\bot$	U	$\top$

$\wedge$	$\top$	U	$\bot$
$\top$	$\top$	U	$\bot$
U	U	U	$\bot$
$\bot$	$\bot$	$\bot$	$\bot$

$\leftarrow_{\text{S}}$	$\top$	U	$\bot$
$\top$	$\top$	$\top$	$\top$
U	$\bot$	$\top$	$\top$
$\bot$	$\bot$	$\bot$	$\top$

$\leftrightarrow_{\text{S}}$	$\top$	U	$\bot$
$\top$	$\top$	$\bot$	$\bot$
U	$\bot$	$\top$	$\bot$
$\bot$	$\bot$	$\bot$	$\top$

$\vee$	$\top$	U	$\bot$
$\top$	$\top$	$\top$	$\top$
U	$\top$	U	U
$\bot$	$\top$	U	$\bot$

$\leftarrow_{\text{K}}$	$\top$	U	$\bot$
$\top$	$\top$	$\top$	$\top$
U	U	U	$\top$
$\bot$	$\bot$	U	$\top$

$\leftrightarrow_{\text{K}}$	$\top$	U	$\bot$
$\top$	$\top$	U	$\bot$
U	U	U	U
$\bot$	$\bot$	U	$\top$

Table 2.1.: Truth tables for three-valued logics. The $\top$'s highlighted in gray indicate that formulas of the form $A \leftarrow B$ which are true under $\leftarrow_{\text{Ł}}$ are true under $\leftarrow_{\text{S}}$, and vice versa.

($\leftarrow_{\text{K}}$), whose truth table is identical to Łukasiewicz implication ($\leftarrow_{\text{Ł}}$) except in the case where precondition and conclusion are both mapped to U: In this case, the value of $\leftarrow_{\text{K}}$ is U, whereas the value of $\leftarrow_{\text{Ł}}$ is $\top$. The further common variant $\leftarrow_{\text{S}}$ of three-valued implication is called seq_3 introduced by Gottwald [2001].

The displayed versions of equivalence ($\leftrightarrow_{\text{Ł}}$, $\leftrightarrow_{\text{S}}$, $\leftrightarrow_{\text{K}}$) are derived by conjoining the respective implications with flipped arguments. We say that we consider the formula under Łukasiewicz semantics if we understand operators in a formula with the meaning specified in Table 2.1 for $\{\neg, \wedge, \vee, \leftarrow_{\text{Ł}}, \leftrightarrow_{\text{Ł}}\}$. Example 2.4 shows a particular case where $\leftarrow_{\text{Ł}}$ and $\leftarrow_{\text{S}}$ are different from $\leftarrow_{\text{K}}$.

Fitting [1985] combined the truth tables for $\neg$, $\vee$, $\wedge$ from Łukasiewicz with the equivalence $\leftrightarrow_{\text{S}}$ for investigations within logic programming. The set of connectives Fitting used is $\{\neg, \wedge, \vee, \leftrightarrow_{\text{S}}\}$.[2] Stenning and van Lambalgen [2008] suggested to model Byrne's suppression task by extending the logic used by Fitting with $\leftarrow_{\text{K}}$. If we understand operators in this way, that is, with the meanings of $\{\neg, \wedge, \vee, \leftarrow_{\text{K}}, \leftrightarrow_{\text{S}}\}$, we call this SvL-semantics. Hölldobler and Kencana Ramli [2009b] showed that SvL-semantics leads to technical errors. They proposed to use Łukasiewicz semantics (cf. Table 2.1), which corrects these and allows to adequately model Byrne's [1989] suppression task. The erroneous effects of the original suggestion by Stenning and van Lambalgen [2008] will be demonstrated by two examples in Chapter 5.

[2]Note that Fitting considered logic programs under their completion and did not specify the interpretation for the implication.

Example 2.4. Consider again $\mathcal{P}$ from Example 2.3:

$$\begin{array}{lcl} p(a) & \leftarrow & q(a). \\ q(a) & \leftarrow & \bot. \end{array}$$

Under Łukasiewicz semantics, all interpretations except of I_3 and I_4, are models of $g\mathcal{P}$. I_1 is its knowledge-least and I_7 is its truth-least model.
One should observe that in contrast to two-valued logic

$$p(a) \leftarrow_{\text{Ł}} q(a) \quad \text{and} \quad p(a) \leftarrow_{\text{S}} q(a)$$

are not semantically equivalent to $p(a) \vee \neg q(a)$. For interpretation I_1, where $I_1(A) = I_1(B) = \text{U}$, the following interpretations of disjunction and implication are

$$I_1(A \vee \neg B) = \text{U} \qquad \text{whereas} \qquad I_1(A \leftarrow_{\text{Ł}} B) \;=\; I_1(A \leftarrow_{\text{S}} B) \;=\; \top.$$

However, this is different for the $\leftarrow_{\text{K}}$ implication:

$$I_1(A \leftarrow_{\text{K}} B) \;=\; \text{U}.$$

Under the Well-founded Semantics, which we will discuss later, the interpretation of the implication corresponds to $\leftarrow_{\text{S}}$ [Przymusinski, 1989], which belongs to the three-valued logic $\mathsf{S_3}$ [Rescher, 1969], that is, $\{\neg, \wedge, \vee, \leftarrow_{\text{S}}, \leftrightarrow_{\text{S}}\}$. If we understand operators in a formula with these meanings, we say that we consider the S-semantics. Example 2.5 shows yet another way of understanding the truth values. As indicated by the highlighted $\top$ signs in Table 2.1, whenever a formula is evaluated under S-semantics, is true under $\leftarrow_S$ then this formula evaluated under Łukasiewicz semantics, is true under $\leftarrow_{\text{Ł}}$, and vice versa. The same holds for $\leftrightarrow_{\text{Ł}}$ and $\leftrightarrow_S$. From this follows that the models of a program or a set of equivalences obtained by completing a program under S-semantics are exactly the same as under Łukasiewicz semantics. Table 2.2 gives an overview of the three-valued semantics together with the sets of connectives. In the following, if not specified otherwise, we consider them as underlying semantics.

2.3. Computing Least Models

A logic program can have several models. How to know which model is the intended one? In Logic Programming and Computational Logic the intended models are often least models, if they exist. Least models of logic programs can often be specified as least fixed points of appropriate semantic operators [Apt and van Emden, 1982].

$$\mathsf{T}_{\mathcal{P}}(I) \;=\; \{A \mid A \leftarrow \mathit{body} \in \mathsf{def}(A, \mathcal{P}) \text{ and } I(\mathit{body}) = \top\}$$

Example 2.5. Consider $\mathcal{P}$ consisting of the following clause:

$$p(a) \quad \leftarrow \quad q(a).$$

Except of $I_3 = \langle \emptyset, \{p(a)\} \rangle$ and $I_4 = \langle \{q(a)\}, \emptyset \rangle$ from Example 2.3, all other interpretations are models of $\mathcal{P}$ under Łukasiewicz semantics and S-semantics. $I_1 = \langle \emptyset, \emptyset \rangle$ is the knowledge-least model and $I_7 = \langle \emptyset, \{p(a), q(a)\} \rangle$ is the truth-least model.
Note that, under SvL-semantics, $I_1 = \langle \emptyset, \emptyset \rangle$ is not a model: Both, $p(a)$ and $q(a)$ are mapped to unknown, which according to Table 2.1, maps the implication to unknown as well.
Przymusinski [1989] proposed to view an interpretation $I = \langle I^\top, I^\bot \rangle$ as a mapping from $\mathsf{atoms}(\mathcal{P})$ to the set $\mathcal{V} = \{0, \frac{1}{2}, 1\}$.

$$I(A) \quad = \quad \begin{cases} 1 & \text{if } A \in I^\top \\ 0 & \text{if } A \in I^\bot \\ \frac{1}{2} & \text{otherwise} \end{cases}$$

If $I(A)$ is 1 or 0, then $I(\neg A)$ is 0 or 1, respectively. In case $I(A) = \frac{1}{2}$, $I(\neg A) = \frac{1}{2}$ as well. The interpretation of a conjunction of literals is the smallest value among all of its literals. Przymusinski then defines that

$$I(A \leftarrow \mathit{body}) \quad = \quad \begin{cases} \top & \text{if } I(A) \geq I(\mathit{body}) \\ \bot & \text{otherwise} \end{cases}$$

A clause is true if and only if the truth value of the head is bigger or equal to the interpretation of the body, otherwise it is false. This corresponds to the interpretation of the implication under S-semantics.
Consider again I_3 and I_4: According to Przymusinski's ordering, $I(p(a))$ is smaller than $I(q(a))$ and therefore $I(p(a) \leftarrow q(a)) = \bot$.

Semantics	Abbreviations	Set of Connectives				
Fitting	(F)	$\neg$	$\wedge$	$\vee$		$\leftrightarrow_{\mathrm{S}}$
Kleene	(K)	$\neg$	$\wedge$	$\vee$	$\leftarrow_{\mathrm{K}}$	
Łukasiewicz	(Ł)	$\neg$	$\wedge$	$\vee$	$\leftarrow_{Ł}$	$\leftrightarrow_{Ł}$
S-semantics	(S)	$\neg$	$\wedge$	$\vee$	$\leftarrow_{\mathrm{S}}$	$\leftrightarrow_{\mathrm{S}}$
SvL-semantics	(SvL)	$\neg$	$\wedge$	$\vee$	$\leftarrow_{\mathrm{K}}$	$\leftrightarrow_{\mathrm{S}}$

Table 2.2.: Overview of the three-valued semantics together with the sets of connectives.

I is a two-valued interpretation and $\mathcal{P}$ a program. The least fixed point of the $\mathsf{T}_\mathcal{P}$ operator ($\mathsf{lfp}\,\mathsf{T}_\mathcal{P}$) corresponds to the least two-valued model of $\mathcal{P}$ ($\mathsf{lm}_2\mathcal{P}$), if it exists. Let us define the consequence relation, $\models_{\mathsf{T}_\mathcal{P}}$, where, given a program $\mathcal{P}$ and an atom A, $\mathcal{P} \models_{\mathsf{T}_\mathcal{P}} A$ if and only if $A \in \mathsf{lfp}\,\mathsf{T}_\mathcal{P}$. If $\mathcal{P}$ is definite, then it has always a least model. However, this does not necessarily hold, if $\mathcal{P}$ is not definite.

As already mentioned, Hölldobler and Kencana Ramli [2009b] proposed the Weak Completion Semantics, an approach that extends the two-valued semantics to (three-valued) Łukasiewicz semantics. The model intersection property holds for logic programs and their weak completion under Łukasiewicz semantics.

$$\cap\{I \mid I \models_{Ł} \mathcal{P}\} \quad \models_{Ł} \quad \mathcal{P} \qquad \text{and} \qquad \cap\{I \mid I \models_{Ł} \mathsf{wc}\,\mathcal{P}\} \quad \models_{Ł} \quad \mathsf{wc}\,\mathcal{P},$$

given that I is an interpretation and $\mathcal{P}$ a program, $I \models_{Ł} \mathcal{P}$ holds if and only if each clause occurring in $\mathcal{P}$ is true under Łukasiewicz semantics. Analogously, $I \models_{Ł} \mathsf{wc}\,\mathcal{P}$ holds if and only if each equivalence occurring in $\mathsf{wc}\,\mathcal{P}$ is true under Łukasiewicz semantics. This property guarantees that each logic program and its weak completion has a (knowledge-) least model. Additionally, the least model of the weak completion of a program $\mathcal{P}$ under Łukasiewicz semantics ($\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}$) is identical to the least fixed point of the following semantic operator, $\Phi_\mathcal{P}$, which was introduced by Stenning and van Lambalgen [2008] for propositional programs and has been generalized for first-order programs in [Hölldobler and Kencana Ramli, 2009a]. Let I be an interpretation and $\mathcal{P}$ be a program. Then the application of Φ to I and $\mathcal{P}$, denoted by $\Phi_\mathcal{P}(I)$, is the interpretation $J = \langle J^\top, J^\bot \rangle$.

$$\begin{aligned} J^\top &= \{A \mid A \leftarrow \mathit{body} \in \mathsf{def}(A, \mathcal{P}) \text{ and } I(\mathit{body}) = \top\} \\ J^\bot &= \{A \mid \mathsf{def}(A, \mathcal{P}) \neq \emptyset \text{ and} \\ &\qquad \text{for all } A \leftarrow \mathit{body} \in \mathsf{def}(A, \mathcal{P}) \text{ we find that } I(\mathit{body}) = \bot\} \end{aligned}$$

Proposition 3.21 in [Kencana Ramli, 2009] shows that the $\Phi_\mathcal{P}$ operator is monotonic, i.e. given a program $\mathcal{P}$ and two interpretations I and J, if $I \subseteq J$ then $\Phi_\mathcal{P}(I) \subseteq \Phi_\mathcal{P}(J)$. Additionally, the $\Phi_\mathcal{P}$ operator guarantees a least fixed point for all programs, which has been shown in [Hölldobler and Kencana Ramli, 2009b]. Example 2.6 shows for some program, how the least fixed point of $\Phi_\mathcal{P}$ ($\mathsf{lfp}\,\Phi_\mathcal{P}$) can be computed.

Example 2.6. Consider again $g\mathcal{P}$ from Example 2.2:

$$\begin{array}{lcl} p(a) & \leftarrow & q(a). \\ p(a) & \leftarrow & r(a). \\ q(a) & \leftarrow & \bot. \end{array}$$

Let us start computing $\Phi_{\mathcal{P}}$ with $I_0 = \langle \emptyset, \emptyset \rangle$:

$$\begin{array}{lcccl} \Phi_{\mathcal{P}}(I_0) & = & \langle \emptyset, \{q(a)\} \rangle & = & I_1, \\ \Phi_{\mathcal{P}}(I_1) & = & \langle \emptyset, \{q(a)\} \rangle & = & I_1. \end{array}$$

I_1 is the least fixed point.

The consequence relation $\models_{wcs}$, which will be used in the following, is defined as follows: Given a program $\mathcal{P}$ and a formula F, $\mathcal{P} \models_{wcs} F$ iff $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(F) = \top$.

Proposition 2.1. *Given a definite $\mathcal{P}$ and an atom A, the following holds:*

$$\mathcal{P} \models_{\mathsf{T}_{\mathcal{P}}} A \qquad \textit{if and only if} \qquad \mathcal{P} \models_{wcs} A.$$

Proof.
According to the definition for the $\mathsf{T}_{\mathcal{P}}$ operator, $\mathcal{P} \models_{\mathsf{T}_{\mathcal{P}}} A$ if and only if at some moment during the fixed point iteration of $\mathsf{T}_{\mathcal{P}}$ there exists a clause $A \leftarrow \mathit{body} \in \mathsf{def}(A, \mathcal{P})$ with $I(\mathit{body}) = \top$. If this is the case, and only then, according to the definition for the $\Phi_{\mathcal{P}}$ operator, $A \in I^{\top}$ at some moment during the fixed point iteration of $\Phi_{\mathcal{P}}$. As $\Phi_{\mathcal{P}}$ is monotonic, A is also true in $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}$. □

The operator defined by Stenning and van Lambalgen [2008] differs in a subtle way from the well-known operator Φ_{F}, introduced by Fitting [1985]. Let I be an interpretation and $\mathcal{P}$ be a program. Then the application of Φ_{F} to I and $\mathcal{P}$, denoted by $\Phi_{\mathsf{F},\mathcal{P}}(I)$, is the interpretation $J = \langle J^{\top}, J^{\bot} \rangle$.

$$\begin{array}{l} J^{\top} = \{A \mid A \leftarrow \mathit{body} \in \mathsf{def}(A, \mathcal{P}) \text{ and } I(\mathit{body}) = \top\} \\ J^{\bot} = \{A \mid \text{for all } A \leftarrow \mathit{body} \in \mathsf{def}(A, \mathcal{P}) \text{ we find that } I(\mathit{body}) = \bot\} \end{array}$$

The definition of $\Phi_{\mathsf{F},\mathcal{P}}$ is like that of $\Phi_{\mathcal{P}}$, except that in the specification of $J^{\bot}$ the first line "$\mathsf{def}(A, \mathcal{P}) \neq \emptyset$ and" is dropped in Φ_{F}. The least fixed point of $\Phi_{\mathsf{F},\mathcal{P}}$ corresponds to the least model of the completion of $\mathcal{P}$under S-semantics, or equivalently under Łukasiewicz semantics. If an atom A is undefined in the program $\mathcal{P}$, then, for arbitrary interpretations I it holds that $A \in J^{\bot}$ in $\Phi_{\mathsf{F},\mathcal{P}}(I) = \langle J^{\top}, J^{\bot} \rangle$, whereas, if $\Phi_{\mathcal{P}}$ is applied instead of $\Phi_{\mathsf{F},\mathcal{P}}$, this does not hold. Example 2.7 discusses the weak completion of a program, the completion of a program and the correspondence to the Fitting operator. In the sequel, as we are mainly interested in the $\Phi_{\mathcal{P}}$ operator.

Summing up, the *Weak Completion Semantics* is the approach to consider weakly completed programs, to compute their least models, and to reason with respect to these models.

Example 2.7. The weak completion of $g\mathcal{P}$ from Example 2.2 consists of the following two equivalences:

$$\begin{array}{lcl} p(a) & \leftrightarrow & q(a) \vee r(a). \\ q(a) & \leftrightarrow & \bot. \end{array}$$

I_1, I_2 and I_3 are models of the weak completion of $\mathcal{P}$.

$$\begin{array}{lcl} I_1 & = & \langle \emptyset, \{q(a)\} \rangle \\ I_2 & = & \langle \emptyset, \{q(a), r(a), p(a)\} \rangle \\ I_3 & = & \langle \{r(a), p(a)\}, \{q(a)\} \rangle \end{array}$$

Its knowledge-least model, I_1, also corresponds to the least fixed point of $\Phi_{\mathcal{P}}$ as shown in Example 2.6. Consider the completion of $g\mathcal{P}$:

$$\begin{array}{lcl} p(a) & \leftrightarrow & q(a) \vee r(a). \\ q(a) & \leftrightarrow & \bot. \\ r(a) & \leftrightarrow & \bot. \end{array}$$

In this case, I_2 is the only interpretation, which is a model of $\mathsf{c}\,\mathcal{P}$. This corresponds to the least fixed point of $\Phi_{\mathsf{F},\mathcal{P}}$, which always computes the least model of $\mathsf{c}\,\mathcal{P}$, if it exists.

2.4. Integrity Constraints

Until now, integrity constraints have not been examined in the context of the Weak Completion Semantics. Yet, they might be useful, and therefore we will explain how we can understand them under three-valued logics and how we will deal with them. Usually, under two-valued semantics a set of *integrity constraints* $\mathcal{IC}$, consists of clauses of the following form:

$$\bot \leftarrow \mathit{body},$$

where *body* is a conjunction of literals. $\mathcal{P}$ *satisfies* $\mathcal{IC}$ if and only if $\mathcal{P} \cup \mathcal{IC}$ is satisfiable. Under two-valued semantics a set of clauses is satisfiable if there exists a two-valued model. This implies that the *body* of each clause in $\mathcal{IC}$ is mapped to false under this model.

Under three-valued semantics, there are different ways on how to understand integrity constraints: Either we require that the *body* of the clause occurring in the set of integrity constraints is false under the model under consideration or that the *body* is unknown. At first glance, it might be natural to assume that the *body* of the $\mathcal{IC}$ should be false. However, considering that we are interested in modeling human reasoning, this understanding of integrity constraints might not deliver the desired result. Assume that we want to formalize the following conditional in a logic program:

If it rains then they will not go to the beach.

The consequence of this conditional is the negation of *they will go to the beach*. Let us assume that *beach* denotes *they will go to the beach* and *rain* denotes *it rains*. As we do not allow negative literals in the head of clauses, we need to introduce an auxiliary atom which represents the negation of the consequence, e.g. $beach'$. The logic program $\mathcal{P}$ representing the conditional consists of the following two clauses:

$$\begin{array}{rcl} beach' & \leftarrow & rain. \\ beach & \leftarrow & \neg beach'. \end{array}$$

The second conditional states that *beach* will be true if $beach'$ is false. If an interpretation $\langle I^\top, I^\bot \rangle$ contains both *beach* and $beach'$ in $I^\top$ it should be invalidated as a model of $\mathcal{P}$ in general. This can be specified by the following integrity constraint:

$$\bot \leftarrow beach \wedge beach',$$

which, given Table 2.1, implies that either $beach'$ or *beach* has to be false. Both cannot stay unknown, even though possibly nothing is stated about the truth of them. As we already have discussed in Section 2.2, we are not interested in finding the truth-least but the knowledge-least model, that is, both $I^\top$ and $I^\bot$ should be minimized, or, in other words, the unknown values should be maximized. Therefore, we understand integrity constraints instead as

$$\mathrm{U} \leftarrow body.$$

As in the following, we only consider integrity constraints under either Łukasiewicz semantics or S-semantics, the body of the integrity constraint can be either false or unknown according to Table 2.1. For the example above we modify the integrity constraint accordingly. The $\mathcal{IC}$ is defined as

$$\mathrm{U} \leftarrow beach \wedge beach'.$$

This understanding that the body can be either false or unknown is similar to the definition of the integrity constraints for the Well-founded Semantics in [Pereira, Aparício, and Alferes, 1991b]. Of course, someone could think of allowing both kinds of $\mathcal{IC}$s, ones with U in the head the others with $\bot$, depending on the representation someone wants to choose for the knowledge under consideration. However, in the sequel, if we consider $\mathcal{IC}$s under three-valued semantics, we will refer to the kind of integrity constraints where only U is allowed in the head of the clause, if not stated otherwise. Note that in the case we will consider integrity constraints under two-valued semantics, they will necessarily have to be understood as $\bot \leftarrow body$, i.e. only $\bot$ is allowed in the head of the clause. In the following, we will consider the $\mathcal{IC}$s satisfying the least model of the weak completion of the given program.

Given an interpretation I and a set of integrity constraints $\mathcal{IC}$, I *satisfies* $\mathcal{IC}$ if and only if all clauses in $\mathcal{IC}$ are true under I (according to either Łukasiewicz semantics or S-semantics being used). Accordingly, we extend the model intersection property for all

models of the weak completion that satisfy $\mathcal{IC}$.

Proposition 2.2. *If there exists a model of the weak completion of program $\mathcal{P}$ that satisfies a set of integrity constraints $\mathcal{IC}$, then there exists a least model of the weak completion of $\mathcal{P}$ that satisfies $\mathcal{IC}$.*

Proof.
This follows immediately from the fact that the model intersection property holds for logic programs under their weak completion under Łukasiewicz semantics. □

2.5. Abduction

We will mainly focus on three-valued abduction and briefly show the correspondence to two-valued abduction. Based on [Kakas, Kowalski, and Toni, 1993], *an abductive framework* is a quadruple $\langle \mathcal{P}, \mathcal{A}, \mathcal{IC}, \models \rangle$, consisting of a program $\mathcal{P}$ as knowledge base, a finite set of abducibles $\mathcal{A}_\mathcal{P}$, a finite set of integrity constraints $\mathcal{IC}$, and a consequence relation $\models$. A *two-valued abductive framework* is the quadruple $\langle \mathcal{P}, \mathcal{A}_{2,\mathcal{P}}, \mathcal{IC}, \models_{\mathsf{T}_\mathcal{P}} \rangle$, where $\mathcal{P}$ is definite, $\models_{\mathsf{T}_\mathcal{P}}$ is the consequence relation with respect $\mathsf{T}_\mathcal{P}$ and $\mathcal{A}_{2,\mathcal{P}}$ is defined as

$$\{A \leftarrow \top \mid A \in \mathsf{undef}(\mathcal{P})\}.$$

Clauses in $\mathcal{IC}$ are of the form $\bot \leftarrow \mathit{body}$. *Observation* $\mathcal{O}$ is a non-empty set of literals.

Definition 2.1. *Let $\langle \mathcal{P}, \mathcal{A}_{2,\mathcal{P}}, \mathcal{IC}, \models_{\mathsf{T}_\mathcal{P}} \rangle$ be a two-valued abductive framework where $\mathcal{P}$ satisfies $\mathcal{IC}$, $\mathcal{E} \subseteq \mathcal{A}_{2,\mathcal{P}}$ and $\mathcal{O}$ is an observation.*

$\mathcal{O}$ is two-valued explained by $\mathcal{E}$ given $\mathcal{P}$ and $\mathcal{IC}$ iff $\mathcal{P} \cup \mathcal{E} \models_{\mathsf{T}_\mathcal{P}} \mathcal{O}$ and $\mathcal{P} \cup \mathcal{E} \models_{\mathsf{T}_\mathcal{P}} \mathcal{IC}$.

$\mathcal{O}$ is two-valued explainable given $\mathcal{P}$ and $\mathcal{IC}$ iff there exists an $\mathcal{E}$ such that $\mathcal{O}$ is two-valued explained by $\mathcal{E}$ given $\mathcal{P}$ and $\mathcal{IC}$.

Normally, only set inclusion minimal (or otherwise preferred) explanations are considered. We assume henceforth that explanations are minimal, that means, there exists no other explanation $\mathcal{E}' \subset \mathcal{E}$ for $\mathcal{O}$. Someone might possibly think of some other preference criterion instead. Note that if $\mathcal{P} \models_{\mathsf{T}_\mathcal{P}} \mathcal{O}$ then $\mathcal{E}$ is empty.

Similarly, for the three-valued semantics considered in this thesis, we define a *three-valued abductive framework* as a quadruple $\langle \mathcal{P}, \mathcal{A}, \mathcal{IC}, \models_{wcs} \rangle$, consisting of a program $\mathcal{P}$ as knowledge base, a set of abducibles $\mathcal{A}$, a set of integrity constraints $\mathcal{IC}$, and the logical consequence relation $\models_{wcs}$. *Observation* $\mathcal{O}$ is a non-empty set of literals. As we are employing the Weak Completion Semantics, abducibles may now not only be facts, but can also take the form of assumptions, otherwise they remain unknown. Therefore,

the set of abducibles $\mathcal{A}_\mathcal{P}$ available for the three-valued abduction is extended with the corresponding assumptions.

$$\mathcal{A}_\mathcal{P} \quad = \quad \{A \leftarrow \top \mid A \in \mathsf{undef}(\mathcal{P})\} \quad \cup \quad \{A \leftarrow \bot \mid A \in \mathsf{undef}(\mathcal{P})\}$$

Proposition 2.3. *Given a definite program $\mathcal{P}$, the following holds:*

$$\textit{If } \{A \leftarrow \top\} \subseteq \mathcal{A}_{2,\mathcal{P}} \qquad \textit{then} \qquad \{A \leftarrow \top, A \leftarrow \bot\} \subseteq \mathcal{A}_\mathcal{P}.$$

Proof.
This follows immediately from the definitions for $\mathcal{A}_{2,\mathcal{P}}$ and $\mathcal{A}_\mathcal{P}$. □

Definition 2.2. *Let $\langle \mathcal{P}, \mathcal{A}_\mathcal{P}, \mathcal{IC}, \models_{wcs} \rangle$ be a three-valued abductive framework where $\mathcal{P}$ satisfies $\mathcal{IC}$, $\mathcal{E} \subseteq \mathcal{A}_\mathcal{P}$ and $\mathcal{O}$ is an observation.*

$\mathcal{O}$ is three-valued explained by $\mathcal{E}$ given $\mathcal{P}$ and $\mathcal{IC}$ iff $\mathcal{P} \cup \mathcal{E} \models_{wcs} \mathcal{O}$ and $\mathcal{P} \cup \mathcal{E} \models_{wcs} \mathcal{IC}$.

$\mathcal{O}$ is three-valued explainable given $\mathcal{P}$ and $\mathcal{IC}$ iff there exists an $\mathcal{E}$ such that $\mathcal{O}$ is three-valued explained by $\mathcal{E}$ given $\mathcal{P}$ and $\mathcal{IC}$.

In abduction, we distinguish between *credulous* and *skeptical reasoning*. Credulous reasoning means that there exists at least one model which entails the observation to be explained. Skeptical reasoning demands that every model of the program entails the observation.

F follows skeptically from $\mathcal{P}$, $\mathcal{IC}$ and $\mathcal{O}$ iff $\mathcal{O}$ can be three-valued explainable given $\mathcal{P}$ and $\mathcal{IC}$, and for all $\mathcal{E}$ for $\mathcal{O}$ it holds that $\mathcal{P} \cup \mathcal{E} \models_{wcs} F$.

F follows credulously from $\mathcal{P}$, $\mathcal{IC}$ and $\mathcal{O}$ iff there exists a $\mathcal{E}$ for $\mathcal{O}$ and it holds that $\mathcal{P} \cup \mathcal{E} \models_{wcs} F$.

Three-valued abduction is illustrated in Example 2.8. $\mathcal{P}, \mathcal{O} \models^s_{wcs} F$ denotes that F follows skeptically from $\mathcal{P}$ and $\mathcal{O}$. $\mathcal{P}, \mathcal{O} \models^c_{wcs} F$ denotes that F follows credulously from $\mathcal{P}$ and $\mathcal{O}$. Note that in the case the abducibles are not abduced as facts or assumptions, they stay unknown in the least model of the weak completion. If we do not want to allow each undefined atom to be an abducible, i.e. if we want to allow for unknown and non-abducible knowledge, we can simply add the clause $A \leftarrow A$ for any such atom A.

Proposition 2.4. *Given a two-valued abductive framework $\langle \mathcal{P}, \mathcal{A}_{2,\mathcal{P}}, \mathcal{IC}, \models_{\mathsf{T}_\mathcal{P}} \rangle$, a three-valued abductive framework $\langle \mathcal{P}, \mathcal{A}_\mathcal{P}, \mathcal{IC}, \models_{wcs} \rangle$, where $\mathcal{P}$ is definite, $\mathcal{E} \subseteq \mathcal{A}_{2,\mathcal{P}}$ and observation $\mathcal{O}$ is a non-empty set of literals. The following holds:*

1. *If $\mathcal{E}$ is a two-valued explanation for $\mathcal{O}$ given $\mathcal{P}$ and $\mathcal{IC}$ then $\mathcal{E}$ is an explanation for $\mathcal{O}$ given $\mathcal{P}$ and $\mathcal{IC}$.*

2. *If $\mathcal{O}$ is two-valued explained given $\mathcal{P}$ and $\mathcal{IC}$ then $\mathcal{O}$ is three-valued explained given $\mathcal{P}$ and $\mathcal{IC}$.*

Proof.
(2) follows from (1), so we show that (1) holds. Let us assume that $\mathcal{E}$ is a two-valued explanation for $\mathcal{O}$ given $\mathcal{P}$ and $\mathcal{IC}$, then $\mathcal{P} \cup \mathcal{E} \models_{\mathsf{T}_\mathcal{P}} \mathcal{O}$ and $\mathcal{P} \cup \mathcal{E} \models_{\mathsf{T}_\mathcal{P}} \mathcal{IC}$. To show: $\mathcal{P} \cup \mathcal{E} \models_{wcs} \mathcal{O}$ and $\mathcal{P} \cup \mathcal{E} \models_{wcs} \mathcal{IC}$.

1. $\mathcal{P} \cup \mathcal{E} \models_{wcs} \mathcal{O}$ follows from $\mathcal{P} \cup \mathcal{E} \models_{\mathsf{T}_\mathcal{P}} \mathcal{O}$ and Proposition 2.1.

2. $\mathcal{P} \cup \mathcal{E} \models_{wcs} \mathcal{IC}$ means that $\mathsf{lm}_2(\mathcal{P} \cup \mathcal{E} \cup \mathcal{IC})$ is satisfiable. This implies that the body of all clauses in $\mathcal{IC}$ is mapped to false in $\mathsf{lm}_2(\mathcal{P} \cup \mathcal{E})$. If they were true in $\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P} \cup \mathcal{E})$, then, according to Proposition 2.1, they would also have to be true in $\mathsf{lm}_2(\mathcal{P} \cup \mathcal{E})$. Therefore, the body of all clauses in $\mathcal{IC}$ is false in $\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P} \cup \mathcal{E})$. Accordingly, $\mathcal{P} \cup \mathcal{E} \models_{wcs} \mathcal{IC}$. □

The other direction does not hold. Consider program $\mathcal{P}$, which consists of one clause:

$$p \leftarrow q.$$

Given $\mathcal{O} = \{\neg p\}$, the only three-valued explanation is $\mathcal{E} = \{q \leftarrow \bot\}$, where $\mathcal{E} \in \mathcal{A}_\mathcal{P}$. However, $\mathcal{E} \notin \mathcal{A}_{2,\mathcal{P}}$ and therefore $\mathcal{E}$ cannot be a two-valued explanation for $\mathcal{O}$.

As in the following we will mainly consider three-valued semantics, we implicitly assume all the abductive frameworks and explanations to be three-valued, if not explicitly stated otherwise.[3] The entailment relations $\models^s_{wcs}$ and $\models^c_{wcs}$ are abbreviations for expressing that a formula follows skeptically or credulously, respectively.

[3]Luís Moniz Pereira observed that, different to the classical $\mathsf{T}_\mathcal{P}$ based definition, under the Weak Completion Semantics we can also allow for both facts and assumptions in two-valued abduction. (personal communication, February 10, 2016)

Example 2.8. Consider program $\mathcal{P}$ consisting of the following three clauses:

$$\begin{array}{rcl} p(X) & \leftarrow & \neg q(X) \wedge r(X) \wedge t(X). \\ p(X) & \leftarrow & \neg s(X) \wedge r(X). \\ t(a) & \leftarrow & \top. \end{array}$$

Assume that $\mathcal{IC} = \emptyset$ and that $\mathcal{O} = \{p(a)\}$. $g\mathcal{P}$ consists of the following three clauses:

$$\begin{array}{rcl} p(a) & \leftarrow & \neg q(a) \wedge r(a) \wedge t(a). \\ p(a) & \leftarrow & \neg s(a) \wedge r(a). \\ t(a) & \leftarrow & \top. \end{array}$$

Let us consider this observation in the three-valued abductive framework $\langle \mathcal{P}, \mathcal{A}_\mathcal{P}, \mathcal{IC}, \models_{wcs} \rangle$, where the set of abducibles $\mathcal{A}_\mathcal{P}$ consists of the following facts and assumptions:

$$\begin{array}{rcl} q(a) & \leftarrow & \top. \\ q(a) & \leftarrow & \bot. \\ \\ r(a) & \leftarrow & \top. \\ r(a) & \leftarrow & \bot. \\ \\ s(a) & \leftarrow & \top. \\ s(a) & \leftarrow & \bot. \end{array}$$

There are two (minimal) explanations for $\mathcal{O}$:

$$\begin{array}{lcll} \mathcal{E}_{rq} & = & \{\ r(a) \leftarrow \top, & q(a) \leftarrow \bot\ \} \quad \text{and} \\ \mathcal{E}_{sr} & = & \{\ s(a) \leftarrow \bot, & r(a) \leftarrow \top\ \}. \end{array}$$

As $r(a)$ follows from all (minimal) explanations, it follows skeptically from $\mathcal{P}$ and $\mathcal{O}$, whereas $\neg q(a)$ and $\neg s(a)$ only follow credulously.

3. Correspondence to Related Semantics

As the Weak Completion Semantics is a novel technique in the field of Computational Logic, we are interested in how it corresponds to other already established non-monotonic approaches. For this purpose we investigate the relation of the Weak Completion with respect to the Completion and the (Partial) Stable Model Semantics in Section 3.2. Section 3.3 shows the main result of this chapter, namely that the well-founded Semantics, a widely accepted approach in the field of non-monotonic reasoning, corresponds to the Weak Completion Semantics for a specific class of modified programs. We finally present in Section 3.4 some initial results of a psychological study, where humans were asked to reason with cyclic conditionals.[1]

3.1. Introduction

As often described in the literature, most logic programming approaches differ in the way they behave with respect to cycles. A program is said to contain a cycle when at least one atom depends on itself, in the following sense: For all clauses of the form $p \leftarrow q_1 \wedge \ldots \wedge q_m \wedge \neg r_1 \wedge \ldots \wedge \neg r_n$ occurring in a program, the head atom p *depends on* all atoms occurring in the body, that is, on $q_1, \ldots, q_m, r_1, \ldots, r_n$. In addition, *depends on* is the least transitive relation that contains p, q_i and p, r_j for all i, $1 \leq i \leq m$, and for all j, $1 \leq j \leq n$. Consider the following two normal logic program examples, adapted from [Przymusinski, 1994]:

$$\mathcal{P}_{\mathit{fly}} \quad = \quad \{\mathit{fly} \leftarrow \mathit{bird} \wedge \neg \mathit{abnormal},\ \mathit{bird}\}$$

and

$$\mathcal{P}_{\mathit{cycle}} \quad = \quad \{\mathit{abnormal} \leftarrow \mathit{irregular},\ \mathit{irregular} \leftarrow \mathit{abnormal}\}.$$

The program $\mathcal{P}_{\mathit{cycle}}$ contains two cycles because *abnormal* and *irregular* depend on themselves. Przymusinski [1994] shows that programs with cycles might have models, which might not seem intuitive. For instance, under Clark's [1978] Completion Semantics we can conclude *fly* from $\mathcal{P}_{\mathit{fly}}$. However, if we extend $\mathcal{P}_{\mathit{fly}}$ with $\mathcal{P}_{\mathit{cycle}}$, we cannot conclude *fly* anymore. This seems to be counterintuitive. Moreover, under the Completion Semantics

[1] The original idea for this chapter has been published in [Dietz and Hölldobler, 2012]. The main results of Section 3.2 and Section 3.3 have been published in [Dietz, Hölldobler, and Wernhard, 2014]. Section 3.4 has been published in [Dietz, Hölldobler, and Ragni, 2013].

as well as under the Stable Model Semantics [Gelfond and Lifschitz, 1988], cycles established through an odd number of negated atom occurrences can lead to inconsistency, that is, to programs which do not have a model: A program containing a clause $p \leftarrow \neg p$ does not have a two-valued stable model and the completion of this clause, $p \leftrightarrow \neg p$, is inconsistent.

A solution to these problems is to consider three-valued interpretations instead of two-valued interpretations. Przymusinski [1990] proposed the three-valued Stable Model Semantics, also known as the Partial Stable Model Semantics, a conservative extension of the Stable Model Semantics, which preserves stable models. Under the Partial Stable Model Semantics the program $\{p \leftarrow \neg p\}$ has a unique three-valued model in which p is *unknown*. If we extend $\mathcal{P}_{fly}$ with

$$\mathcal{P}_{neg\text{-}cycle} = \{abnormal \leftarrow \neg regular,\ regular \leftarrow \neg abnormal\}$$

we do not obtain just one unique three-valued stable model but three three-valued stable models: One model where *fly*, *bird* and *regular* are true whereas *abnormal* is false, another one where *bird* and *abnormal* are true whereas *fly* and *regular* are false, and finally one where *bird* is true and all other atoms are unknown. Here, the challenge is to find the model that corresponds most likely to the model a human would generate in a certain commonsense setting, rather than the perfect model in a purely logical context.

The Well-founded Semantics introduced by Van Gelder, Ross, and Schlipf [1991] is a widely accepted approach in the field of non-monotonic reasoning and coincides with the least three-valued stable model [Przymusinski, 1990]. Compared to Clark's (two-valued) Completion or the (two-valued) Stable Model Semantics, the Well-founded Semantics is considered to be more accurate for programs with positive or negative cycles [Przymusinski, 1994]. For instance, in the well-founded model of $\mathcal{P}_{neg\text{-}cycle}$, *abnormal* and *regular* are unknown and in the well-founded model of $\mathcal{P}_{cycle}$, *abnormal* and *irregular* are false. Under the Completion Semantics there is not even a model for $\mathcal{P}_{neg\text{-}cycle}$. In general, under the Well-founded Semantics, atoms involved in just positive cycles are false whereas atoms involved in just negative cycles are unknown. The intuition behind this distinction is that the negation of *abnormal* or *irregular* shall not support the truth of any other atom in the program. For instance let us consider $\mathcal{P}_{fly} \cup \mathcal{P}_{neg\text{-}cycle}$, where if *regular* would be false in the well-founded model, then necessarily *abnormal* would have to be true. But then the negation of *abnormal* would provide misleading support for further positive conclusions.

As the Well-founded Semantics is a well-established approach in the literature, we want to investigate how it relates to the Weak Completion Semantics. What are the similarities and where do they differ? Can both approaches adequately represent human reasoning episodes such as the suppression task?

3.2. Related Semantics

In order to show the correspondence between the Weak Completion and the Well-founded Semantics, we will now review the latter and the related Partial Stable Model Semantics. We proceed by giving definitions of certain relevant program classes, which constrain the allowed possibilities of circular dependency in programs. On this basis, we then develop three-valued generalizations of the concepts of supported and well-supported models, which have been originally specified just for two-valued semantics.

As has been discussed in Chapter 2.2, Łukasiewicz semantics and S-semantics lead to the same model relationship for the relevant classes of formulas. In the following, unless specified otherwise, we implicitly consider them as underlying semantics. For simplicity, in this chapter programs are assumed to be propositional.

3.2.1. Stable Model Semantics and Well-Founded Semantics

Stable models have been originally defined by Gelfond and Lifschitz [1988] in terms of a program transformation that is often called *Gelfond-Lifschitz transformation*. Their approach has been extended by Przymusinski [1990] to the Partial Stable Model Semantics in order to show the relationship to the Well-founded Semantics. Intuitively, a general difference between the Weak Completion Semantics and the ones which will be presented here, is how negation is understood in the body of a clause: Under the (Partial) Stable Model Semantics and the Well-founded Semantics, the closed-world assumption is assumed, and therefore negation is assumed by default, the set of atoms that is false, is tried to be maximized. On the other hand under the Weak Completion Semantics, the open-world assumption is assumed and therefore negation is not assumed by default, the set of atoms that is unknown, is tried to be maximized instead.

The *reduct of a normal program* $\mathcal{P}$ with respect to an interpretation I, denoted by $\mathcal{P}|_I$, is obtained from $\mathcal{P}$ by replacing in the bodies of all clauses $\mathcal{P}$ each negative literal $\neg A$ by $I(\neg A)$, that is, with the truth value constant corresponding to the value of $\neg A$ under I. Note that a reduct is still a set of clauses, although, because truth value constants may now occur in bodies, it is possibly not a program according to our specification in Chapter 2. However in this chapter, for the sake of simplicity, when we consider models with respect to reducts of programs, we assume that $\top$, U and $\bot$ are atoms. An interpretation I is a *three-valued stable model* of $\mathcal{P}$ if and only if I is a truth-minimal model of $\mathcal{P}|_I$. Example 3.1 shows a programs reduct and its corresponding stable models. In the sequel, when we discuss interpretations and models, we mean three-valued interpretations and three-valued models, except if explicitly stated otherwise.

By analogy to the well-known $T_\mathcal{P}$ operator for two-valued interpretations [Van Emden and Kowalski, 1976], Przymusinski [1990] introduced an operator $\Psi_\mathcal{P}$ for three-valued interpretations: Suppose that $\mathcal{P}$ is a normal logic program and I is an interpretation of $\mathcal{P}$: Define $\Psi_\mathcal{P}(I) = \langle J^\top, J^\bot \rangle$ to be the interpretation given by

Example 3.1. Consider the program $\mathcal{P}_1$ consisting of the following clause:

$$p \leftarrow q.$$

where $\mathcal{A}\mathsf{t} = \{p, q\}$. As $\mathcal{P}_1$ does not contain an occurrence of a negative literal in the body of a clause, we get the reduct $\mathcal{P}_1|_I = \mathcal{P}_1$ for any interpretation I. We obtain six different models of $\mathcal{P}_1$.

$$\begin{array}{lllllllll} I_1 & = & \langle \emptyset, \{p,q\} \rangle & & I_2 & = & \langle \{p,q\}, \emptyset \rangle & & I_3 & = & \langle \emptyset, \emptyset \rangle \\ I_4 & = & \langle \{p\}, \{q\} \rangle & & I_5 & = & \langle \{p\}, \emptyset \rangle & & I_6 & = & \langle \emptyset, \{q\} \rangle \end{array}$$

The only stable model is I_1 because $I_1 \preceq_t I_j$ for all $j \in [2, 6]$.
Let program $\mathcal{P}_2$ consist of the following two clauses:

$$\begin{array}{lll} p & \leftarrow & \neg q. \\ q & \leftarrow & \neg p. \end{array}$$

$\mathcal{P}_2$ has three different models.

$$I_1 = \langle \{p\}, \{q\} \rangle, \qquad I_2 = \langle \{q\}, \{p\} \rangle \qquad \text{and} \qquad I_3 = \langle \emptyset, \emptyset \rangle.$$

We obtain three reducts of $\mathcal{P}_2$ for each of these interpretations.

$$\begin{array}{lll} \mathcal{P}_2|_{I_1} & = & \{p \leftarrow \top,\ q \leftarrow \bot\} \\ \mathcal{P}_2|_{I_2} & = & \{p \leftarrow \bot,\ q \leftarrow \top\} \\ \mathcal{P}_2|_{I_3} & = & \{p \leftarrow \mathrm{U},\ q \leftarrow \mathrm{U}\} \end{array}$$

All three interpretations I_1, I_2 and I_3 are truth-minimal models of the corresponding reducts and, hence, they are stable models of $\mathcal{P}_2$. It is easy to see that they are the only stable models of $\mathcal{P}_2$.
As $I_3 \preceq_k I_1$ and $I_3 \preceq_k I_2$, I_3 is the knowledge-least stable model of $\mathcal{P}$.

(*i*) $A \in J^\top$ if there exists a clause $A \leftarrow \mathit{body} \in \mathcal{P}$ such that $I(\mathit{body}) = \top$,

(*ii*) $A \notin (J^\top \cup J^\bot)$ if $A \notin J^\top$ and if there exists a clause $A \leftarrow \mathit{body} \in \mathcal{P}$ such that $I(\mathit{body}) = \mathrm{U}$,

(*iii*) $A \in J^\bot$, otherwise.

This operator can be applied to the sets of implications obtained as reduct $\mathcal{P}|_I$. As shown by Przymusinski, the least fixed point of $\Psi_{\mathcal{P}|_I}$ is the truth-least model of $\mathcal{P}|_I$.

It has been further shown by Przymusinski that each normal program has a knowledge-least stable model, which coincides with the *well-founded model*.

The Well-founded Semantics has been defined as follows [Van Gelder, Ross, and Schlipf, 1991]: A set of atoms $U \subseteq \mathsf{atoms}(\mathcal{P})$ is said to be an *unfounded set of* $\mathcal{P}$ with respect

to interpretation I if each atom $A \in U$ satisfies the following condition:
For each clause $A \leftarrow \mathit{body} \in \mathcal{P}$, at least one of the following holds:

1. $I(\mathit{body}) = \bot$.
2. There exists a literal L in $\mathsf{pos}(\mathit{body})$ with $L \in U$.

Given I and $\mathcal{P}$, the transformations $T_\mathcal{P}$, $\mathcal{U}_\mathcal{P}$, and $W_\mathcal{P}$ are defined as follows:

$T_\mathcal{P}(I) = \{A \mid$ there exists a clause $A \leftarrow \mathit{body} \in \mathcal{P}$ with $I(\mathit{body}) = \top\}$,

$\mathcal{U}_\mathcal{P}(I)$ is the *greatest unfounded set* of $\mathcal{P}$ with respect to I, and

$W_\mathcal{P}(I) = \langle T_\mathcal{P}(I), \mathcal{U}_\mathcal{P}(I)\rangle$,

where the greatest unfounded set $\mathcal{U}_\mathcal{P}(I)$ of $\mathcal{P}$ with respect to I is the union of all unfounded sets of $\mathcal{P}$ with respect to I.[2]

$T_\mathcal{P}$, $\mathcal{U}_\mathcal{P}$ and $W_\mathcal{P}$ are monotonic transformations. The least fixed point of $W_\mathcal{P}(I)$ can be recursively defined as follows: Let α range over all countable ordinals. The interpretations I_α and I^∞ are defined recursively by starting with $I_0 = \langle\emptyset, \emptyset\rangle$:

1. For limit ordinal α, $I_\alpha = \bigcup_{\beta < \alpha} I_\beta$.
2. For successor ordinal $\alpha = \gamma + 1$, $I_{\gamma+1} = W_\mathcal{P}(I_\gamma)$.
3. Finally, define $I^\infty = \bigcup_\alpha I_\alpha$.

I_α is the least fixed point of $W_\mathcal{P}$ where $I_\alpha = W_\mathcal{P}(I_\alpha)$. The least fixed point of $W_\mathcal{P}(I)$ is the *well-founded model* of $\mathcal{P}$ ($\mathsf{wfm}\ \mathcal{P}$). Example 3.2 demonstrates the least fixed point computation of $W_\mathcal{P}$ given a simple program. A constructive definition of the Well-founded Semantics can be found in [Van Gelder, 1989].

3.2.2. Program Classes and Cycles

Let $\mathcal{P}$ be a program and $A, B \in \mathsf{atoms}(\mathcal{P})$. *$A$ depends negatively on B* if and only if $\mathcal{P}$ contains a clause of the form $A \leftarrow \mathit{body}$ and $\neg B$ appears in $\mathsf{neg}(\mathit{body})$. *$A$ depends positively on B* if and only if A does not depend negatively on B and $\mathcal{P}$ contains a clause of the form $A \leftarrow \mathit{body}$ and B appears in $\mathsf{pos}(\mathit{body})$. It is easy to see that A depends on B if and only if A depends positively or negatively on B. As dependency is transitive, if A depends on B and B depends on C, then A depends on C, where one negative dependency is enough to define the whole dependency as negative. Consider Example 3.3 for clarification. Different program classes with respect to the occurrence of cycles are often defined through level mapping characterizations. A *level mapping* for a program $\mathcal{P}$ is a function ℓ which assigns to each atom a natural number. It is extended

[2] In [Van Gelder, Ross, and Schlipf, 1991], three-valued interpretations are defined as sets of literals: $W_\mathcal{P}(I)$ was originally defined as $W_\mathcal{P}(I) = T_\mathcal{P}(I) \cup \neg\, \mathcal{U}_\mathcal{P}(I)$, where $\neg\, \mathcal{U} = \{\neg A \mid A \in \mathcal{U}\}$.

Example 3.2. Consider program $\mathcal{P}$ consisting of the following three clauses:

$$\begin{array}{lcl} p & \leftarrow & q. \\ t & \leftarrow & \neg s. \\ s & \leftarrow & \top. \end{array}$$

The greatest unfounded set $\mathcal{U}$ of $\mathcal{P}$ with respect to $I_0 = \langle \emptyset, \emptyset \rangle$ is

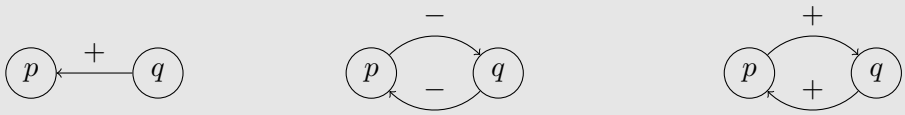
$$\{q, p\}.$$

Let us compute the fixed point of $W_\mathcal{P} = \langle \mathsf{T}_\mathcal{P}(I_0), \mathcal{U}_\mathcal{P}(I_0) \rangle$:

$$\begin{array}{lcllcll} W_\mathcal{P}(I_0) & = & \langle \{s\}, \{q, p\} \rangle & = & I_1, & & \\ W_\mathcal{P}(I_1) & = & \langle \{s\}, \{q, p, t\} \rangle & = & I_2 & = & W_\mathcal{P}(I_2). \end{array}$$

I_2 is the least fixed point and therefore I_2 is the well-founded model of $\mathcal{P}$.

Example 3.3. Consider the three programs below and their representations as graphs, where the nodes represent the atoms and the arcs represent the dependencies: An arc labeled "+" represents a positive dependency and an arc labeled "−" represents a negative dependency.



$\mathcal{P}_1 = \{p \leftarrow q\}$ $\mathcal{P}_2 = \{p \leftarrow \neg q,\ p \leftarrow \neg q\}$ $\mathcal{P}_3 = \{p \leftarrow q,\ q \leftarrow p\}$

A program $\mathcal{P}$ contains a *cycle* if at least one atom occurring in $\mathcal{P}$ depends on itself. The programs $\mathcal{P}_2$ and $\mathcal{P}_3$ contain cycles, whereas $\mathcal{P}_1$ does not.

to literals as follows, where L is a literal and A a atom: $\ell(\neg A) = \ell(A)$. Additionally, ℓ is extended to the truth-value constants $\top$ and $\bot$, where $\ell(\top) = \ell(\bot) = 0$. In the following, ℓ will refer to $\ell^\star$. A program $\mathcal{P}$ is *acyclic with respect to a level mapping* ℓ if and only if for every clause $A \leftarrow \mathit{body} \in \mathcal{P}$ and for all literals L in body we find that $\ell(A) > \ell(L)$. A program $\mathcal{P}$ is *acyclic* if and only if it is acyclic with respect to some level mapping. Consider again $\mathcal{P}_1$ in Example 3.3. With $\ell(p) = 2$ and $\ell(q) = 1$ we find that $\mathcal{P}_1$ is acyclic, whereas $\mathcal{P}_2$ and $\mathcal{P}_3$ are not acyclic.

Stratified programs have been investigated by Przymusinski [1988] and by Apt, Blair, and Walker [1988]. A level mapping characterization of this class of programs is given by Hitzler and Wendt [2005]: A program $\mathcal{P}$ is *stratified with respect to a level mapping* ℓ if and only if for every clause $A \leftarrow \mathit{body} \in \mathcal{P}$ we find that $\ell(A) \geq \ell(L)$ for all literals L in $\mathsf{pos}(\mathit{body})$, and $\ell(A) > \ell(L)$ for all literals L in $\mathsf{neg}(\mathit{body})$. A program $\mathcal{P}$ is *stratified* if and only if it is stratified with respect to some level mapping. Programs which only contain positive cycles are stratified. In Example 3.3, $\mathcal{P}_1$ and $\mathcal{P}_3$ are stratified, but $\mathcal{P}_2$ is not.

Fages [1994] introduced the term *positive-order-consistent* to define programs that do not contain positive cycles. Nowadays, the term *tight* is often used to describe this property [Erdem and Lifschitz, 2003]. A level mapping characterization for this class of programs is defined as follows: A program $\mathcal{P}$ is *tight with respect to a level mapping* ℓ if and only if for every clause $A \leftarrow \mathit{body} \in \mathcal{P}$ we find that $\ell(A) > \ell(L)$ for all literals L in $\mathsf{pos}(\mathit{body})$. A program $\mathcal{P}$ is *tight* if and only if it is tight with respect to some level mapping. Programs which only contain negative cycles are tight programs. In Example 3.3, $\mathcal{P}_1$ and $\mathcal{P}_2$ are tight, but $\mathcal{P}_3$ is not. Under two-valued semantics, *negative odd cycles* lead to inconsistency as shown by Example 3.4. Under the Partial Stable Model Semantics atoms stay unknown when they are involved in negative cycles. Table 3.1 shows the stable models, the models of the completion and the models of the weak completion from the programs discussed in Example 3.3.

3.2.3. Supported Models and Well-Supported Models

In two-valued logic, the notion of supported models provides an alternate characterization for the models of Clark's completion [Apt, Blair, and Walker, 1988]. We adapt this characterization for three-valued logics.

An interpretation I is *supported* with respect to a set of clauses $\mathcal{P}$ if and only if for all atoms A with $I(A) = \top$ there exists a clause $A \leftarrow \mathit{body} \in \mathcal{P}$ such that $I(\mathit{body}) = \top$ and for all atoms A with $I(A) = \mathrm{U}$ there is no clause $A \leftarrow \mathit{body} \in \mathcal{P}$ such that $I(\mathit{body}) = \top$, but there exists a clause $A \leftarrow \mathit{body} \in \mathcal{P}$ such that $I(\mathit{body}) = \mathrm{U}$. Accordingly, as a result, $I(A) = \bot$ iff for all $A \leftarrow \mathit{body} \in \mathsf{def}(A, \mathcal{P})$, $I(\mathit{body}) = \bot$. We say that I is a *supported models* of $\mathcal{P}$ if and only if I is a model of $\mathcal{P}$ and is supported with respect to $\mathcal{P}$. Analogously to the two-valued case, completion and supported models coincide for three-valued logics:

Example 3.4. Consider program $\mathcal{P}$ consisting the following three clauses:

$$\begin{array}{lll} p & \leftarrow & \neg q. \\ q & \leftarrow & \neg r. \\ r & \leftarrow & \neg p. \end{array}$$

There exists no two-valued stable model of $\mathcal{P}$. If we assume q to be false, then p and r have to be true. However because we have that $r \leftarrow \neg p$, r has to be false as well, but then q has to be true, which makes p false again. The completion of $\mathcal{P}$, $\mathsf{c}\,\mathcal{P}$, consists of the following three equivalences:

$$\begin{array}{lll} p & \leftrightarrow & \neg q. \\ q & \leftrightarrow & \neg r. \\ r & \leftrightarrow & \neg p. \end{array}$$

$\mathcal{P}$ does not have a two-valued model.

Lemma 3.1. *For any program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is a model of the completion of $\mathcal{P}$.*
2. *I is a supported model of $\mathcal{P}$.*

Proof
(1) $\rightarrow$ (2): Assume that I is a model of the completion of $\mathcal{P}$. Let A be an atom.

- If $A \in I^{\top}$, then there exists an equivalence $A \leftrightarrow \mathit{body}_1 \vee \mathit{body}_2 \vee \ldots \mathit{body}_n$ in $\mathsf{c}\,\mathcal{P}$, where at least for one body_i, $1 \leq i \leq n$, it holds that $\mathit{body}_i \in I^{\top}$. Therefore, there has to be a clause $A \leftarrow \mathit{body}_i \in \mathcal{P}$ such that $I(\mathit{body}_i) = \top$, and thus I is supported with respect to this clause.
- If $A \notin (I^{\top} \cup I^{\bot})$, then there exists an equivalence $A \leftrightarrow \mathit{body}_1 \vee \mathit{body}_2 \vee \cdots \vee \mathit{body}_n$ in $\mathsf{c}\,\mathcal{P}$, $1 \leq i \leq n$, where for all $\mathit{body}_i \notin I^{\top}$ and for at least one $\mathit{body}_i \notin I^{\bot}$.Therefore, there has to be a clause $A \leftarrow \mathit{body}_i \in \mathcal{P}$ such that $I(\mathit{body}_i) = \mathrm{U}$, and thus I is supported with respect to this clause.

(2) $\rightarrow$ (1): Assume that I is a supported model of $\mathcal{P}$. Let A be an atom.

- If $A \in I^{\top}$, then there exists a clause that supports I, i.e. $A \leftarrow \mathit{body} \in \mathcal{P}$ such that $I(\mathit{body}) = \top$. Accordingly, A is also true in the model of the completion of $\mathcal{P}$.
- If $A \notin (I^{\top} \cup I^{\bot})$, then there is no clause $A \leftarrow \mathit{body} \in \mathcal{P}$ such that $I(\mathit{body}) = \top$ but there exists a clause that supports I, i.e. $A \leftarrow \mathit{body} \in \mathcal{P}$ such that $I(\mathit{body}) = \mathrm{U}$. Accordingly, A is also unknown in the model of the completion of $\mathcal{P}$. □

In order to deal with positive cycles, in some approaches cyclic support for atoms is eliminated: Their truth value is either left unknown or mapped to false. For two-valued

Program $\mathcal{P}$	Stable Models of $\mathcal{P}$	Models of $\mathsf{c}\,\mathcal{P}$	Models of $\mathsf{wc}\,\mathcal{P}$
$\mathcal{P}_1 = \{p \leftarrow q\}$	$\langle \emptyset, \{p,q\}\rangle$	$\langle \emptyset, \{p,q\}\rangle$	$\langle \emptyset, \{p,q\}\rangle$ $\langle \emptyset, \emptyset\rangle$ $\langle \{p,q\}, \emptyset\rangle$
$\mathcal{P}_2 = \{p \leftarrow \neg q,\ q \leftarrow \neg p\}$	$\langle \{p\}, \{q\}\rangle$ $\langle \emptyset, \emptyset\rangle$ $\langle \{q\}, \{p\}\rangle$	$\langle \{p\}, \{q\}\rangle$ $\langle \emptyset, \emptyset\rangle$ $\langle \{q\}, \{p\}\rangle$	$\langle \{p\}, \{q\}\rangle$ $\langle \emptyset, \emptyset\rangle$ $\langle \{q\}, \{p\}\rangle$
$\mathcal{P}_3 = \{p \leftarrow q,\ q \leftarrow p\}$	$\langle \emptyset, \{p,q\}\rangle$	$\langle \emptyset, \{p,q\}\rangle$ $\langle \emptyset, \emptyset\rangle$ $\langle \{p,q\}, \emptyset\rangle$	$\langle \emptyset, \{p,q\}\rangle$ $\langle \emptyset, \emptyset\rangle$ $\langle \{p,q\}, \emptyset\rangle$

Table 3.1.: Program examples and the corresponding stable models, models of the completion and models of the weak completion, under the assumption that $\mathcal{A}\mathsf{t} = \mathsf{atoms}(\mathcal{P}) = \{p, q\}$.

logics, this is captured for example by the notions of grounded models [Elkan, 1990] and well-supported models [Fages, 1991], that is, models which are supported and assign $\top$ only to atoms that are not involved in positive cycles. Well-supported models in this sense are exactly the (two-valued) stable models. We now extend this concept to three-valued logics:

An interpretation I is *well-supported* with respect to a level mapping ℓ and a finite set of clauses $\mathcal{P}$ if and only if for all atoms A with $I(A) \neq \bot$ there exists a clause $A \leftarrow \mathit{body} \in \mathcal{P}$ such that:

1. if $A = \top$, then $I(\mathit{body}) = \top$ and for all literals L in $\mathsf{pos}(\mathit{body})$ it holds that $\ell(L) < \ell(A)$, else,
2. if $A = \mathrm{U}$ then $I(\mathit{body}) = \mathrm{U}$.

We call a clause $A \leftarrow \mathit{body}$ that meets the requirement of the definition for well-supported interpretations, a *supporting justification of* A. We say that I is a *well-supported model* of $\mathcal{P}$ if and only if I is a model of $\mathcal{P}$ and is well-supported with respect to $\mathcal{P}$ and some level mapping. The following lemma follows immediately from the definitions of supported and well-supported models:

Lemma 3.2. *Well-supported models of a program $\mathcal{P}$ are supported models of $\mathcal{P}$.*

For two-valued logics, the correspondence between stable models and well-supported models has been developed by Elkan [1990] in a stepwise way. We now adapt these steps to our three-valued setting in the following Lemmas 3.3–3.7. As in the case of completion and supported models, these propositions apply to both Łukasiewicz semantics and S-semantics.

Lemma 3.3. *Any model I of a normal program $\mathcal{P}$ is also a model of $\mathcal{P}|_I$.*

Proof.
Immediate from the definition of $\mathcal{P}|_I$: We obtain $\mathcal{P}|_I$ from $\mathcal{P}$ by replacing the body of clauses with truth value constants corresponding to their value under I. □

Lemma 3.4. *Any well-supported model I of a normal program $\mathcal{P}$ is also a well-supported model of $\mathcal{P}|_I$.*

Proof.
The transformation of $\mathcal{P}$ given an interpretation I to a reduct of $\mathcal{P}|_I$ does not change the semantics of a program with respect to I. On the contrary, the semantics is included into the program by replacing the bodies of the clauses with their truth values with respect to I. Therefore, a well-supported model I of $\mathcal{P}$ is a well-supported model of $\mathcal{P}|_I$. Let $A \leftarrow \mathit{body}$ be a supporting justification of A in $\mathcal{P}$ with respect to I and a level mapping ℓ such that $\ell(A) < \ell(L)$ for each $L \in \mathsf{pos}(\mathit{body})$. Given that $\mathsf{neg}(\mathit{body}) = \neg B_1 \wedge \ldots \wedge \neg B_n$, for $n \geq 0$, let $\mathit{body}' = \mathsf{pos}(\mathit{body}) \wedge I(\neg B_1) \wedge \ldots \wedge I(\neg B_n)$. The clause $A \leftarrow \mathit{body}'$ is then a supporting justification in $\mathcal{P}|_I$: It is an element of $\mathcal{P}|_I$, it holds that $\ell(A) < \ell(L)$ for each L in $\mathsf{pos}(\mathit{body}') = \mathsf{pos}(\mathit{body})$ and the semantic requirements are met since $I(\mathit{body}') = I(\mathit{body})$. □

Lemma 3.5. *Well-supported models of program are truth-minimal.*

Proof.
We show the lemma by contradiction: Let $\mathcal{P}$ be a program and let I be a well-supported model of $\mathcal{P}$ with respect to a level mapping ℓ. Assume that I is not truth-minimal, i.e. there exists a model J of $\mathcal{P}$ such that $J \preceq_t I$ and $J \neq I$. Let $I^U = \mathcal{A}\mathsf{t} \setminus (I^\top \cup I^\bot)$, i.e. the set of atoms mapped to U by I, and analogously let $J^U = \mathcal{A}\mathsf{t} \setminus (J^\top \cup J^\bot)$. The condition $J \preceq_t I$ and $J \neq I$ is equivalent to $(J^\top \cup J^U) \subset (I^\top \cup I^U)$. Then, we know that the set of atoms $\Delta = (I^\top \cup I^U) \setminus (J^\top \cup J^U)$ is non-empty, and that for all $A \in \Delta$ it holds that $J(A) = \bot$ and $I(A) \neq \bot$.

Now, let $A \in \Delta$ be an atom such that $\ell(A)$ is minimal among Δ, i.e. the value $\ell(A)$ is least among the values of ℓ of the elements in Δ. Let $A \leftarrow \mathit{body} \in \mathcal{P}$ be a supporting justification of A with respect to I. Then it holds that $I(\mathit{body}) \neq \bot$ and that $\ell(L) < \ell(A)$ for each literal L in $\mathsf{pos}(\mathit{body})$. Because J is a model of $\mathcal{P}$ and we have $J(A) = \bot$, it follows that $J(\mathit{body}) = \bot$. Thus, there must be a literal L in body such that $J(L) = \bot$ and $I(L) \neq \bot$. Consider the following two cases:

1. In the case that L is a negative literal $\neg B$ it must hold that $J(B) = \top$. As I is not truth minimal, i.e. there exists an interpretation J such that $J \preceq_t I$, accordingly, $J^\top \subseteq I^\top$. It follows that $I(B) = \top$ and thus $I(L) = \bot$, in contradiction to $I(L) \neq \bot$.

2. In the case that L is a positive literal, we know that $J(L) = \bot$, which implies that $L \notin (J^\top \cup J^U)$ and we know that $I(L) \neq \bot$, which implies that $L \in (I^\top \cup I^U)$. Therefore $L \in \Delta$. But then $\ell(L) < \ell(A)$ contradicts the fact that $l(A)$ is a least level mapping value among all elements in Δ. □

Lemma 3.6. *For any normal program $\mathcal{P}$ and interpretation I, the truth-minimal model of $\mathcal{P}|_I$ is well-supported.*

Proof.
This follows from the fixed point construction of the truth-minimal model of $\mathcal{P}|_I$ by the operator Ψ introduced in [Przymusinski, 1990] (see Section 3.2.1). Well-supportedness is assured by any level mapping, where an atom is assigned level i if its value is determined in the ith iteration of the application of Ψ. □

Lemma 3.7. *For any normal program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is a stable model of $\mathcal{P}$.*
2. *I is a well-supported model of $\mathcal{P}$.*

Proof.
(1) → (2): Let I be a stable model of $\mathcal{P}$, i.e. I is a model of $\mathcal{P}$ and it is a truth-minimal model of $\mathcal{P}|_I$. By Lemma 3.6, there exists a level mapping ℓ such that I is well-supported with respect to $\mathcal{P}|_I$. Let $A \leftarrow \mathit{body}$ be a justification of atom A with respect to $\mathcal{P}|_I$. It then holds that $I(A) = I(\mathit{body}) \neq \bot$ and $\ell(A) > \ell(L)$ for all literals L in body. In $\mathcal{P}$ there must be a clause $A \leftarrow \mathit{body}'$ from which $A \leftarrow \mathit{body}$ has been obtained in forming the reduct. From the construction of $\mathcal{P}|_I$ it follows that $\mathsf{pos}(\mathit{body}') = \mathsf{pos}(\mathit{body})$ and that I is a model of $\mathsf{neg}(\mathit{body}')$ and thus $A \leftarrow \mathit{body}'$ is a justification of A with respect to $\mathcal{P}$.
(2) → (1): By Lemma 3.4 and 3.5 any well-supported model I of $\mathcal{P}$ is a truth-minimal model of $\mathcal{P}|_I$, and thus a stable model of $\mathcal{P}$. □

Lemma 3.8. *Any stable model I of $\mathcal{P}$ is a model of the completion of $\mathcal{P}$.*

Proof.
By Lemma 3.7, any stable model I of $\mathcal{P}$ is a well-supported model of $\mathcal{P}$. Accordingly, by Lemma 3.2, I is a supported model of $\mathcal{P}$. Finally, by Lemma 3.1, I is a model of the completion of $\mathcal{P}$. □

The overview in Table A.1 in Appendix A summarizes the correspondences between several two- and three-valued semantics, including results reported in the literature so far. For instance, Fages [1994] showed that for tight logic programs under two-valued semantics, the stable models coincide with the models of the completion. Przymusinski [1990] showed that the knowledge-least stable model coincides with the well-founded model.

3.3. Correspondence

With Theorem 3.9 below we now show that the knowledge-least model of the weak completion is identical to the well-founded model of the program, after a transformation that essentially effects simulation of the treatment of undefined atoms under the weak

completion. This transformation is specified as follows: We assume that $\mathcal{A}\mathsf{t}$ is the union of disjoint sets $\mathcal{A}\mathsf{t}'$ and $\mathsf{auxatoms} = \{A' \mid A \in \mathcal{A}\mathsf{t}'\}$. For program $\mathcal{P}$ we define

$$\mathcal{P}^{\mathsf{mod}} \;=\; \mathcal{P}^{+} \cup \bigcup_{A \in \mathsf{undef}(\mathcal{P})} \{A \leftarrow \neg A',\; A' \leftarrow \neg A\}.$$

We assume that atoms in $\mathsf{auxatoms}$ only occur in programs $\mathcal{P}^{\mathsf{mod}}$ resulting from the indicated transformation. Again, also in this section our considerations apply to both Łukasiewicz semantics and S-semantics. The correspondence of the Weak Completion Semantics and the Well-founded Semantics can now be stated as follows:

Theorem 3.9. *For any tight program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is the knowledge-least model of the weak completion of $\mathcal{P}$.*
2. *I is the well-founded model of $\mathcal{P}^{\mathsf{mod}}$.*

In the rest of this section we develop the proof of Theorem 3.9, which involves further auxiliary definitions and some intermediate results, in particular about the correspondence between the Completion Semantics and the Stable Model Semantics.[3] We first note some properties of $\mathcal{P}^{\mathsf{mod}}$, which follow easily from its definition:

Proposition 3.10. *Given a program $\mathcal{P}$, the following holds:*

(i) *If $\mathcal{P}$ is tight, then $\mathcal{P}^{\mathsf{mod}}$ is also tight.*

(ii) *$\mathcal{P}^{\mathsf{mod}}$ is a normal program.*

If we consider not just knowledge-least models, we have to map between interpretations that assign to the members of $\mathsf{auxatoms}$ values as required by $\mathcal{P}^{\mathsf{mod}}$ and interpretations where the value of members of $\mathsf{auxatoms}$ is always unknown. To this end, we define the two conversions for interpretations I and sets of atoms S. First, I_S^{mod} is the interpretation $\langle J^{\top}, J^{\bot}\rangle$ where $J^{\top}$ ($J^{\bot}$) contains $I^{\top}$ ($I^{\bot}$) together with the auxiliary atoms A' if $A \in S$ and $A \in I^{\bot}$ ($A \in I^{\top}$):

$$J^{\top} \;=\; I^{\top} \cup \{A' \mid A \in S \cap I^{\bot}\} \qquad \text{and} \qquad J^{\bot} \;=\; I^{\bot} \cup \{A' \mid A \in S \cap I^{\top}\}.$$

Second, I^{invmod} is the interpretation $\langle K^{\top}, K^{\bot}\rangle$ where $K^{\top}$ ($K^{\bot}$) contains all atoms which are in $I^{\top}$ ($I^{\bot}$) but not in $\mathsf{auxatoms}$:

$$K^{\top} \;=\; I^{\top} \setminus \mathsf{auxatoms} \qquad \text{and} \qquad K^{\bot} \;=\; I^{\bot} \setminus \mathsf{auxatoms}.$$

[3] Pereira, Aparício, and Alferes [1991a] showed the correspondence between the contradiction free extended Stable Model Semantics and the extended Stable Model Semantics, an extension of the Well-founded Semantics by introducing a similar transformation as for $\mathcal{P}^{\mathsf{mod}}$ where the transformed program is extended with the following clauses: $A \leftarrow \neg A'$, $A' \leftarrow \neg A$ and $A' \leftarrow \neg A'$. A further early documented use of $A \leftarrow \neg A'$, $A' \leftarrow \neg A$ was presented by Satoh and Iwayama [1991].

Note that for all sets of atoms $S \subseteq \mathcal{A}\mathsf{t}'$, whenever an interpretation I is a model of $\{A' \leftrightarrow \neg A \mid A \in S\}$, then

$$I = (I^{\mathsf{invmod}})^{\mathsf{mod}}_S.$$

We conclude from $I \models \mathcal{P}^{\mathsf{mod}}$ that $(I^{\mathsf{invmod}})^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})} \models \mathcal{P}^{\mathsf{mod}}$, and that for all interpretations I such that $I \models \{A' \leftrightarrow \neg A \mid A \in \mathsf{undef}(\mathcal{P})\}$ the statements $I \models \mathcal{P}^{\mathsf{mod}}$ and $(I^{\mathsf{invmod}})^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})} \models \mathcal{P}^{\mathsf{mod}}$ are equivalent. We state the following correspondence:

Lemma 3.11. *For any program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is a model of the weak completion of $\mathcal{P}$.*
2. *$I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$ is a model of the completion of $\mathcal{P}^{\mathsf{mod}}$.*

Proof.
Note that if I is a model of $\mathsf{wc}\,\mathcal{P}$, it is not necessarily the case that I is a model of the completion of $\mathcal{P}$, because for all $A \in \mathsf{undef}(\mathcal{P})$ they could be either *false*, *unknown* or *true* in I. However, by the definition of the completion every model of the completion of $\mathcal{P}$ maps A to false, for all $A \in \mathsf{undef}(\mathcal{P})$.

Nevertheless, it is easy to see that all atoms, which are neither in $\mathsf{undef}(\mathcal{P})$ nor auxiliary atoms of the form A' (only occurring in $\mathcal{P}^{\mathsf{mod}}$), are mapped to the same truth values under I and $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$. Therefore, we only need to show that I and $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$ correspond with respect to all $A \in \mathsf{undef}(\mathcal{P})$ and auxiliary atoms A'.

(1) $\rightarrow$ (2): Assume that I is a model of $\mathsf{wc}\,\mathcal{P}$. What needs to be shown, is that $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$, is a model of the completion $\mathcal{P}^{\mathsf{mod}}$. By the definition of $\mathcal{P}^{\mathsf{mod}}$, for all $A \in \mathsf{undef}(\mathcal{P})$, $\mathcal{P}^{\mathsf{mod}}$ contains the two clauses $A \leftarrow \neg A'$ and $A' \leftarrow \neg A$. Accordingly, for all $A \in \mathsf{undef}(\mathcal{P})$, A and A' can be either *false*, *unknown* or *true* under any model I of the completion of $\mathcal{P}^{\mathsf{mod}}$. These models are expressed by $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$. We distinguish between three cases.

1. If $A \in \mathsf{undef}(\mathcal{P})$ and $A \notin (I^{\top} \cup I^{\bot})$, then A and A' are *unknown* in $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$.
2. If $A \in \mathsf{undef}(\mathcal{P})$ and $A \in I^{\top}$, then A' is *false* in $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$.
3. If $A \in \mathsf{undef}(\mathcal{P})$ and $A \in I^{\bot}$, then A' is *true* in $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$.

The three cases cover all the possible truth values A and A' and show that in each case $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$ is a model of the completion $\mathcal{P}^{\mathsf{mod}}$.

(2) $\rightarrow$ (1): Assume that $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$ is a model of the completion of $\mathcal{P}^{\mathsf{mod}}$. I is $I^{\mathsf{invmod}} = I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})} \setminus \mathsf{auxatoms}$. $\mathcal{P}$ is $\mathcal{P}^{\mathsf{mod}}$ without the clauses $\{A \leftarrow \neg A', A' \leftarrow \neg A \mid A \in \mathsf{undef}(\mathcal{P})\}$. As we assume that atoms in $\mathsf{auxatoms}$ only occur after the transformation of $\mathcal{P}$ in the programs $\mathcal{P}^{\mathsf{mod}}$, we know that all A' do not occur in any model of the weak completion of $\mathcal{P}$. Thus I cannot contain any $A' \in \mathsf{auxatoms}$, which corresponds to I^{invmod}. Under the

weak completion of $\mathcal{P}$ all $A \in \mathsf{undef}(\mathcal{P})$ can be either *true*, *false* or *unknown*. Accordingly, if $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$ is a model of the completion of $\mathcal{P}^{\mathsf{mod}}$, then $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$ without auxiliary atoms, that is I, is a model under the weak completion of $\mathcal{P}$. □

The relationship between I^{mod}_S and I^{invmod} indicated above allows to express Lemma 3.11 equivalently also with respect to interpretations I and I^{invmod}:

Lemma 3.12. *For any program $\mathcal{P}$ and interpretation I such that $I \models \{A' \leftrightarrow \neg A \mid A \in \mathsf{undef}(\mathcal{P})\}$ the following two statements are equivalent:*

1. *I^{invmod} is a model of the weak completion of $\mathcal{P}$.*
2. *I is a model of the completion of $\mathcal{P}^{\mathsf{mod}}$.*

We now transfer Lemma 3.11 to knowledge-least models:

Lemma 3.13. *For any program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is the knowledge-least model of the weak completion of $\mathcal{P}$.*
2. *I is the knowledge-least model of the completion of $\mathcal{P}^{\mathsf{mod}}$.*

Proof.
(1) → (2): Assume that I is the knowledge-least model of $\mathsf{wc}\,\mathcal{P}$. By Lemma 3.11 we know that $I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$ is a model of the completion of $\mathcal{P}$. By results from [Hölldobler and Kencana Ramli, 2009b] it follows for the knowledge-least model I of the weak completion of $\mathcal{P}$, that for all atoms $A \in \mathsf{undef}(\mathcal{P})$ it holds that $I(A) =$ U. Thus, if I satisfies (1), then $I = I^{\mathsf{mod}}_{\mathsf{undef}(\mathcal{P})}$. Hence, I is also the knowledge-least model of $\mathcal{P}$.
(2) → (1): Assume that I is the knowledge-least model of the completion of $\mathcal{P}^{\mathsf{mod}}$. According to Lemma 3.12 I^{invmod} is a model of the weak completion of $\mathcal{P}$. As I is knowledge-least for the completion of $\mathcal{P}^{\mathsf{mod}}$, for all atoms $A \in \mathsf{undef}(\mathcal{P})$ it holds that $I(A) =$ U. Accordingly, for all auxiliary atoms occurring in $\mathcal{P}^{\mathsf{mod}}$ $A' \notin (I^{\top} \cup I^{\bot})$. But then $I = I^{\mathsf{invmod}}$ and I is also a knowledge-least model of the completion of $\mathcal{P}$. □

The following proposition follows immediately from Lemma 3.13 and from the model intersection property for the Weak Completion Semantics shown in [Hölldobler and Kencana Ramli, 2009a] and discussed in Chapter 2.3:

Proposition 3.14. *The knowledge-least model of the completion of $\mathcal{P}^{\mathsf{mod}}$ is the intersection of all models of the completion of $\mathcal{P}^{\mathsf{mod}}$.*

Fages [1994] showed that under two-valued semantics the models of the completion of a normal logic program $\mathcal{P}$ coincide with the two-valued stable models of $\mathcal{P}$ if $\mathcal{P}$ is tight. In the following lemma, we transfer this result, which is sometimes called *Fages' theorem*, to three-valued semantics.

Lemma 3.15. *For any tight and normal program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is a model of the completion of $\mathcal{P}$.*
2. *I is a stable model of $\mathcal{P}$.*

Proof.
(1) $\rightarrow$ (2): This follows immediately from Lemma 3.8.
(2) $\rightarrow$ (1): By contradiction: Let $\mathcal{P}$ be a tight program, I a model of the completion of $\mathcal{P}$, and assume that I is not a stable model. By Lemma 3.1 and 3.7, interpretation I is supported but not well-supported. Then for all level mappings ℓ there exists an atom $A \notin I^{\perp}$ such that for all clauses $A \leftarrow \mathit{body} \in \mathcal{P}$ with L in $\mathsf{pos}(\mathit{body})$ such that $\ell(L) < \ell(A)$ does not hold. Because I is a model of the completion of $\mathcal{P}$ such a clause must indeed exist. But then there is a positive cycle in the program, in contradiction to the precondition that $\mathcal{P}$ is tight. □

In the following corollary, we instantiate Lemma 3.15 with $\mathcal{P}^{\mathsf{mod}}$:

Corollary 3.16. *For any tight and normal program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is a model of the completion of $\mathcal{P}^{\mathsf{mod}}$.*
2. *I is a stable model of $\mathcal{P}^{\mathsf{mod}}$.*

Proof.
By Lemma 3.10 it holds for all tight programs P that $\mathcal{P}^{\mathsf{mod}}$ is normal and tight. The corollary then is an immediate consequence of Lemma 3.15. □

The following proposition follows immediately from Proposition 3.14 and from Corollary 3.16:

Proposition 3.17. *Given that $\mathcal{P}$ is a tight and normal program, the knowledge-least stable model of $\mathcal{P}^{\mathsf{mod}}$ is the intersection of all stable models of $\mathcal{P}^{\mathsf{mod}}$.*

In the following corollary, restrict the considered interpretations to knowledge-least models.

Corollary 3.18. *For any tight and normal program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is the knowledge-least model of the completion of $\mathcal{P}^{\mathsf{mod}}$.*
2. *I is the knowledge-least stable model of $\mathcal{P}^{\mathsf{mod}}$.*

Proof.
Corollary 3.16 states that the set of stable models of $\mathcal{P}^{\mathsf{mod}}$ and the set of models of the completion of $\mathcal{P}^{\mathsf{mod}}$ are the same. This Corollary follows immediately given Proposition 3.14 and Proposition 3.17. □

Przymusinski [1990] has shown that knowledge-least stable models coincide with well-founded models:

Lemma 3.19. *For any normal program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is the knowledge-least stable model of $\mathcal{P}$.*
2. *I is the well-founded model of $\mathcal{P}$.*

In the following corollary we instantiate this result by Przymusinski with $\mathcal{P}^{\mathsf{mod}}$.

Corollary 3.20. *For any program $\mathcal{P}$ and interpretation I the following two statements are equivalent:*

1. *I is the knowledge-least stable model of $\mathcal{P}^{\mathsf{mod}}$.*
2. *I is the well-founded model of $\mathcal{P}^{\mathsf{mod}}$.*

Proof.
Follows as corollary from Lemma 3.10(ii) and Lemma 3.19. □

Finally we combine the material developed in this section to prove Theorem 3.9:

Proof of Theorem 3.9.
Let $\mathcal{P}$ be a tight program and let I be an interpretation. Then the following four statements are equivalent:

1. I is the knowledge-least model of the weak completion of $\mathcal{P}$.
2. I is the knowledge-least model of the completion of $\mathcal{P}^{\mathsf{mod}}$ (by Lemma 3.13).
3. I is the knowledge-least stable model of $\mathcal{P}^{\mathsf{mod}}$ (by Lemma 3.10(i) and Corollary 3.18).
4. I is the well-founded model of $\mathcal{P}^{\mathsf{mod}}$ (by Corollary 3.20). □

Appendix B shows the correspondence between the knowledge-least model of the weak completion and the well-founded model with another proof technique, where level mapping characterizations of both semantics are directly compared. While this only shows the correspondences between knowledge-least models, with the techniques applied in this section, we have been able to prove results that apply to three-valued models in general, in particular Lemma 3.11 and 3.15.

3.4. Evaluation: A Psychological Study

There are two main differences between the Weak Completion and the Well-founded Semantics: First, how they deal with positive cycles in logic programs and, second, that Well-founded Semantics adopts the closed-world assumption for undefined atoms. While

under the Well-founded Semantics atoms are false if they are not facts and involved in positive cycles, they stay unknown under the Weak Completion Semantics. A natural way to determine which semantics is more adequate for human reasoning, is to investigate which conclusions are typically drawn by human reasoners with respect to (positive) cyclic conditionals. For this purpose, we carried out a psychological study.

Participants We tested 35 participants on an online website (Amazon Mechanical Turk). They were paid for their participation.

Material, Procedure and Design Participants were presented with 17 problems consisting of cyclic conditionals of length 1, 2 and 3. Consider the following cyclic conditional of length 1:

If they open the window, then they open the window.

Participants were asked about the consequences of this conditional and could choose between one of the following three offered conclusions:

- ☐ *They open the window.*
- ☐ *They do not open the window.*
- ☐ *It is unknown whether they open the window.*

Another example is the following positive cyclic conditional of length 3:

If they open the window, then it is cold.
If it is cold, then they wear their jackets.
If they wear their jackets, then they open the window.

We investigated three kinds of facts, namely whether they open the window, whether it is cold, and whether they wear their jacket; each of them under the three conditions positive, negative, and unknown.

Results and Discussion The results summarized in Table 3.2 indicate two kinds of groups each taking a different interpretation of the statements: One group consists of participants understanding the programs as a conditional, which in our approach, for positive cycles of length one, is modeled by the following clause:

$$p \leftarrow p \wedge \neg ab.$$

Length of Cycle	Chosen Answer in Percentage			Mean Response Times in Msec[4]
	Positive	Negative	Unknown	
1	75	0	25	5267
2	60	3	37	11516
3	55	4	41	11680

Table 3.2.: The results of the participants' answers.

For cycles of length 2, it is modeled by the following two clauses:

$$\begin{array}{lcl} p & \leftarrow & q \wedge \neg ab_1. \\ q & \leftarrow & p \wedge \neg ab_2. \end{array}$$

For cycles of length 3 it is modeled by three clauses, analogously. If we assume that nothing abnormal is known, (i.e. $ab \leftarrow \bot$), then the least model of the weak completion is $\langle \emptyset, \{ab\} \rangle$. In contrast, the Well-founded Semantics always and independently of the truth value of ab concludes $\neg p$, a conclusion almost no participant had drawn. The other interpretation, where participants' chose to give a positive answer, apparently treats the statement as a fact, $p \leftarrow \top$. If we consider this as the result of the first step of the Stenning and van Lambalgen procedure (reasoning towards an adequate representation), then both, the Weak Completion and the Well-founded Semantics, seem to be adequate. The findings show that the chosen answers associated with facts decrease from cycles of length 1 (75% positive answers) to cycles of length 3 (55% positive answers) accompanied by a raise in choosing the truth-value unknown. The response times indicate a higher degree of uncertainty in case of problems involving cycles of length 2 and 3 in contrast to the simpler problems involving a cycle of length 1. Taken together, the increase in choosing the truth value unknown and the increase in response time shows an increasing likelihood of the participants to respond in accord with the Weak Completion Semantics.

When participants were given conditionals with negative cycles of the form

$$\begin{array}{lcl} p & \leftarrow & \neg q \wedge \neg ab_1. \\ q & \leftarrow & \neg p \wedge \neg ab_2. \end{array} \qquad \begin{array}{lcl} ab_1 & \leftarrow & \bot. \\ ab_2 & \leftarrow & \bot. \end{array}$$

then the majority concluded that the given facts were unknown. This result corresponds to both the Weak Completion and the Well-founded Semantics.

Summing up, it seems that when we consider the two representational forms for the conditionals, the Weak Completion Semantics can better directly explain and predict participants' responses than the Well-founded Semantics. As discussed by Wernhard [2012], it would be interesting to further examine whether there are real world situations

[4]Time after having read the conditionals.

in which humans actually reason with cycles and how they extract knowledge based on these seemingly meaningless data.[5]

3.5. Conclusion

In order to show the correspondence between the Weak Completion Semantics and the Well-founded Semantics, we extended the two-valued characterization for supported and well-supported models to three-valued logics and examined quite generally the properties of the Weak Completion, the Completion and the Stable Model Semantics. When we restrict ourselves to tight logic programs and apply some technical modifications on the program, the Weak Completion Semantics and the Partial Stable Model Semantics, which rests on the Well-founded Semantics, yield the same results.

This gives us insights into the behavior of the considered semantics. Undefined atoms, i.e. there is no clause in which these atoms are the head of, are always false under the (Partial) Stable Model Semantics. The same holds for atoms that can only be justified through positive cycles. If the only possibility for justification available is through some cycle that involves negation, atoms are unknown in the well-founded model.

In the context of aiming at adequately modeling human reasoning, it is natural to ask if these technical properties are somehow reflected by human behavior. We give a first attempt at an evaluation in Section 3.4. The empirical results obtained so far indicate that, in the presence of positive circular dependencies, people tend to infer unknown in contrast to false. This result is in the spirit of the Weak Completion Semantics in contrast to what is suggested by the direct application of the Well-founded or the Stable Model Semantics.

[5] According to Luís Moniz Pereira, this data is not so meaningless: The premise of a positive loop may be seen as an abduction (viz 'if they open the window'), hence it allows itself as a conclusion. As the length of the cycle increases that abduction is further away and not recalled as well. (personal communication, February 11, 2016)

4. Contextual Reasoning

One of the properties of the Weak Completion Semantics is the open world assumption with respect to undefined atoms. This is a characteristic that is different to other common Logic Programming semantics, a property that seems suitable when modeling human reasoning. However, we have noticed that the famous Tweety default reasoning example, originally introduced by Reiter, cannot be modeled directly under the Weak Completion Semantics. To address the issue and taking Pereira and Pinto's inspection points as inspiration, we develop a notion of contextual reasoning for which we introduce contextual logic programs. We then reconsider the formal properties of the Weak Completion Semantics with respect to these and verify whether they still hold. Finally, we present contextual abduction and show that not only the original Tweety example can be nicely modeled within the new approach, but that we can specify the relations between observations and their contextual explanations, such as contextual side-effects, (strict) possible side-effects, contextual contestable side-effects, and (jointly supported) contextual relevant consequences.[1]

4.1. Introduction

Consider the famous Tweety example from Reiter [1980]: *Usually birds can fly. Tweety and Jerry are birds.* Adapted from $\mathcal{P}_{fly}$ of the introduction in Chapter 3, this example can be encoded by the following (datalog) program, $\mathcal{P}^1_{fly}$, where ab stands for abnormal:

$$\begin{array}{rcl} can\_fly(X) & \leftarrow & bird(X) \wedge \neg ab(X). \\ ab(X) & \leftarrow & \bot. \\ bird(tweety) & \leftarrow & \top. \\ bird(jerry) & \leftarrow & \top. \end{array}$$

We derive $can\_fly(tweety)$ and $can\_fly(jerry)$ as nothing is abnormal with respect to Tweety and Jerry.

[1] The original idea for this chapter has been published in [Pereira, Dietz, and Hölldobler, 2014a]. The formalization within the Weak Completion Semantics presented in Section 4.2,4.3 and 4.4 has been published in [Dietz Saldanha, Hölldobler, and Pereira, 2017b]. The results of Section 4.5 have been published in [Pereira, Dietz, and Hölldobler, 2014b].

We modify the example by replacing the first statement with: *Usually birds can fly, but kiwis and penguins cannot.* This is encoded by $\mathcal{P}^2_{fly}$:

$$\begin{array}{rcl} can\_fly(X) & \leftarrow & bird(X) \wedge \neg ab(X). \\ ab(X) & \leftarrow & kiwi(X). \\ ab(X) & \leftarrow & penguin(X). \\ bird(tweety) & \leftarrow & \top. \\ bird(jerry) & \leftarrow & \top. \end{array}$$

The ground instances of the weak completion of this program are as follows:

$$\begin{array}{rcl} can\_fly(tweety) & \leftrightarrow & bird(tweety) \wedge \neg ab(tweety). \\ can\_fly(jerry) & \leftrightarrow & bird(jerry) \wedge \neg ab(jerry). \\ ab(tweety) & \leftrightarrow & kiwi(tweety) \vee penguin(tweety). \\ ab(jerry) & \leftrightarrow & kiwi(jerry) \vee penguin(jerry). \\ bird(tweety) & \leftrightarrow & \top. \\ bird(jerry) & \leftrightarrow & \top. \end{array}$$

The least model of the weak completion of $\mathcal{P}^2_{fly}$ is

$$\langle \{bird(tweety), bird(jerry)\}, \emptyset \rangle.$$

Different than under Clark's [1978] Completion Semantics, the Stable Model Semantics [Gelfond and Lifschitz, 1988] and the Well-Founded Semantics [Van Gelder, Ross, and Schlipf, 1991], the closed-world assumption does not apply for undefined atoms, i.e. $kiwi(tweety)$, $penguin(tweety)$, $kiwi(jerry)$ and $penguin(jerry)$ are not false, but stay unknown under the Weak Completion Semantics. In other words, they are neither true nor false. This leads to the following consequence under WCS: As we don't know whether Tweety and Jerry are penguins or kiwis, we cannot derive that they can fly. Even if we model this case with the help of abduction, e.g. we observe that Jerry flies,

$$\mathcal{O} = \{can\_fly(jerry)\},$$

The set of abducibles $\mathcal{A}_{\mathcal{P}^2_{fly}}$ consists of the following facts and assumptions:

$$\begin{array}{rcl} kiwi(tweety) & \leftarrow & \bot. \\ kiwi(tweety) & \leftarrow & \top. \\ kiwi(jerry) & \leftarrow & \bot. \\ kiwi(jerry) & \leftarrow & \top. \end{array} \qquad \begin{array}{rcl} penguin(tweety) & \leftarrow & \bot. \\ penguin(tweety) & \leftarrow & \top. \\ penguin(jerry) & \leftarrow & \bot. \\ penguin(jerry) & \leftarrow & \top. \end{array}$$

Considering the abductive framework $\langle \mathcal{P}^2_{fly}, \mathcal{A}_{\mathcal{P}^2_{fly}}, \emptyset, \models_{wcs} \rangle$ and $\mathcal{O} = \{can\_fly(jerry)\}$ we obtain the minimal explanation $\mathcal{E} = \{kiwi(jerry) \leftarrow \bot,\ penguin(jerry) \leftarrow \bot\}$. In other words, in order to explain the observation that *Jerry can fly* we have to assume *Jerry* does not belong to any of the known exceptions. The question that arises already in [Reiter, 1980] is whether and how we can avoid the explicit investigation into all

L	$\mathsf{ctxt}(L)$
$\top$	$\top$
$\bot$	$\bot$
U	$\bot$

Table 4.1.: Truth table for $\mathsf{ctxt}(L)$ where L is a literal.

known exceptions and conclude instead that *Jerry can fly by default*? In other words, do humans normally, knowing the statements mentioned above and observing that *Jerry can fly*, accept this observation as *default* or do they accept this observation only after explicitly reasoning just in case *Jerry is not a kiwi* and *Jerry is not a penguin*?

We want to avoid explicitly stating that all exception cases are false such as the Completion Semantics, the Stable Model Semantics or Well-Founded Semantics will do. We don't think that humans actively apply the closed world assumption in reality, i.e. that they explicitly add negation to the cases they don't know anything about. Instead, we assume that humans, if they are not for some reason aware of exceptions, simply ignore these cases. In other words, they do not consciously become aware of all exceptions when they are reasoning.[2] Accordingly, when modeling these cases with logic programs, we should leave the truth values of these exception cases unknown and find a mechanism that just ignores them. At the moment, we cannot express this syntactically in WCS programs.

In the next section, we present contextual programs and verify whether the same properties of the $\Phi_{\mathcal{P}}$ operator hold for contextual programs, as for the programs we have considered so far in our modeling of applications. Section 4.3 presents contextual abductive reasoning and specifies the notion of contextual side-effects. We finish with conclusions, including open questions.

4.2. Contextual Programs

In [Pereira and Pinto, 2011], the authors introduced *inspection points*: $\mathit{inspect}(L)$ can only be abduced to explain some observation in case L has been abduced to explain some other observation: A set of literals is only an explanation if for each $\mathit{inspect}(L)$ we find that L is in the explanation.[3] Pereira and Pinto employ the concepts of a *consumer*, represented by the inspection point $\mathit{inspect}(L)$, which must have a matching *producer* corresponding to the usual abducibles, thus L.

Inspired by the idea underlying inspection points, we introduce a new truth-functional operator ctxt (called *context*), whose meaning is specified in Table 4.1. As we will see

[2] Currently, we know of at least 40 species of birds that can't fly.

[3] Note that, different than in this thesis, Pereira and Pinto define the set of abducibles as a set of literals.

Example 4.1. Consider the program $\mathcal{P}$ consisting of the following two clauses:

$$\begin{array}{lll} p & \leftarrow & q. \\ p & \leftarrow & \bot. \end{array}$$

Consider the weak completion of $\mathcal{P}$:

$$p \quad \leftrightarrow \quad q \vee \bot.$$

which is semantically equivalent to

$$p \quad \leftrightarrow \quad q.$$

The least model of the weak completion of $\mathcal{P}$ is

$$\langle \emptyset, \emptyset \rangle.$$

The assumption about p has been overridden by the first clause of $\mathcal{P}$ and does not have any effect at all.

later, with the help of ctxt, preferences on explanations, among other things, can be syntactically specified. These preferences are context-dependent.

The interpretation of ctxt is specified in Table 4.1 and can be understood as a mapping from three-valuedness to two-valuedness. It is one possible way on how to understand negation as failure under three-valued semantics. The idea of negation as failure is to derive the negation of A in case we fail to derive A [Clark, 1978]. Negation as failure does not exist under the Weak Completion Semantics, quite the contrary is the case: Example 4.1 shows that undefined bodies are prioritized over assumptions. Under the Weak Completion Semantics, unknown information is always prioritized over negative information.

We extend the definition for logic programs from Chapter 2 by allowing expressions of the form $\mathsf{ctxt}(L)$ in the body of the clauses. Formally, *contextual clauses* are expressions of the form

$$A \quad \leftarrow \quad L_1 \wedge \ldots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \ldots \wedge \mathsf{ctxt}(L_{m+p}),$$

where $m, p \in \mathbb{N}$ such that $m + p \geq 1$. A *contextual datalog program* is a finite set of contextual clauses, facts and assumptions. Example 4.2 shows a contextual program with the ctxt operator and the corresponding truth tables which indicate the corresponding models of $\mathcal{P}$ and the models of $\mathsf{wc}\,\mathcal{P}$. Example 4.3 shows a program and a contextual program and their corresponding least fixed points.

By means of ctxt, the common syntactical form for integrity constraints can be re-established: $\mathcal{IC}$ comprises clauses of the form $\bot \leftarrow \mathit{body}$, where *body* is a conjunction of

Example 4.2. Consider the program $\mathcal{P}$, consisting of exactly one clause:

$$p \leftarrow \mathsf{ctxt}(q).$$

Model of $\mathsf{wc}\,\mathcal{P}$	Model of $\mathcal{P}$	$\mathsf{wc}\,\mathcal{P}$	$\mathcal{P}$	p	q	$\mathsf{ctxt}(q)$
✓	✓	$\top$	$\top$	$\top$	$\top$	$\top$
	✓	$\bot$	$\top$	$\top$	U	$\bot$
	✓	$\bot$	$\top$	$\top$	$\bot$	$\bot$
		$\bot$	$\bot$	U	$\top$	$\top$
	✓	U	$\top$	U	U	$\bot$
	✓	U	$\top$	U	$\bot$	$\bot$
		$\bot$	$\bot$	$\bot$	$\top$	$\top$
✓	✓	$\top$	$\top$	$\bot$	U	$\bot$
✓	✓	$\top$	$\top$	$\bot$	$\bot$	$\bot$

✓ indicates whether the interpretation is a model of $\mathcal{P}$ or $\mathsf{wc}\,\mathcal{P}$, respectively.

Example 4.3. Consider the following two programs consisting each of one clause where only $\mathcal{P}_2$ contains a ctxt operator in the body of the clause. Additionally, consider their weak completion:

$$\begin{array}{rclcrcl} \mathcal{P}_1 & = & \{p \leftarrow \neg q\}, & & \mathcal{P}_2 & = & \{p \leftarrow \mathsf{ctxt}(\neg q)\}, \\ \mathsf{wc}\,\mathcal{P}_1 & = & \{p \leftrightarrow \neg q\}, & & \mathsf{wc}\,\mathcal{P}_2 & = & \{p \leftrightarrow \mathsf{ctxt}(\neg q)\}. \end{array}$$

Starting with $I_0 = \langle \emptyset, \emptyset \rangle$ we compute the corresponding least fixed points of $\Phi_{\mathcal{P}}$:

$$\Phi_{\mathcal{P}_1}(I_0) = \langle \emptyset, \emptyset \rangle = I_0, \qquad \begin{array}{rcccl} \Phi_{\mathcal{P}_2}(I_0) & = & \langle \emptyset, \{p\} \rangle & = & I_1, \\ \Phi_{\mathcal{P}_2}(I_1) & = & \langle \emptyset, \{p\} \rangle & = & I_1. \end{array}$$

Here, $\mathsf{ctxt}(\neg q)$ in $\mathcal{P}_2$ behaves as negation as failure:
Nothing is known about q, therefore we derive that p is false in $\mathcal{P}_2$.

$L_1 \wedge \ldots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \ldots \wedge \mathsf{ctxt}(L_{m+p})$, $m, p \in \mathbb{N}$ and $m+p \geq 1$. Given $\mathcal{P}$ and $\mathcal{IC}$, $\mathcal{P}$ *satisfies* $\mathcal{IC}$ iff for all $\bot \leftarrow body \in \mathcal{IC}$, we find that under the least Ł-model of $\mathcal{P}$, $\mathcal{M}_\mathcal{P}$, $\mathcal{M}_\mathcal{P}(body) = \bot$. Because ctxt is allowed in the body of these clauses, the same understanding as we discussed in Chapter 2.4 can be maintained: If a literal L should be either unknown or false, then we can simply write $\bot \leftarrow \mathsf{ctxt}(L)$. In the remainder of this Chapter, we assume that the underlying semantics of these contextual programs is Łukasiewicz semantics as defined in Chapter 2 extended by the truth table for ctxt defined in Table 4.1.

Although the $\Phi_\mathcal{P}$ operator admits a least fixed point for $\mathcal{P}_2$ in Example 4.3, we need to check if this holds in general, i.e. whether there exists a least fixed point for all contextual programs under the modified logic. The Knaster-Tarski theorem ensures that every monotonic mapping on a complete partial order has a least fixed point [Tarski, 1955].

Theorem 4.1. *Let C be a complete partial order and f be a monotonic mapping on C. Then f has a least fixed point.*

A proof can be found in [Davey and Priestley, 2002].

Kencana Ramli [2009] has shown that the space of interpretations $\mathcal{I}$ is a complete partial order with respect to the set inclusion $\subseteq$ and that $\Phi_\mathcal{P}$ is monotonic with respect to all programs $\mathcal{P}$, i.e. for all programs $\mathcal{P}$ and all interpretations $I, J \in \mathcal{I}$

$$I \subseteq J \qquad \text{implies} \qquad \Phi_\mathcal{P}(I) \subseteq \Phi_\mathcal{P}(J).$$

According to the Knaster-Tarski Fixpoint Theorem, the $\Phi_\mathcal{P}$ operator has a least fixed point for all programs $\mathcal{P}$. Is $\Phi_\mathcal{P}$ monotonic for all contextual programs $\mathcal{P}$? As Example 4.4 shows, this is not the case. Furthermore, as Example 4.5 shows, $\Phi_\mathcal{P}$ does not have a (least) fixed point for every program. Nevertheless, we can possibly guarantee a least fixed point for a particular class of contextual programs. We will follow an idea first developed by Fitting [1994] for programs under Kripke-Kleene logic. The idea was adapted to programs under the Weak Completion Semantics in [Hölldobler and Kencana Ramli, 2009c, Kencana Ramli, 2009].

The Banach Contraction Theorem states that every contraction mapping has a unique fixed point. The idea is to show that $\Phi_\mathcal{P}$ is a contraction on an appropriately defined metric space. Unfortunately, in general semantic operators are not contractions and as Example 4.5 shows, the $\Phi_\mathcal{P}$ operator does not necessarily have a least fixed point for contextual programs. The usual restriction to acceptable programs for the Fitting [1994] operator can not be applied to the $\Phi_\mathcal{P}$ operator [Hölldobler and Kencana Ramli, 2009c, Kencana Ramli, 2009]. However, as has been shown in [Hölldobler and Kencana Ramli, 2009c, Kencana Ramli, 2009], the $\Phi_\mathcal{P}$ operator is a contraction for all acyclic programs. Analogously, we will show, that the $\Phi_\mathcal{P}$ operator is a contraction for all acyclic contextual programs.

Before we can show that every acyclic contextual program is a contraction, we need to introduce some definitions.

Example 4.4. Consider the program $\mathcal{P}$ consisting of three clauses:

$$\begin{array}{lcl} p & \leftarrow & q \wedge \mathsf{ctxt}(r). \\ r & \leftarrow & \neg s. \\ s & \leftarrow & \mathsf{ctxt}(t). \end{array}$$

Iterating $\Phi_{\mathcal{P}}$ starting with the empty interpretation $I_0 = \langle \emptyset, \emptyset \rangle$ results in

$$\begin{array}{lcccl} \Phi_{\mathcal{P}}(I_0) & = & \langle \emptyset, \{p, s\} \rangle & = & I_1, \\ \Phi_{\mathcal{P}}(I_1) & = & \langle \{r\}, \{p, s\} \rangle & = & I_2, \\ \Phi_{\mathcal{P}}(I_2) & = & \langle \{r\}, \{s\} \rangle & = & I_3, \\ \Phi_{\mathcal{P}}(I_3) & = & \langle \{r\}, \{s\} \rangle & = & \mathsf{lfp}\,(\Phi_{\mathcal{P}}). \end{array}$$

$I_1 \subseteq I_2$, but $\Phi_{\mathcal{P}}(I_1) \subseteq \Phi_{\mathcal{P}}(I_2)$ does not hold. $\Phi_{\mathcal{P}}$ is not monotonic!

Example 4.5. Consider program $\mathcal{P}$ which consists of exactly one clause:

$$p \leftarrow \mathsf{ctxt}(\neg p).$$

Let us try to compute the least fixed point of $\Phi_{\mathcal{P}}$ starting with $I_0 = \langle \emptyset, \emptyset \rangle$:

$$\begin{array}{lcccl} \Phi_{\mathcal{P}}(I_0) & = & \langle \emptyset, \{p\} \rangle & = & I_1, \\ \Phi_{\mathcal{P}}(I_1) & = & \langle \{p\}, \emptyset \rangle & = & I_2, \\ \Phi_{\mathcal{P}}(I_2) & = & \langle \{\emptyset, \{p\} \rangle & = & I_3, \\ \Phi_{\mathcal{P}}(I_3) & = & \langle \emptyset, \{p\} \rangle & = & I_1 \ldots \end{array}$$

For some contextual programs, $\Phi_{\mathcal{P}}$ does not have a (least) fixed point.

A *metric* or a distance function on a space $\mathcal{M}$ is a mapping

$$d : \mathcal{M} \times \mathcal{M} \mapsto \mathbb{R}$$

such that

$$d(x, y) = 0 \text{ if and only if } x = y \qquad (\mathrm{m}_1)$$
$$d(x, y) = d(y, x) \qquad (\mathrm{m}_2)$$
$$d(x, y) \leq d(x, z) + d(z, y) \qquad (\mathrm{m}_3)$$

A metric space $\mathcal{M}$ is *complete* if every Cauchy sequence converges.

A sequence $s_1, s_2, s_3, \ldots$ is *Cauchy* if, for every $\epsilon > 0$ there is an integer N such that for all $n, m \geq N, d(s_n, s_m) \leq \epsilon$. The sequence *converges* if there is an s such that, for every $\epsilon > 0$, there is an integer N such that for all $n \geq N, d(s_n, s) \leq \epsilon$.

Let $\mathcal{M}$ be a metric space: A mapping

$$f : \mathcal{M} \mapsto \mathcal{M}$$

is a *contraction* if for all $x, y \in \mathcal{M}$ there exists a $k \in \mathbb{R}$ with $0 < k < 1$ such that

$$d(f(x), f(y)) \leq k \cdot d(x, y).$$

The Banach Contraction Theorem ensures that every contraction has a unique fixed point [Banach, 1922].

Theorem 4.2. *A contraction mapping f on a complete metric space has a unique fixed point. Further, the sequence $x, f(x), f(f(x)), \ldots$ converges to this fixed point for any x.*

We specify acyclic contextual programs through a level mapping characterization, which we have already done in Chapter 3 for programs. A *level mapping* for a contextual program $\mathcal{P}$ is a function ℓ which assigns to each ground atom a natural number. It is extended to ground literals and expressions of the form $\mathsf{ctxt}(L)$ as follows, where L is a ground literal and A a ground atom: $\ell(\neg A) = \ell(A)$ and $\ell(\mathsf{ctxt}(L)) = \ell(L)$. Additionally, ℓ is extended to the truth-value constants $\top$ and $\bot$, where $\ell(\top) = \ell(\bot) = 0$. A *contextual program $\mathcal{P}$ is acyclic with respect to a level mapping ℓ* iff for every $A \leftarrow L_1 \wedge \ldots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \ldots \wedge \mathsf{ctxt}(L_{m+p}) \in \mathcal{P}$ and for all i, $0 \leq i \leq m$, we find that $\ell(A) > \ell(L_i)$ and for all j, $m + 1 \leq j \leq m + p$, we find that $\ell(A) > \ell(\mathsf{ctxt}(L_j))$. A *contextual program $\mathcal{P}$ is acyclic* iff it is acyclic with respect to some level mapping ℓ.

Consider again $\mathcal{P}$ in Example 4.4: With $\ell(t) = 1$, $\ell(s) = 2$, $\ell(q) = \ell(r) = 3$ and $\ell(p) = 4$, we find that $\mathcal{P}_1$ is acyclic. On the other hand, $\mathcal{P}$ in Example 4.5 is not acyclic as we find that $\ell(p) = \ell(\neg p) = \ell(\mathsf{ctxt}(\neg p))$.

Kencana Ramli [2009] showed that the space of all three-valued interpretations $\mathcal{I}$ is a metric by specifying such metrics based on level mappings. Let ℓ be a level mapping

and I and J be interpretations. The function $d_\ell : \mathcal{I} \times \mathcal{I} \mapsto \mathbb{R}$ is defined as

$$d_\ell(I,J) = \begin{cases} (\frac{1}{2})^n & I \neq J \text{ and } I(A) = J(A) \neq \mathrm{U} \text{ for all } A \text{ with } \ell(A) < n \text{ and,} \\ & \text{for some } A \text{ with } \ell(A) = n, I(A) \neq J(A) \text{ or } I(A) = J(A) = \mathrm{U}, \\ 0 & \text{otherwise.} \end{cases}$$

Proposition 4.3 (Kencana Ramli [2009])**.** *d_ℓ is a metric.*

Proposition 4.4 (Kencana Ramli [2009])**.** *The space of three-valued interpretations $\mathcal{I}$ using the metric d_ℓ is a complete metric space.*

Theorem 4.5. *Let $\mathcal{P}$ be an acyclic contextual program with respect to the level mapping ℓ. Then $\Phi_\mathcal{P}$ is a contraction on $(\mathcal{I}, d_\ell)$.*

Proof.
Given that $\mathcal{P}$ is a contextual program, we will show that

$$d_\ell(\Phi_\mathcal{P}(I), \Phi_\mathcal{P}(J)) \leq \frac{1}{2} \cdot d_\ell(I,J). \tag{1}$$

If $I = J$, then $\Phi_\mathcal{P}(I) = \Phi_\mathcal{P}(J)$, $d_\ell(\Phi_\mathcal{P}(I), \Phi_\mathcal{P}(J)) = d_\ell(I,J) = 0$, and (1) holds.
If $I \neq J$, then since ℓ is total, we obtain $d_\ell(I,J) = \frac{1}{2}^n$ for some $n \in \mathbb{N}$. We will show that $d_\ell(\Phi_\mathcal{P}(I), \Phi_\mathcal{P}(J)) \leq (\frac{1}{2})^{n+1}$, i.e. for all ground atoms $A \in g\mathcal{P}$, with $\ell(A) \leq n$ we have that $\Phi_\mathcal{P}(I)(A) = \Phi_\mathcal{P}(J)(A)$.
Let us take some A with $\ell(A) \leq n$ and let $\mathsf{def}(A, \mathcal{P})$ be the set of all clauses in $g\mathcal{P}$ where A is the head of. As $\mathcal{P}$ is acyclic, for any clause

$$A \leftarrow L_1 \wedge \ldots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \ldots \wedge \mathsf{ctxt}(L_{m+p}) \in \mathsf{def}(A, \mathcal{P})$$

for all i, $1 \leq i \leq m$ we obtain

$$\ell(L_i) < \ell(A) \leq n,$$

and for all j, $m+1 \leq j \leq m+p$, we obtain

$$\ell(L_j) < \ell(A) \leq n.$$

We know that $d_\ell(I,J) \leq (\frac{1}{2})^n$, so for all i, $1 \leq i \leq m$, $I(L_i) = J(L_i)$, and for all j, $m+1 \leq j \leq m+p$,

$$I(L_j) = J(L_j).$$

Therefore, I and J interpret identically all bodies of clauses with A in the head. Consequently, $\Phi_\mathcal{P}(I)(A) = \Phi_\mathcal{P}(J)(A)$. □

Corollary 4.6. *Let $\mathcal{P}$ be an acyclic contextual logic program. Then $\Phi_\mathcal{P}$ has a unique fixed point. Further, this fixed point can be reached by iterating a finite number of times starting from any interpretation.*

Proof.

1. By Proposition 4.4 the space of three-valued interpretations $\mathcal{I}$ using the metric d_ℓ is a complete metric space.
2. By Theorem 4.5, $\Phi_\mathcal{P}$ is a contraction for acyclic contextual $\mathcal{P}$ on $\mathcal{I}$ using the metric d_ℓ.
3. By 1., 2. and the Banach Contraction Theorem, Theorem 4.2, for any acyclic contextual $\mathcal{P}$, $\Phi_\mathcal{P}$ has a unique fixed point. Further, this fixed point that can be reached by starting from any interpretation.
4. By 3. and because $\mathcal{P}$ is finite, $\Phi_\mathcal{P}$ can be reached in a finite number of times starting from any interpretation. □

The proof of the following result for contextual $\mathcal{P}$ is analogous to the proof for $\mathcal{P}$ in [Kencana Ramli, 2009].

Proposition 4.7. *Let $\mathcal{P}$ be an acyclic contextual program.*
Then $\mathsf{lfp}\,(\Phi_\mathcal{P})$ is a model of $\mathsf{wc}\,\mathcal{P}$.

Proof.
Assume that $\mathsf{lfp}\,\Phi_\mathcal{P} = \langle I^\top, I^\bot \rangle$ and $A \leftrightarrow F \in \mathsf{wc}\,\mathcal{P}$. We distinguish between 3 cases:

1. If $I(A) = \top$, then according to the definition of $\Phi_\mathcal{P}$, there exists a clause $A \leftarrow L_1 \wedge \cdots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \cdots \wedge \mathsf{ctxt}(L_{m+p})$, such that for all i, $1 \leq i \leq m+p$, $I(L_i) = \top$. As $L_1 \wedge \cdots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \cdots \wedge \mathsf{ctxt}(L_{m+p})$ is one of the disjuncts in F, $I(F) = \top$, and thus $I(A \leftrightarrow F) = \top$.
2. If $I(A) = \mathrm{U}$, then according to the definition of $\Phi_\mathcal{P}$, there is no clause $A \leftarrow L_1 \wedge \cdots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \cdots \wedge \mathsf{ctxt}(L_{m+p})$, such that for all i, $1 \leq i \leq m+p$, $I(L_i) = \top$ and there exists at least one clause $A \leftarrow L_1 \wedge \cdots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \cdots \wedge \mathsf{ctxt}(L_{m+p})$, such that for all i, $1 \leq i \leq m+p$, $I(L_i) \neq \bot$ and there exists j, $1 \leq j \leq m$, $I(L_j) = \mathrm{U}$. As none of the disjuncts in F is true, and at least one is unknown, $I(F) = \mathrm{U}$ and thus $I(A \leftrightarrow F) = \top$.
3. If $I(A) = \bot$, then according to the definition of $\Phi_\mathcal{P}$, there exists a clause $A \leftarrow L_1 \wedge \cdots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \cdots \wedge \mathsf{ctxt}(L_{m+p})$ and for all clauses $A \leftarrow L_1 \wedge \cdots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \cdots \wedge \mathsf{ctxt}(L_{m+p})$, there exists i, $1 \leq i \leq m+p$ such that $I(L_i) = \bot$ or there exists j, $m+1 \leq j \leq m+p$, such that $I(L_j) \neq \top$. As all disjuncts in F are false, $I(F) = \bot$ and thus $I(A \leftrightarrow F) = \top$. □

The least fixed point of $\Phi_\mathcal{P}$ is identical to the least model of the weak completion of (non-contextual) $\mathcal{P}$, which always exists [Hölldobler and Kencana Ramli, 2009a]. As Example 4.6 shows, this property does not extend to contextual programs: The weak completion of contextual programs can have more than one minimal model. In the sequel of this chapter, $\models_{wcs}$ is defined as $\mathcal{P} \models_{wcs} F$ iff $\mathcal{P}$ is acyclic and $\mathsf{lfp}\ \Phi_\mathcal{P} \models F$.

Example 4.6. Consider $\mathcal{P}$ consisting of the following three clauses:

$$\begin{array}{lcl} s & \leftarrow & \neg r. \\ r & \leftarrow & \neg p \wedge q. \\ q & \leftarrow & \mathsf{ctxt}(\neg p). \end{array}$$

Its weak completion, $\mathsf{wc}\,\mathcal{P}$, consists of the following equivalences:

$$\begin{array}{lcl} s & \leftrightarrow & \neg r. \\ r & \leftrightarrow & \neg p \wedge q. \\ q & \leftrightarrow & \mathsf{ctxt}(\neg p) \end{array}$$

The least fixed point of $\Phi_{\mathcal{P}}$ is $\langle \{s\}, \{q, r\} \rangle$, which is a minimal model of $\mathsf{wc}\,\mathcal{P}$. However, yet another minimal model of $\mathsf{wc}\,\mathcal{P}$ is $\langle \{q, r\}, \{p, s\} \rangle$.

Example 4.7. The program $\mathcal{P}$ consists of the following two clauses:

$$\begin{array}{lcl} p & \leftarrow & r. \\ p & \leftarrow & \mathsf{ctxt}(q). \end{array}$$

p depends on r, p depends on $\neg r$, $\neg p$ depends on r and $\neg p$ depends on $\neg r$. However, p does not depend on q, neither on $\mathsf{ctxt}(q)$.

4.3. Contextual Abduction

How can we prefer explanations that explain the normal cases to explanations that explain the exception cases? How can we express that some explanations have to be considered only if there is some evidence for considering the exception cases? We want to avoid having to consider all explanations if there is no evidence for considering exception cases. On the other hand, as illustrated in the introduction, we don't want to state that all exception cases are false, as, given $\mathcal{P}^1_{fly}$, we must do for

$$\mathcal{O} = \{can\_fly(jerry)\}.$$

Consider the following definition for restricted dependency in contextual programs: Given a clause $A \leftarrow L_1 \wedge \ldots \wedge L_m \wedge \mathsf{ctxt}(L_{m+1}) \wedge \ldots \wedge \mathsf{ctxt}(L_{m+p})$ for all i, $1 \leq i \leq m$, A 'strongly depends on' L_i. The 'strongly depends on' relation is transitive. If A strongly depends on L_i, then $\neg A$ strongly depends on L_i. Furthermore, if $L_i = B$, then A strongly depends on $\neg B$ and if $L_i = \neg B$, then A strongly depends on B. Example 4.7 clarifies this notion of dependency.

A contextual abductive framework is a quadruple $\langle \mathcal{P}, \mathcal{A}, \mathcal{IC}, \models_{wcs} \rangle$, consisting of an acyclic contextual program $\mathcal{P}$, a set of abducibles $\mathcal{A}$, a set of integrity constraints $\mathcal{IC}$, and the entailment relation $\models_{wcs}$, where $\mathcal{P} \models_{wcs} F$ if and only if $\mathcal{P}$ is acyclic and

$\mathsf{lfp}\ \Phi_{\mathcal{P}}(F) = \top$. The set of abducibles is defined as in Section 2.5:

$$\mathcal{A}_{\mathcal{P}} \quad = \quad \{A \leftarrow \top \mid A \in \mathsf{undef}(\mathcal{P})\} \ \cup \ \{A \leftarrow \bot \mid A \in \mathsf{undef}(\mathcal{P})\}.$$

Let *observation* $\mathcal{O}$ be a non-empty set of ground literals. Note that $\mathcal{O}$ does not contain formulas of the form $\mathsf{ctxt}(L)$.

Definition 4.1. *Given the contextual abductive framework $\langle \mathcal{P}, \mathcal{A}, \mathcal{IC}, \models_{wcs} \rangle$, $\mathcal{E}$ is a contextual explanation of $\mathcal{O}$ given $\mathcal{P}$ and $\mathcal{IC}$ if and only if*

1. *$\mathcal{O}$ is explained by $\mathcal{E}$ given $\mathcal{P}$ and $\mathcal{IC}$*
2. *for all $A \leftarrow \top \in \mathcal{E}$ and for all $A \leftarrow \bot \in \mathcal{E}$ there exists an $L \in \mathcal{O}$, such that L strongly depends on A.*

Note that, compared to explanations, contextual explanations have an additional requirement: There has to be a literal in the observation, which depends on some atom for which there exists a fact or assumption in $\mathcal{E}$. We distinguish between skeptical and credulous reasoning in the usual way:

F *contextually follows skeptically from $\mathcal{P}$, $\mathcal{IC}$ and $\mathcal{O}$*iff $\mathcal{O}$ can be contextually explained given $\mathcal{P}$ and $\mathcal{IC}$, and for all $\mathcal{E}$ for $\mathcal{O}$ it holds that $\mathcal{P} \cup \mathcal{E} \models_{wcs} F$.

F *contextually follows credulously from $\mathcal{P}$, $\mathcal{IC}$ and $\mathcal{O}$* iff there exists an $\mathcal{E}$ that contextually explains $\mathcal{O}$ and it holds that $\mathcal{P} \cup \mathcal{E} \models_{wcs} F$.

Whereas in [Pereira and Pinto, 2011], inspection points are meta-predicates for which a meta-abduction procedure is required, here, ctxt is part of the logic, with which we can naturally extend abduction. Pereira and Pinto's intuition of the concepts of *consumers* and *producers* is still assured as Example 4.8 shows: Given the first clause in $\mathcal{P}$, q cannot be produced, i.e. when (only) p is observed, then we cannot abduce $\{q \leftarrow \top\}$ as a contextual explanation. On the other hand, given the third clause of $\mathcal{P}$, q can be produced, i.e. when t is observed, then we can abduce the contextual explanation $\mathcal{E} = \{q \leftarrow \top\}$. This explanation in turn can be consumed by $\mathsf{ctxt}(q)$ in the first clause, i.e. $\mathsf{ctxt}(q)$ is true given $\mathcal{E}$, and thus p will be true as well.

4.4. Tweety and Jerry

Let us adapt $\mathcal{P}^2_{fly}$ from the introduction such that all exceptions, viz. X being a penguin or a kiwi, are evaluated with respect to their context, ctxt, instead. The new program

Example 4.8. Consider the program $\mathcal{P}$ consisting of the following clauses:

$$\begin{array}{rcl} p & \leftarrow & \mathsf{ctxt}(q). \\ p & \leftarrow & r. \\ t & \leftarrow & q. \end{array}$$

Consider the following two observations $\mathcal{O}_p = \{p\}$ and $\mathcal{O}_{p,t} = \{p, t\}$.
$\mathcal{A}_\mathcal{P}$ consists of the following four clauses:

$$\begin{array}{rcl} q & \leftarrow & \top. \\ q & \leftarrow & \bot. \\ r & \leftarrow & \top. \\ r & \leftarrow & \bot. \end{array}$$

$\mathcal{O}_p$ can only be contextually explained by $\mathcal{E}_1 = \{r \leftarrow \top\}$ but not by $\{\mathsf{ctxt}(q) \leftarrow \top\}$ because $\mathsf{ctxt}(q) \leftarrow \top$ is not in $\mathcal{A}_\mathcal{P}$. Furthermore, $\mathcal{O}_p$ cannot be contextually explained by $q \leftarrow \top$ because p does not depend on q! On the other hand, $\mathcal{O}_{p,t}$ has only the following (minimal) explanation:

$$\mathcal{E}_2 = \{q \leftarrow \top\}.$$

$\mathcal{E}_2$ is a valid contextual explanation for $p \in \mathcal{O}_{p,t}$, as $\mathsf{ctxt}(q)$ is true because of $q \leftarrow \top$. $q \leftarrow \top$ is allowed to be in $\mathcal{E}_2$ because $t \in \mathcal{O}_{p,t}$ depends on q.

$\mathcal{P}^3_{fly}$ consists of the following clauses:

$$\begin{array}{rcl} \mathit{can\_fly}(X) & \leftarrow & \mathit{bird}(X) \wedge \neg \mathit{ab}_1(X). \\ \mathit{ab}_1(X) & \leftarrow & \mathsf{ctxt}(\mathit{kiwi}(X)). \\ \mathit{ab}_1(X) & \leftarrow & \mathsf{ctxt}(\mathit{penguin}(X)). \\ \mathit{bird}(\mathit{tweety}) & \leftarrow & \top. \\ \mathit{bird}(\mathit{jerry}) & \leftarrow & \top. \end{array}$$

The least model of the weak completion of $\mathcal{P}^3_{fly} = \langle I^\top, I^\bot \rangle$ is

$$\begin{array}{rcl} I^\top & = & \{\mathit{bird}(\mathit{tweety}), \mathit{bird}(\mathit{jerry}), \mathit{can\_fly}(\mathit{tweety}), \mathit{can\_fly}(\mathit{jerry})\}, \\ I^\bot & = & \{\mathit{ab}_1(\mathit{tweety}), \mathit{ab}_1(\mathit{jerry})\}. \end{array}$$

This model already entails the observation $\mathcal{O} = \{\mathit{can\_fly}(\mathit{jerry})\}$ and, thus, $\mathcal{O}$ does not need any explanation beyond the minimal empty one.

4.4.1. More about Tweety

Consider the following additional (very simplified) information about birds: *Usually birds with feathers like hair are kiwis. Usually birds that are black and white are penguins. Usually, kiwis and penguins don't live in Europe.* We encode this information by extending $\mathcal{P}^3_{fly}$ with the following clauses:

$$\begin{array}{rcl} \mathcal{P}^4_{fly} = \mathcal{P}^3_{fly} \cup \{ \quad \mathit{kiwi}(X) & \leftarrow & \mathit{bird}(X) \wedge \mathit{featherslikeHair}(X) \wedge \neg \mathit{ab}_2(X), \\ \mathit{penguin}(X) & \leftarrow & \mathit{bird}(X) \wedge \mathit{blackAndWhite}(X) \wedge \neg \mathit{ab}_3(X), \\ \mathit{ab}_2(X) & \leftarrow & \mathsf{ctxt}(\mathit{inEurope}(X)) \wedge \mathit{ab}_4(X), \\ \mathit{ab}_3(X) & \leftarrow & \mathsf{ctxt}(\mathit{inEurope}(X)) \wedge \mathit{ab}_5(X), \\ \mathit{ab}_4(X) & \leftarrow & \bot, \\ \mathit{ab}_5(X) & \leftarrow & \bot \quad \}. \end{array}$$

The least model of the weak completion of $\mathcal{P}^4_{fly} = \langle I^\top, I^\bot \rangle$ is

$$\begin{array}{rcl} I^\top & = & \{\mathit{bird}(\mathit{tweety}), \mathit{bird}(\mathit{jerry}), \mathit{can\_fly}(\mathit{jerry}), \mathit{can\_fly}(\mathit{tweety})\}, \\ I^\bot & = & \{\mathit{ab}_i(\mathit{tweety}) \mid 1 \leq i \leq 5\} \cup \{\mathit{ab}_i(\mathit{jerry}) \mid 1 \leq i \leq 5\}. \end{array}$$

We observe that Tweety cannot fly and that Tweety has feathers like hair:

$$\mathcal{O}_t = \{\neg \mathit{can\_fly}(\mathit{tweety}),\ \mathit{featherslikeHair}(\mathit{tweety})\}.$$

The set of abducibles, $\mathcal{A}_{\mathcal{P}^4_{fly}}$, is:

$$\begin{array}{rcl@{\quad}rcl}
\mathcal{A}_{\mathcal{P}^2_{fly}} \cup \{ \quad featherslikeHair(jerry) & \leftarrow & \top, & featherslikeHair(jerry) & \leftarrow & \bot, \\
featherslikeHair(tweety) & \leftarrow & \top, & featherslikeHair(tweety) & \leftarrow & \bot, \\
blackAndWhite(jerry) & \leftarrow & \top, & blackAndWhite(jerry) & \leftarrow & \bot, \\
blackAndWhite(tweety) & \leftarrow & \top, & blackAndWhite(tweety) & \leftarrow & \bot, \\
inEurope(jerry) & \leftarrow & \top, & inEurope(jerry) & \leftarrow & \bot, \\
inEurope(tweety) & \leftarrow & \top, & inEurope(tweety) & \leftarrow & \bot \quad \}.
\end{array}$$

$\mathcal{E}_t = \{featherslikeHair(tweety) \leftarrow \top\}$ is the only (minimal) contextual explanation for $\mathcal{O}_t$. The least model of the weak completion of $\mathcal{P}^4_{fly} \cup \mathcal{E}_t$, $\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}^4_{fly} \cup \mathcal{E}_t) = \langle I^\top, I^\bot \rangle$ is

$$\begin{array}{rl}
I^\top & = \{\, bird(tweety), featherslikeHair(tweety), kiwi(tweety), ab_1(tweety), \\
& \quad bird(jerry), can_fly(jerry) \,\}, \\
I^\bot & = \{\, can_fly(tweety)\} \cup \{ab_i(tweety) \mid 2 \leq i \leq 5\,\} \cup \{ab_i(jerry) \mid 1 \leq i \leq 5\,\}.
\end{array}$$

From this model we derive that Tweety is a kiwi, even though $inEurope(tweety)$ is unknown. This is what we assume humans do while reasoning: They do not need to assume anything about Tweety (not) living in Europe, as the information $featherslikeHair(tweety)$ is enough to conclude that Tweety is a kiwi.

Let us consider again the concepts of consumers and producers here: The observation $\neg can\_fly(tweety)$ alone could not have been contextually explained by $featherslikeHair(tweety)$ because by the ctxt in the clause

$$ab_1(tweety) \leftarrow \mathsf{ctxt}(kiwi(tweety)) \in g\mathcal{P}^4_{fly},$$

$kiwi(tweety)$ cannot be produced, but only consumed. We will consider this example again in Section 4.5.1.

4.4.2. More about Jerry

Consider again $\mathcal{P}^4_{fly}$ from the previous example together with the observation that Jerry flies and Jerry lives in Europe: $\mathcal{O}_j = \{can\_fly(jerry), inEurope(jerry)\}$.
$\mathcal{A}_{\mathcal{P}^4_{fly}}$ is defined in the previous example. The only contextual explanation for $\mathcal{O}_j$ is $\mathcal{E}_j = \{inEurope(jerry) \leftarrow \top\}$. The least model of the weak completion of $\mathcal{P}^4_{fly} \cup \mathcal{E}_j$, $\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}^4_{fly} \cup \mathcal{E}_j) = \langle I^\top, I^\bot \rangle$ is

$$\begin{array}{rcl}
I^\top & = & \{\, inEurope(jerry), bird(jerry), can_fly(jerry), ab_2(jerry), ab_3(jerry), \\
& & \quad bird(tweety), can_fly(tweety) \,\}, \\
I^\bot & = & \{\, kiwi(jerry), penguin(jerry), ab_1(jerry), ab_4(jerry), ab_5(jerry)\} \\
& \cup & \{\, ab_i(tweety) \mid 1 \leq i \leq 5\,\}.
\end{array}$$

From this model, we can derive that Jerry is not a penguin and Jerry is not a kiwi!

4.5. Contextual Side-effects and Consequences

In the following, we will formalize the notions of side-effects and consequences within contextual abduction and provide definitions and examples to clarify how contextual abduction enriches the expressiveness of abduction.

4.5.1. Contextual Side-effects

The following definition captures the idea of contextual side-effects:

Definition 4.2. *Given a program $\mathcal{P}$ and a set of integrity constraints $\mathcal{IC}$, let $\mathcal{O}_1$ and $\mathcal{O}_2$ be two observations and $\mathcal{E}_1$ be a contextual explanation for $\mathcal{O}_1$.*

*$\mathcal{O}_2$ is a **necessary contextual side-effect of** $\mathcal{O}_1$ **given** $\mathcal{P}$ **and** $\mathcal{IC}$ iff $\mathcal{O}_2$ cannot be contextually explained but $\mathcal{O}_1 \cup \mathcal{O}_2$ is contextually explained by $\mathcal{E}_1$.*

*$\mathcal{O}_2$ is a **possible contextual side-effect of** $\mathcal{O}_1$ **given** $\mathcal{P}$ **and** $\mathcal{IC}$ iff $\mathcal{O}_2$ cannot be contextually explained by $\mathcal{E}_1$ but $\mathcal{O}_1 \cup \mathcal{O}_2$ is contextually explained by $\mathcal{E}_1$.*

The notion behind contextual side-effects is that every explanation $\mathcal{E}_1$ for $\mathcal{O}_1$ gives us an explanation for $\mathcal{O}_2$. Note that a necessary contextual side-effect is also a possible contextual side-effect.

Consider again the Tweety example in Section 4.4.1, where, given the program $\mathcal{P}^3_{fly}$ and the observation $\mathcal{O}_t = \{\neg \mathit{can\_fly}(\mathit{tweety}), \mathit{featherslikeHair}(\mathit{tweety})\}$, the only contextual explanation for $\mathcal{O}_t$ is $\mathcal{E}_t = \{\mathit{featherslikeHair}(\mathit{tweety}) \leftarrow \top\}$. Consider now the observation $\mathcal{O}' = \{\neg \mathit{can\_fly}(\mathit{tweety})\} \subset \mathcal{O}_t$: $\mathcal{O}'$ cannot be contextually explained by $\mathcal{E}_t$, as $\mathit{can\_fly}(\mathit{tweety})$ does not strongly depend on $\mathit{featherslikeHair}(\mathit{tweety})$. According to Definition 4.2, $\mathcal{O}'$ is a necessary contextual side-effect of

$$\mathcal{O}_t \setminus \mathcal{O}' = \{\mathit{featherslikeHair}(\mathit{tweety})\}.$$

On the other hand, consider again the Jerry example in Section 4.4.2, where, for the observation $\mathcal{O}_j = \{\mathit{can\_fly}(\mathit{jerry}), \mathit{inEurope}(\mathit{jerry})\}$, the only contextual explanation is $\mathcal{E}_j = \{\mathit{inEurope}(\mathit{jerry}) \leftarrow \top\}$. As $\mathcal{O}' = \{\mathit{can\_fly}(\mathit{jerry})\}$ already follows from the empty explanation, $\mathcal{E}' = \emptyset$, $\mathcal{O}'$ cannot be considered a contextual side-effect of

$$\mathcal{O}_j \setminus \mathcal{O}' = \{\mathit{inEurope}(\mathit{jerry})\}.$$

$\mathcal{O}'' = \{\neg \mathit{penguin}(\mathit{jerry})\}$ is a possible contextual side-effect of $\mathcal{O}_j$, as $\mathcal{O}''$ cannot be contextually explained by $\mathcal{E}_j$. Note that $\mathcal{O}''$ can also be contextually explained by

$$\mathcal{E}'' = \{\mathit{blackAndWhite}(\mathit{jerry}) \leftarrow \bot\}.$$

4.5.2. Contestable Contextual Side-effects

Analogously to Definition 4.2, its counterpart, contestable contextual side-effects, is defined as follows:

Definition 4.3. *Given a contextual program $\mathcal{P}$ and a set of integrity constraints $\mathcal{IC}$. Let $\mathcal{O}_1$ and $\mathcal{O}_2$ be two observations and $\mathcal{E}_1$ be a contextual explanation for $\mathcal{O}_1$. The negation of the observation $\mathcal{O}_2$ is $\neg\mathcal{O}_2$, where $\neg\mathcal{O}_2 = \{\neg L \mid L \in \mathcal{O}_2\}$.*

*$\mathcal{O}_2$ is a **necessarily contested contextual side-effect of** $\mathcal{O}_1$ **given** $\mathcal{P}$ **and** $\mathcal{IC}$ iff $\neg\mathcal{O}_2$ cannot be contextually explained by $\mathcal{E}_1$ but $\mathcal{O}_1 \cup \neg\mathcal{O}_2$ can be contextually explained by $\mathcal{E}_1$.*

*$\mathcal{O}_2$ is a **possibly contested contextual side-effect of** $\mathcal{O}_1$ **given** $\mathcal{P}$ **and** $\mathcal{IC}$ iff $\neg\mathcal{O}_2$ cannot be contextually explained by $\mathcal{E}_1$ but $\mathcal{O}_1 \cup \neg\mathcal{O}_2$ can be contextually explained by $\mathcal{E}_1$.*

Reconsider the examples of the previous subsection: It is easy to see that given $\mathcal{O}'_\neg = \{can\_fly(tweety)\}$, $\mathcal{O}'_\neg$ is a necessary contested contextual side-effect of $\mathcal{O}_t$. Analogously, given that $\mathcal{O}''_\neg = \{penguin(jerry)\}$, $\mathcal{O}''_\neg$ is a possible contested contextual side-effect of $\mathcal{O}_j$.

4.5.3. Contextual Relevant Consequences

We define two notions of contextual relevant consequences as follows:

Definition 4.4. *Given a contextual program $\mathcal{P}$ and a set of integrity constraints $\mathcal{IC}$. Let $\mathcal{O}_1$ and $\mathcal{O}_2$ be two observations and $\mathcal{E}_1$ be a contextual explanation for $\mathcal{O}_1$.*

*$\mathcal{O}_2$ is a **necessary contextual relevant consequence of** $\mathcal{O}_1$ **given** $\mathcal{P}$ **and** $\mathcal{IC}$ iff $\mathcal{O}_2$ cannot be contextually explained by $\mathcal{E}_1$ but $\mathcal{O}_1 \cup \mathcal{O}_2$ can be contextually explained by $\mathcal{E}_2$, where $\mathcal{E}_1 \subset \mathcal{E}_2$.*

*$\mathcal{O}_2$ is a **possible contextual relevant consequence of** $\mathcal{O}_1$ **given** $\mathcal{P}$ **and** $\mathcal{IC}$ iff $\mathcal{O}_2$ cannot be contextually explained by $\mathcal{E}_1$ but $\mathcal{O}_1 \cup \mathcal{O}_2$ can be contextually explained by $\mathcal{E}_2$, where $\mathcal{E}_1 \subset \mathcal{E}_2$.*

Furthermore, it might be the case that two observations contain contextual relevant consequences of each other, simultaneously, i.e. they are mutually plausibly explainable together, but not each by itself. This notion is stronger than Definition 4.4:

Definition 4.5. *Given a program $\mathcal{P}$ and a set of integrity constraints $\mathcal{IC}$. Let $\mathcal{O}_1$, $\mathcal{O}_2$ be observations.*

*$\mathcal{O}_1$ and $\mathcal{O}_2$ are **necessarily jointly supported contextual relevant consequences given $\mathcal{P}$ and $\mathcal{IC}$** iff $\mathcal{O}_1$ is a necessary contextual relevant consequence of $\mathcal{O}_2$ and $\mathcal{O}_2$ is a necessary contextual relevant consequence of $\mathcal{O}_1$.*

*$\mathcal{O}_1$ and $\mathcal{O}_2$ are **possibly jointly supported contextual relevant consequences given $\mathcal{P}$ and $\mathcal{IC}$** iff $\mathcal{O}_1$ is a possible contextual relevant consequence of $\mathcal{O}_2$ and $\mathcal{O}_2$ is a possible contextual relevant consequence of $\mathcal{O}_1$.*

Consider the contextual program $\mathcal{P}^{fire}$ consisting of the following two clauses:

$$\begin{array}{rcl} \mathit{smoke} & \leftarrow & \mathit{fire} \wedge \mathsf{ctxt}(\mathit{firefighters}). \\ \mathit{sirens} & \leftarrow & \mathsf{ctxt}(\mathit{fire}) \wedge \mathit{firefighters}. \end{array}$$

Let us observe

$$\mathcal{O}_{smoke} \quad = \quad \{\mathit{smoke}\}.$$

We can abduce $\mathit{fire} \leftarrow \top$ but not $\mathit{firefighters} \leftarrow \top$, because *smoke* does not depend on *firefighters*. On the other hand, by observing

$$\mathcal{O}_{sirens} \quad = \quad \{\mathit{sirens}\},$$

we can abduce *firefighters* but not *fire*. However, if we observe both, *smoke* and *sirens*, *fire* can be abduced by $\mathcal{O}_{smoke}$ because *smoke* depends on *fire* and *firefighters* can be abduced by $\mathcal{O}_{sirens}$ because *sirens* depends on *firefighters*. Accordingly, the explanation for $\mathcal{O}_{smoke} \cup \mathcal{O}_{sirens}$ is

$$\mathcal{E} \quad = \quad \{\mathit{firefighters} \leftarrow \top, \mathit{fire} \leftarrow \top\}.$$

$\mathcal{O}_{smoke}$ and $\mathcal{O}_{sir}$ are necessarily jointly supported contextual relevant consequences given $\mathcal{P}^{fire}$ and $\mathcal{IC}_{fire}$.

4.6. Conclusion

Motivated by the famous Tweety example, we first show that the Weak Completion Semantics does not yield the desired results. We would like to avoid having to abductively consider all exception cases and to automatically prefer normal explanations to those explanations specifying such exception cases. To do so, we set forth contextual programs, for the purpose of which we introduce ctxt, a new truth-functional operator, which turns out to fit quite well with the interpretation of negation as failure under three-valued semantics. Unfortunately, the $\Phi_{\mathcal{P}}$ operator is not monotonic with respect to these contextual programs anymore. Even worse, the $\Phi_{\mathcal{P}}$ operator might not even have a least fixed point for some contextual programs. Nevertheless, we can show that the

$\Phi_{\mathcal{P}}$ operator does always have a least fixed point if we restrict contextual programs to the class of acyclic ones and introduce the concept of contextual abduction, and model the Tweety example as desired. In the last part of this chapter, we specify the relations between observations and explanations under contextual abduction, allowing us to define notions with regard to contextual side-effects, contestable contextual side-effects and contextual relevant consequences. The main advantage of the contextual reasoning approach here over the approach presented in [Pereira and Pinto, 2011] is that the ctxt operator is part of the logic, whereas in [Pereira and Pinto, 2011] in order to evaluate the inspection points, a meta-abduction transformation procedure is required.

Some open questions are left to be investigated in the future. For instance, can the requirements for the classes of acyclic contextual programs be relaxed to those that are only acyclic with respect to the truth functional operator ctxt, so that the $\Phi_{\mathcal{P}}$ operator is still guaranteed to yield a least fixed point? Furthermore, as the Weak Completion Semantics seems to adequately model human reasoning, a natural question to ask is whether the assumptions made for the development of contextual reasoning fit with the findings from Cognitive Science? For this purpose, we are particularly interested in psychological experiments that deal with context sensitive information.

Part II.

Human Reasoning Tasks

5. Byrne's Suppression Task and Wason's Selection Task

In this chapter, we first discuss the formalization of Byrne's [1989] suppression task, which we have briefly explained in the introduction. The first part has originally been presented in [Hölldobler and Kencana Ramli, 2009a,b] and the second part has originally been presented in [Hölldobler, Philipp, and Wernhard, 2011]. The second part, Section 5.2, presents a formalization of Wason's [1968] selection task and Griggs and Cox [1982] isomorphic representation of this task in a social context.[1]

5.1. Byrne's Suppression Task

We have briefly presented the layout of the task in the introduction, which will now be discussed in detail. Byrne's suppression task consists of two parts, where the results of the first part are achieved by forward reasoning, whereas the results of the second part are achieved by backward reasoning. Sections 5.1.1 and 5.1.2 explain Stenning and van Lambalgen's two step approach and their techniques by means of the first part of Byrne's suppression task. Furthermore, we show the cases, where their approach fails and why it requires the Weak Completion Semantics. These techniques are analogously applied to the second part of Byrne's suppression task, which is presented in Section 5.1.3.

5.1.1. Representation as Logic Programs

Stenning and van Lambalgen's first step of formalizing human reasoning, reasoning towards an appropriate representation, deals with conceptual cognitive adequacy, already discussed in the introduction. In particular, Stenning and van Lambalgen argue that conditionals shall not be encoded by inferences straight away, but rather by *licenses for inference*. Consider again the simple conditional from the introduction: '*if she has an essay to write (e), then she will study late in the library (l)*' should be encoded by the clause $l \leftarrow e \wedge \neg ab_1$, where ab_1 is an *abnormality predicate* and true if something

[1] Section 5.1.4 has been published in [Dietz, Hölldobler, and Wernhard, 2014]. Section 5.3 has been published in [Dietz, Hölldobler, and Ragni, 2012a]. The original idea of Section 5.2 has first been published in [Dietz, Hölldobler, and Ragni, 2012b] and an improved version has been published in [Dietz, Hölldobler, and Ragni, 2013]

		Conditionals				Facts and Assumptions
$\mathcal{P}_{e}$	=	$\{l \leftarrow e \wedge \neg ab_1,$		$ab_1 \leftarrow \bot,$		$e \leftarrow \top\}$
$\mathcal{P}_{e+Alt}$	=	$\{l \leftarrow e \wedge \neg ab_1,$	$l \leftarrow t \wedge \neg ab_2,$	$ab_1 \leftarrow \bot,$	$ab_2 \leftarrow \bot,$	$e \leftarrow \top\}$
$\mathcal{P}_{e+Add}$	=	$\{l \leftarrow e \wedge \neg ab_1,$	$l \leftarrow o \wedge \neg ab_3,$	$ab_1 \leftarrow \neg o,$	$ab_3 \leftarrow \neg e,$	$e \leftarrow \top\}$
$\mathcal{P}_{\neg e}$	=	$\{l \leftarrow e \wedge \neg ab_1,$		$ab_1 \leftarrow \bot,$		$e \leftarrow \bot\}$
$\mathcal{P}_{\neg e+Alt}$	=	$\{l \leftarrow e \wedge \neg ab_1,$	$l \leftarrow t \wedge \neg ab_2,$	$ab_1 \leftarrow \bot,$	$ab_2 \leftarrow \bot,$	$e \leftarrow \bot\}$
$\mathcal{P}_{\neg e+Add}$	=	$\{l \leftarrow e \wedge \neg ab_1,$	$l \leftarrow o \wedge \neg ab_3,$	$ab_1 \leftarrow \neg o,$	$ab_3 \leftarrow e,$	$e \leftarrow \bot\}$

Table 5.1.: Representational form of the first part of the suppression task according to Stenning and van Lambalgen [2008].

abnormal is known. In other words, l holds if e is true and nothing abnormal is known ($\neg ab_1$), i.e. everything abnormal is false.

Table 5.1 shows the representational form of the first part of the suppression task as modeled by Stenning and van Lambalgen. In the first three cases, in addition to the conditionals, the participants had to draw conclusions based on the fact that '*she has an essay to write*' ($e \leftarrow \top$). In the last three cases, they had to draw conclusions based on the assumption that '*she does not have an essay to write*' ($e \leftarrow \bot$). The predicates ab_1, ab_2 and ab_3 represent abnormalities with respect to the different conditionals, respectively.

The programs $\mathcal{P}_{e+Alt}$ and $\mathcal{P}_{e+Add}$ contain two clauses with the conclusion l. They differ in the way that the premise of the second clause in in $\mathcal{P}_{e+Alt}$ is an *alternative* to the first clause, whereas in $\mathcal{P}_{e+Add}$ the premise of the second clause is an *addition* to the first clause. The second clause in $\mathcal{P}_{e+Add}$ ($l \leftarrow o \wedge \neg ab_3$) takes effect as an additional precondition for l. This is represented by the clause stating that ab_1 is true when '*the library does not stay open*' ($ab_1 \leftarrow \neg o$) and the clause that states that ab_3 is true when '*she does not have an essay to write*' ($ab_3 \leftarrow \neg e$).

5.1.2. Forward Reasoning

Adopting the programs obtained by Stenning and van Lambalgen as result of the first step of reasoning towards an appropriate representation, we will now focus on the second step, the inferential aspect, which corresponds to inferential cognitive adequacy.

The second column of Table 5.2 shows the weak completion of the programs encoding the first six examples of the suppression task. As already discussed in Chapter 2.3, the weak completion of all programs admits the model intersection property, therefore we

	The weak completion of $\mathcal{P}$	$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}$		Byrne
$\mathsf{wc}\,\mathcal{P}_{e}$	$= \{l \leftrightarrow e \wedge \neg ab_1,\ ab_1 \leftrightarrow \bot,\ e \leftrightarrow \top\}$	$\langle\{e,l\},\{ab_1\}\rangle$	$\models_{wcs} l$	96% L
$\mathsf{wc}\,\mathcal{P}_{e+Alt}$	$= \{l \leftrightarrow (e \wedge \neg ab_1) \vee (t \wedge \neg ab_2),$ $ab_1 \leftrightarrow \bot,\ ab_2 \leftrightarrow \bot,\ e \leftrightarrow \top\}$	$\langle\{e,l\},\{ab_1,ab_2\}\rangle$	$\models_{wcs} l$	96% L
$\mathsf{wc}\,\mathcal{P}_{e+Add}$	$= \{l \leftrightarrow (e \wedge \neg ab_1) \vee (o \wedge \neg ab_3),$ $ab_1 \leftrightarrow \neg o,\ ab_3 \leftrightarrow \neg e,\ e \leftrightarrow \top\}$	$\langle\{e\},\{ab_3\}\rangle$	$\not\models_{wcs} l \vee \neg l$	38% L
$\mathsf{wc}\,\mathcal{P}_{\neg e}$	$= \{l \leftrightarrow e \wedge \neg ab_1,\ ab_1 \leftrightarrow \bot,\ e \leftrightarrow \bot\}$	$\langle\emptyset,\{e,l,ab_1\}\rangle$	$\models_{wcs} \neg l$	46% $\overline{L}$
$\mathsf{wc}\,\mathcal{P}_{\neg e+Alt}$	$= \{l \leftrightarrow (e \wedge \neg ab_1) \vee (t \wedge \neg ab_2),$ $ab_1 \leftrightarrow \bot,\ ab_2 \leftrightarrow \bot,\ e \leftrightarrow \bot\}$	$\langle\emptyset,\{e,ab_1,ab_2\}\rangle$	$\models_{wcs} l \vee \neg l$	4% $\overline{L}$
$\mathsf{wc}\,\mathcal{P}_{\neg e+Add}$	$= \{l \leftrightarrow (e \wedge \neg ab_1) \vee (o \wedge \neg ab_3),$ $ab_1 \leftrightarrow \neg o,\ ab_3 \leftrightarrow \neg e,\ e \leftrightarrow \bot\}$	$\langle\{ab_3\},\{e,l\}\rangle$	$\models_{wcs} \neg l$	63% $\overline{L}$

Table 5.2.: The weak completion and the least models of the corresponding programs and the experimental results. The fourth column shows whether l or $\neg l$ follow from the least models. The information in the last column refers to the experimental results of Byrne [1989].

can reason with respect to their least models. The third column in Table 5.2 depicts the corresponding least models. As the last column of Table 5.2 shows, our approach coincides with the seemingly favored results of the suppression task and thus appears to be inferentially adequate. Consider in Table 5.2 for example $\mathcal{P}_{e+Add}$ and its weak completion. The interpretations $\langle\{e,o\},\{ab_1,ab_3,l\}\rangle$ and $\langle\{e\},\{ab_3\}\rangle$ are both models of $\mathsf{wc}\,\mathcal{P}_{e+Add}$. Given that $I_0 = \langle\emptyset,\emptyset\rangle$, $\Phi_{\mathcal{P}_{e+Add}}$ is computed as follows:

$$\begin{array}{lllllll} I_1 & = & \Phi_{\mathcal{P}_{e+Add}}(I_0) & = & \langle\{e\},\emptyset\rangle, & & \\ I_2 & = & \Phi_{\mathcal{P}_{e+Add}}(I_1) & = & \langle\{e\},\{ab_3\}\rangle & = & \Phi_{\mathcal{P}_{e+Add}}(I_2). \end{array}$$

As shown by Hölldobler and Kencana Ramli [2009a,b], $\langle\{e\},\{ab_3\}\rangle$ is not a model of $\mathcal{P}_{e+Add}$ under SvL-semantics because the clause $l \leftarrow o \wedge ab_3 \in \mathcal{P}_{e+Add}$ is mapped to U under SvL-semantics and not to $\top$ as under Łukasiewicz semantics. This is a counter-example to Lemma 4 (1.) in [Stenning and van Lambalgen, 2008, p. 194f], which states that the least fixed point of the $\Phi_{\mathcal{P}}$ operator under SvL-semantics is the (knowledge-) least model of $\mathcal{P}$. Furthermore, Stenning and van Lambalgen [2008] claim in Lemma 4 (3.) that all models of the completion of $\mathcal{P}$ are fixed points of $\Phi_{\mathcal{P}}$ and every fixed point is a model. The following example shows that both claims are not true. Consider the completion of $\mathcal{P}_{\neg e+Alt}$, i.e.

$$\{l \leftrightarrow (e \wedge \neg ab_1) \vee (t \wedge \neg ab_2),\ ab_1 \leftrightarrow \bot,\ ab_2 \leftrightarrow \bot,\ e \leftrightarrow \bot,\ t \leftrightarrow \bot\}.$$

	Conditionals			$\mathcal{O}$	Explanations
$\mathcal{P}$	$= \{l \leftarrow e \wedge \neg ab_1,$	$ab_1 \leftarrow \bot\}$		$\{l\}$	$\{e \leftarrow \top\}$
$\mathcal{P}_{Alt}$	$= \{l \leftarrow e \wedge \neg ab_1, l \leftarrow t \wedge \neg ab_2,$	$ab_1 \leftarrow \bot,$	$ab_2 \leftarrow \bot\}$	$\{l\}$	$\{e \leftarrow \top\}, \{t \leftarrow \top\}$
$\mathcal{P}_{Add}$	$= \{l \leftarrow e \wedge \neg ab_1, l \leftarrow o \wedge \neg ab_3,$	$ab_1 \leftarrow \neg o,$	$ab_3 \leftarrow \neg e\}$	$\{l\}$	$\{e \leftarrow \top,\ o \leftarrow \top\}$
$\mathcal{P}$	$= \{l \leftarrow e \wedge \neg ab_1,$	$ab_1 \leftarrow \bot\}$		$\{\neg l\}$	$\{e \leftarrow \bot\}$
$\mathcal{P}_{Alt}$	$= \{l \leftarrow e \wedge \neg ab_1, l \leftarrow t \wedge \neg ab_2,$	$ab_1 \leftarrow \bot,$	$ab_2 \leftarrow \bot\}$	$\{\neg l\}$	$\{e \leftarrow \bot,\ t \leftarrow \bot\}$
$\mathcal{P}_{Add}$	$= \{l \leftarrow e \wedge \neg ab_1, l \leftarrow o \wedge \neg ab_3,$	$ab_1 \leftarrow \neg o$	$ab_3 \leftarrow e\}$	$\{\neg l\}$	$\{e \leftarrow \bot\}, \{o \leftarrow \bot\}$

Table 5.3.: The Representational form of the second part of the suppression task according to Stenning and van Lambalgen [2008] and Hölldobler, Philipp, and Wernhard [2011].

Both, t and e are mapped to $\bot$ and, consequently, l is mapped to $\bot$ as well. However, as pointed out by Hölldobler and Kencana Ramli [2009a,b], the least fixed point of $\Phi_{\mathcal{P}_{\neg e+Alt}}$ is $\langle \emptyset, \{e, ab_1, ab_2\}\rangle$, where t and l are unknown, and is not a model of the completion of $\mathcal{P}_{\neg e+Alt}$. This example also shows that reasoning under SvL-semantics with respect to the completion of a program is not adequate, as only 4% of the subjects conclude $\neg l$ in this case.

5.1.3. Backward Reasoning

The second part of the suppression task can best be described as abductive, that is, a plausible explanation is computed given some observation. This notion of abductive consequence with respect to least models of the weak completion has been elaborated by Hölldobler, Philipp, and Wernhard [2011] to model the backward reasoning cases of the suppression task. Table 5.3 shows the representational form of these instances, including the observations and their respective explanations. In the first three cases, additionally to the conditionals, the participants had to draw conclusions based on the fact that '*she goes to the library.*' In the last three cases, they had to draw conclusions based on the assumption that '*she does not go to the library.*' We consider an abductive framework as introduced in Chapter 2.5, consisting of a program $\mathcal{P}$ as knowledge base, a set $\mathcal{A}$ of abducibles consisting of the facts and assumptions for each undefined atom in $\mathcal{P}$,[2] and the logical consequence relation $\models_{wcs}$. As observations we consider a set of literals. For instance, consider the following two programs under skeptical reasoning:

1. $\mathcal{P}_{Alt}$ where $\mathcal{O} = \{l\}$: $\mathcal{A} = \{e \leftarrow \top, e \leftarrow \bot, t \leftarrow \top, t \leftarrow \bot\}$ and $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{Alt} = \langle \emptyset, \{ab_1, ab_2\}\rangle$. There are two explanations with either $\{e \leftarrow \top\}$ or $\{t \leftarrow \top\}$. Accordingly, we cannot conclude that '*she has an essay to finish*'.

[2]Recall that A is *undefined* in $\mathcal{P}$ iff $\mathcal{P}$ does not contain a clause of the form $A \leftarrow body$.

$\mathcal{O}$	The least model of the weak completion of $\mathcal{P} \cup \mathcal{E}$				Byrne
l	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P} \cup \{e \leftarrow \top\})$	$=$	$\langle\{e, l\}, \{ab_1\}\rangle$	$\models^s_{wcs} e$	53% E
l	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}_{Alt} \cup \{e \leftarrow \top\})$	$=$	$\langle\{e, l\}, \{ab_1, ab_2\}\rangle$	$\not\models^s_{wcs} e \vee \neg e$	16% E
l	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}_{Alt} \cup \{t \leftarrow \top\})$	$=$	$\langle\{l, t\}, \{ab_1, ab_2\}\rangle$		
l	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}_{Add} \cup \{e \leftarrow \top, t \leftarrow \top\})$	$=$	$\langle\{e, l, o\}, \{ab_1, ab_3\}\rangle$	$\models^s_{wcs} e$	55% E
$\neg l$	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P} \cup \{e \leftarrow \bot\})$	$=$	$\langle\emptyset, \{e, l, ab_1\}\rangle$	$\models^s_{wcs} \neg e$	69% $\overline{E}$
$\neg l$	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}_{Alt} \cup \{e \leftarrow \bot, t \leftarrow \bot\})$	$=$	$\langle\emptyset, \{e, l, t, ab_1, ab_2\}\rangle$	$\models^s_{wcs} \neg e$	69% $\overline{E}$
$\neg l$	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}_{Add} \cup \{e \leftarrow \bot\})$	$=$	$\langle\{ab_3\}, \{e, l\}\rangle$	$\not\models^s_{wcs} e \vee \neg e$	44% $\overline{E}$
$\neg l$	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}_{Add} \cup \{o \leftarrow \bot\})$	$=$	$\langle\{ab_1\}, \{l, o\}\rangle$		

Table 5.4.: The least models of the weak completion of the corresponding programs together with the explanations for $\mathcal{O}$ and the experimental results. The fourth column shows whether e or $\neg e$ follow from the least models. The cases $\mathcal{P} = \mathcal{P}_{Alt}, \mathcal{O} = \{l\}$ and $\mathcal{P} = \mathcal{P}_{Add}, \mathcal{O} = \{\neg l\}$ have two explanations. The last column shows the experimental results of Byrne [1989].

2. $\mathcal{P}_{Add}$ where $\mathcal{O} = \{l\}$: $\mathcal{A} = \{e \leftarrow \top, e \leftarrow \bot, o \leftarrow \top, o \leftarrow \bot\}$ and $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{Add} = \langle\emptyset, \emptyset\rangle$. There is only one explanation $\{e \leftarrow \top, o \leftarrow \top\}$. Accordingly, we can conclude that '*she has an essay to finish.*'

Table 5.4 depicts results of the second part of the suppression task, which are adequate answers if compared to the seemingly favored results of the suppression task. One should observe that credulously we would conclude e from $\mathcal{P} = \mathcal{P}_{Alt}$ and $\mathcal{O} = \{l\}$, which according to Byrne only 16% of the subjects did.

5.1.4. Well-founded Semantics Revisted

We show the results obtained with the Weak Completion Semantics and the Well-founded Semantics for the program representations in Table 5.1 and Table 5.3. We define $\mathcal{A}\mathsf{t}' = \{e, l, o, t, ab_1, ab_2, ab_3\}$ and for the well-founded models $\mathcal{P}^+$ we assume the models with respect to $\mathcal{A}\mathsf{t} = \mathcal{A}\mathsf{t}'$. For the least models of the weak completion of $\mathcal{P}$ and the well-founded models of $\mathcal{P}^{\mathsf{mod}}$ we assume the models with respect to $\mathcal{A}\mathsf{t} = \mathcal{A}\mathsf{t}' \cup \{A' \mid A \in \mathsf{undef}(\mathcal{P})\}$. Table 5.5 shows the least models of the weak completion and the well-founded models from the first part of the suppression task. Note that for the well-founded model only normal logic programs ($\mathcal{P}^+$) are considered. Obviously there are differences between both semantics with respect to the least models. For instance, for $\mathcal{P}_{e+Add}$

$\mathcal{P}$	$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}/\mathsf{wfm}\ \mathcal{P}^{\mathsf{mod}}$	$\mathsf{wfm}\ \mathcal{P}^{+}$	Byrne
$\mathcal{P}_{e}$	$\langle\{e,l\},\{ab_1\}\rangle$	$\langle\{e,l\},\{\ o,t,\ ab_1,\ ab_2,ab_3\ \}\rangle$	96% L
$\mathcal{P}_{e+Alt}$	$\langle\{e,l\},\{ab_1,ab_2\}\rangle$	$\langle\{e,l\},\{\ o,t,\ ab_1,ab_2,\ ab_3\ \}\rangle$	96% L
$\mathcal{P}_{e+Add}$	$\langle\{e\},\{ab_3\}\rangle$	$\langle\{e,\ ab_1\ \},\{\ l,o,t,ab_2,\ ab_3\}\rangle$	38% L
$\mathcal{P}_{\neg e}$	$\langle\emptyset,\{e,l,ab_1\}\rangle$	$\langle\emptyset,\{e,l,\ o,t,\ ab_1,\ ab_2,ab_3\ \}\rangle$	46% $\neg L$
$\mathcal{P}_{\neg e+Alt}$	$\langle\emptyset,\{e,ab_1,ab_2\}\rangle$	$\langle\emptyset,\{e,\ l,o,t,\ ab_1,ab_2,\ ab_3\ \}\rangle$	4% $\neg L$
$\mathcal{P}_{\neg e+Add}$	$\langle\{ab_3\},\{e,l\}\rangle$	$\langle\{ab_3\ \},\{e,l,\ o,t,ab_1,ab_2\ \}\rangle$	63% $\neg L$

Table 5.5.: The results of the first part of the suppression task. The highlighted atoms show the differences between the least models of the weak completion and the well-founded models.

and $\mathcal{P}_{\neg e+Alt}$, under the Weak Completion Semantics, l is neither in $I^{\top}$ or in $I^{\bot}$, whereas in the well-founded model $l \in I^{\bot}$ in both $\mathcal{P}^{+}_{e+Add}$ and $\mathcal{P}^{+}_{\neg e+Alt}$. This is due to the fact, that undefined atoms such as o in $\mathcal{P}^{+}_{e+Add}$ and t in $\mathcal{P}^{+}_{\neg e+Alt}$ are mapped to false in the well-founded model. Considering Byrne's results, the Well-founded Semantics does not represent the participants' conclusions of suppressing information, whereas the Weak Completion Semantics does.

Table 5.6 shows the results from the second part of the suppression task where abduction is required. In the first three cases, both semantics have the same conclusions about e. In the case of $\mathcal{P}_{l+Alt}$ two explanations are possible, $e \leftarrow \top$ or $t \leftarrow \top$, with two different least models. With skeptical reasoning nothing can be concluded about e, which seems to adequately represent Byrne's findings. Similarly, for $\mathcal{P}_{\neg l+Add}$ with skeptical reasoning nothing can be concluded about e under the Weak Completion, whereas e is true under the Well-founded Semantics. Considering Byrne's results, that 44% of the participants concluded $\neg e$, it is arguable which model adequately represents these results.

5.2. Wason's Selection Task

In Wason's [1968] selection task participants had to check a given conditional statement on some instances. The problem was presented as a rather abstract description and almost all participants' conclusions where invalid with respect to classical logic. Griggs and Cox [1982] developed an isomorphic representation of the problem in a social context, and surprisingly almost all of the participants solved this task classical logic correctly. Kowalski [2011] gives an interesting interpretation of this difference, which we will use as starting point for our formalization.

$\mathcal{P}$	$\mathcal{O}$	$\mathcal{E}$	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}\cup\mathcal{E})/$ $\mathsf{wfm}\,((\mathcal{P}\cup\mathcal{E})^{\mathsf{mod}})$	$\mathsf{wfm}\,(\mathcal{P}\cup\mathcal{E})^{+}$	Byrne	
$\mathcal{P}_{l}$	l	$e \leftarrow \top$	$\langle\{e,l\},\{ab_1\}\rangle$	$\langle\{e,l\},\{\,o,t,\ ab_1,\ ab_2,ab_3\,\}\rangle$	53%	E
$\mathcal{P}_{l+Alt}$	l	$e \leftarrow \top$	$\langle\{e,l\},\{ab_1,ab_2\}\rangle$	$\langle\{e,l\},\{\,o,t,\ ab_1,ab_2,\ ab_3\,\}\rangle$	16%	E
		$t \leftarrow \top$	$\langle\{l,t\},\{ab_1,ab_2\}\rangle$	$\langle\{l,t\},\{\,e,o,\ ab_1,ab_2,\ ab_3\,\}\rangle$		
$\mathcal{P}_{l+Add}$	l	$e \leftarrow \top, o \leftarrow \top$	$\langle\{e,l,o\},\{ab_1,ab_3\}\rangle$	$\langle\{e,l,o\},\{\,t,\ ab_1,\ ab_2,\ ab_3\}\rangle$	55%	E
$\mathcal{P}_{\neg l}$	$\neg l$	$e \leftarrow \bot$	$\langle\emptyset,\{e,l,ab_1\}\rangle$	$\langle\emptyset,\{e,l,\ o,t,\ ab_1,\ ab_2,ab_3\,\}\rangle$	69%	$\neg E$
$\mathcal{P}_{\neg l+Alt}$	$\neg l$	$e \leftarrow \bot, t \leftarrow \bot$	$\langle\emptyset,\{e,l,t,ab_1,ab_2\}\rangle$	$\langle\emptyset,\{e,l,\ o,\ t,ab_1,ab_2,\ ab_3\,\}\rangle$	69%	$\neg E$
$\mathcal{P}_{\neg l+Add}$	$\neg l$	$e \leftarrow \bot$	$\langle\{ab_3\},\{e,l\}\rangle$	$\langle\{\,ab_1,\ ab_3\},\{e,l,\ o,t,ab_2\,\}\rangle$	44%	$\neg E$
		$o \leftarrow \bot$	$\langle\{ab_1\},\{l,o\}\rangle$	$\langle\{ab_1,ab_3\,\},\{\,e,l,o,t,ab_2\,\}\rangle$		

Table 5.6.: The results of the second part of the suppression task. The highlighted atoms show the differences between the least models of the weak completion and the well-founded models.

In the original selection task by Wason [1968], participants were presented the following four cards on a table:

They were told that each card had a letter on one side of the card and a number on the other side of the card. Additionally, the participants had to evaluate the following conditional with respect to each of the four cards:

If there is a D on one side of the card,
then there is 3 *on the other side.*

Which cards must be turned over in order to find out whether the conditional holds?

$$3 \leftarrow D \tag{5.1}$$

The conditional is represented in classical propositional logic as the implication in (5.1), where the propositional variable 3 represents the fact that the number 3 is shown and D represents the fact that the letter D is shown. Then, in order to verify the implication one must turn over the cards showing D and 7. However, the results on the left hand side in Table 5.7, taken from Wason, show that participants believed differently. Whereas 89% of the participants (classical logic) correctly conclude that the card showing D must be turned (a number other than 3 on the other side would falsify the implication), 62%

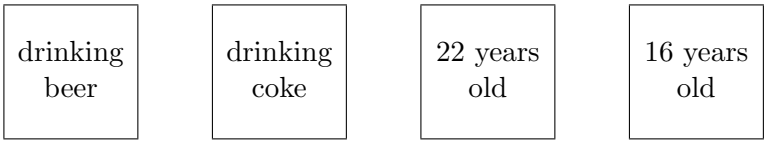
D	F	3	7
89%	16%	62%	25%

beer	coke	22 years old	16 years old
95%	0.025%	0.025%	80%

Table 5.7.: The results of the abstract case (left) and the social case (right) of the selection task. The second row refers to the percentage of participants who stated that the corresponding card needs to be turned.

of the participants incorrectly suggest to turn over the card showing 3 (no relevant information can be found which would falsify the implication). Likewise, whereas 25% of the participants correctly believe that the card showing 7 needs to be turned over (if the other side would show a D, then the implication is falsified), 16% incorrectly believe that the card showing F needs to be turned over (no relevant information can be found which would falsify the implication). In other words, the overall correctness of the answers given for the abstract selection task if modeled by an implication in classical two-valued logic is pretty bad.

Griggs and Cox [1982] adapted Wason's selection task to a social case. Consider the following four cards:



Each card has the person's age on one side of the card and what the person is drinking on the other side of the card. Consider the conditional

If a person is drinking beer,
then the person must be over 19 years of age.

and again the question: Which drinks and persons must be checked to find out whether the conditional holds?

$$o \leftarrow b \tag{5.2}$$

The conditional is represented by the implication in (5.2), where o represents a person being older than 19 years and b represents the person drinking beer. In order to verify the implication one must turn over the cards *drinking beer* and *16 years old*. Participants usually solve the social case of the selection task classical logical correctly. The right hand side of Table 5.7 shows the results presented by Griggs and Cox [1982] for the social case.

Why are the results of these two experiments so different? Several attempts were made to explain these differences. Wason [1968] proposed a *defective truth table* to explain

how humans reason with conditionals. When the antecedent of a conditional is false, then normally people consider the whole conditional as irrelevant and ignore it in further reasoning. Evans [1972] described a phenomenon called the *matching bias*, where people tend to consider only the present values in the conditional. For instance, in the abstract case, card D is the easiest one to solve, because this rule is only true when both values present in the rule are on the card. On the other hand, card 7 is the most difficult one, because people have to make a double mismatch, that is, they have to consider the situation where 3 is not on the card, and therefore, something different than D has to be on the other side. Why do people not make these mistakes in the social case?

One explanation has been given by Kowalski [2011], which we will use to motivate our formalization. He states that people view the conditional in the abstract case as a *belief*. For instance, the participants perceive the task to examine whether the rule is either true or false. On the other hand, in the social case, the participants perceive the rule as a *social constraint*, a conditional that *ought to be* true. People intuitively aim at preventing the violation of such a constraint, which is normally done by observing whether the state of the world complies with the rule.[3] The results presented by Beller and Bender [2012] seem to support this view. They carried out psychological experiments with several variations of the so called abstract deontic selection task. The authors show that the performance of the abstract case can significantly be improved when introducing a deontic notion, even by keeping the abstract formulation. Accordingly, the importance of the context (abstract or social) seems not to be the essential one. Instead, their emphasis is on the deontic value involved in the subjects' interpretations.

As already mentioned in the introduction, Stenning and van Lambalgen distinguish between two steps when modeling human reasoning. We will again implement the conditionals as licenses for inferences. This can be achieved by adding an *abnormality predicate* to the antecedent of the implication. Applying this idea to the Wason selection task we obtain

$$3 \leftarrow D \wedge \neg ab_1 \tag{5.3}$$

instead of (5.1) and

$$o \leftarrow b \wedge \neg ab_2 \tag{5.4}$$

instead of (5.2), where $\neg ab_1$ and $\neg ab_2$ are used to express that the corresponding rules hold unless there are some abnormalities.

5.2.1. Social Case

In this case most humans are quite familiar with the conditional as it is a standard in law. They are also aware – it is common sense knowledge – that there are no exceptions

[3] Luís Moniz Pereira's view, complementing Kowalski's is that in the card setting rules are read as if and only if because the experimenter have devised the test is supposed to know and tell you everything: The Completion Semantics is applied. In the social setting the rules are envisaged as constraints, i.e. $\mathcal{IC}$s to be satisfied by the models but not generating the models. (personal communication, February 10, 2016)

Card	$\mathcal{P}$	lm wc $\mathcal{P}$		Griggs and Cox
beer	$\{ab_2 \leftarrow \bot, b \leftarrow \top\}$	$\langle\{b\}, \{ab_2\}\rangle$	$\not\models_{wcs}$ (5.4)	95%
coke	$\{ab_2 \leftarrow \bot, b \leftarrow \bot\}$	$\langle\emptyset, \{b, ab_2\}\rangle$	$\models_{wcs}$ (5.4)	0.025%
16 years	$\{ab_2 \leftarrow \bot, o \leftarrow \bot\}$	$\langle\emptyset, \{o, ab_2\}\rangle$	$\not\models_{wcs}$ (5.4)	80%
22 years	$\{ab_2 \leftarrow \bot, o \leftarrow \top\}$	$\langle\{o\}, \{ab_2\}\rangle$	$\models_{wcs}$ (5.4)	0.025%

Table 5.8.: The computational logic approach for the social case of the selection task.

or abnormalities and, hence, ab_2 is set to $\bot$.

Let us assume that conditional (5.4) is viewed as a social constraint which must follow logically from the given facts. Now consider the four different cases: One should observe that for the card *16 years old* the least model of the weak completion of $\mathcal{P}$, i.e. $\langle\emptyset, \{o, ab_2\}\rangle$, assigns U to b and, consequently, to both, $b \wedge \neg ab_2$ and (5.4), as well. Overall, for the cards *drinking beer* and *16 years old* the social constraint (5.4) is not entailed by the least model of the weak completion of the program. Hence, we need to turn over these cards and, hopefully, find that the beer drinker is older than 19 and that the 16 years old is not drinking beer. The results of the social case are shown in Table 5.8, where the last column shows the experimental results of Griggs and Cox [1982]. The results of our approach correspond to the majority's responses and, therefore, appears to be an adequate formalization.

5.2.2. Abstract Case

This case is artificial, and consequently, there is no common sense knowledge about the conditional. Following the perspective proposed by Kowalski [2011], let us assume that conditional (5.3) is viewed as a belief. As there are no known abnormalities, ab_1 is set to $\bot$. Furthermore, let D, F, 3, and 7 be propositional variables denoting that the corresponding symbol or number is on one side. Altogether, we obtain the program

$$\mathcal{P} = \{3 \leftarrow D \wedge \neg ab_1,\ ab_1 \leftarrow \bot\},$$

where its weak completion is

$$\mathsf{wc}\,\mathcal{P} = \{3 \leftrightarrow D \wedge \neg ab_1,\ ab_1 \leftrightarrow \bot\}$$

and admits the least model

$$\langle\emptyset, \{ab_1\}\rangle.$$

under the Weak Completion Semantics. Unfortunately, this model does not explain any symbol on any card. We need to extend the program based on which card we observe.

$\mathcal{O}$	$\mathcal{E}$	$\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P} \cup \mathcal{E})$		Wason
D	$\{D \leftarrow \top\}$	$\langle\{D,\ 3\}, \{ab_1\}\rangle$	$\rightsquigarrow$ turn over	89%
F	$\{F \leftarrow \top\}$	$\langle\{F\}, \{ab_1\}\rangle$	$\rightsquigarrow$ no turn over	16%
3	$\{D \leftarrow \top\}$	$\langle\{D,\ 3\}, \{ab_1\}\rangle$	$\rightsquigarrow$ turn over	62%
7	$\{\ 7 \leftarrow \top\}$	$\langle\{\ 7\}, \{ab_1\}\rangle$	$\rightsquigarrow$ no turn over	25%

Table 5.9.: The computational logic approach for the abstract case of the selection task.

In order to explain an observed card, we apply abduction. In the case of the abstract case, the set of abducibles is

$$\{D \leftarrow \top,\ D \leftarrow \bot,\ F \leftarrow \top,\ F \leftarrow \bot,\ 7 \leftarrow \top,\ 7 \leftarrow \bot\}.$$

Now consider the four different cases: In the cases where F or 7 are observed, the least model of the weak completion of $\mathcal{P} \cup \mathcal{E}$ does not contain any information that needs to be verified and simply confirms the observation; no further action is needed. In some sense, the belief about the premises and conclusions of the conditional is irrelevant for these two cases. The truth values of 3 and D are unknown and under Łukasiewicz semantics the conditional $3 \leftarrow D \wedge ab_1$ is mapped to true. In these two cases as well, Stenning and van Lambalgen's suggestion to interpret conditionals under SvL-semantics, would lead to different results: If premise and conclusion are unknown, the conditional, $3 \leftarrow D \wedge ab_1$, is not true but unknown as well. Their suggested least model of the completion would map 3 and D to false. This seems rather counterintuitive, as we do not know anything about the value on the other side of the card. By applying their approach we would have to conclude that both, 3 and D, are false.

In the case where D is observed, the least model maps also 3 to $\top$. That means, in order to be sure that this corresponds to the real situation, we need to check if 3 is true. Therefore, the card showing D is turned over. Likewise, in the case where 3 is observed, D is also mapped to $\top$ in the least model of the weak completion, which can only be confirmed if the card is turned over. As in each case there is only one explanation, there is no need to distinguish between skeptical and credulous reasoning. The results of the abstract case are shown in Table 5.9, where the last column shows the experimental results of Wason [1968]. The results of our approach corresponds to the majority's responses and, therefore, appears to be adequate.

5.3. Conclusion

Before we summarize this chapter, we would like to address a few open questions regarding the Weak Completion Semantics with respect to model human reasoning.

5.3.1. Open Questions

While adequately solving Byrne's suppression task and Wason's selection task, the approach gives rise to a number of open questions concerning the use of Łukasiewicz semantics, unique fixed points, completion versus weak completion, explanations, negation, and skeptical versus credulous approaches in human reasoning.

Łukasiewicz Semantics

This logic was selected because the technical bugs in [Stenning and van Lambalgen, 2008] can be solved by switching from Fitting to Łukasiewicz semantics. In particular, the model intersection property holds under Łukasiewicz semantics. Hence, for each program $\mathcal{P}$ a least model exists which can be computed as least fixed point of the $\Phi_\mathcal{P}$ operator. Moreover, as we have shown, the suppression task and the selection task can be adequately modeled under Weak Completion Semantics, whereas this does not hold for Fitting semantics. Nevertheless, the main question of whether the Weak Completion Semantics is adequate for human reasoning is still open. For example, under Łukasiewicz semantics the Deduction Theorem does not hold, neither does it hold with respect to logic programs in general. Hence, it would be interesting to see how humans deal with the deduction theorem.

Unique Fixed Point

For each program $\mathcal{P}$ discussed here, the $\Phi_\mathcal{P}$ operator is a contraction. Thus, there is a unique fixed point, which can be computed by iterating $\Phi_\mathcal{P}$ on some initial interpretation. Consequently, if in these tasks subjects are influenced towards some initial non-empty interpretation, their performance should not differ provided that they have enough time to compute the least fixed point; it should differ, however, if they are interrupted before the least fixed point is computed and asked to reason with respect to the interpretation computed so far. Another aspect is about level mapping. It might have the additional function to represent some ordering about the subject's knowledge. For instance, consider again the suppression task: It is easy to see that $\ell(t)$, $\ell(o)$, and $\ell(e)$ have to be smaller than $\ell(ab_1), \ell(ab_2)$, and $\ell(ab_3)$ to show that the program to be acyclic. As l does not occur in the body for any clause, $\ell(l)$ is mapped to the highest level. For human reasoning that means l does not imply any further knowledge.

Completion versus Weak Completion

The program $\mathcal{P}_{\neg e+Alt}$ served as an example to illustrate that completion is inadequate for the suppression task whereas weak completion is adequate. Likewise, Hölldobler, Philipp, and Wernhard [2011] have shown in a detailed study that the programs mentioned in

Table 5.4 together with their minimal explanations must be weakly completed in order to adequately model the suppression task, whereas when their completion is considered, the result does not correspond to the results of the suppression task. Are there other human reasoning episodes which support the claim that weak completion is adequate?

Skeptical versus Credulous Reasoning

The case of program $\mathcal{P} = \mathcal{P}_{Alt}$ and observation $\mathcal{O} = \{l\}$ in Table 5.4 shows that we must reason skeptically in order to adequately model this case. Whereas this is a striking case for skeptical reasoning, the case $\mathcal{P} = \mathcal{P}_{Add}$ and $\mathcal{O} = \neg l$ is less convincing. Skeptically we do not conclude $\neg e$, whereas credulously we conclude $\neg e$. Compared to the corresponding case ($\mathcal{P}_{\neg l + Add}$) shown in Table 5.4, 44% of the subjects conclude $\neg E$. Unfortunately, Byrne [1989] (and related publications that we are aware of) gives no account of the distribution of the answers given by those subjects who did not conclude $\overline{E}$. Hence, at the moment we can argue in favor of skeptical reasoning (*the majority of the subjects did not conclude* $\overline{E}$), but – given the complete distribution – it may be the case that one can argue in favor of credulous reasoning (*there are more subjects concluding* $\overline{E}$ *than subjects concluding* E *and subjects answering "I don't know"*). In this context, it might be useful to explicitly differentiate between *inferential* knowledge and facts. For credulous reasoning the amount of *inferential* knowledge does not influence its conclusion. On the other hand, for skeptical reasoning, as more *inferential* knowledge is given, the more supporting facts are necessary to draw some conclusion.

Explanations

The approach presented in this paper is based on minimal explanations. Although, there are findings corroborating the human preference for minimal explanations (over non-minimal ones) [Ormerod, Manktelow, and Jones, 1993] – this holds only partially [Johnson-Laird, Girotto, and Legrenzi, 2004]. Computational models of abduction typically generate explanations iteratively such that minimal explanations are generated first. How are the minimal explanations computed by humans? What happens if there are more than one minimal explanation? Does attention influence the selection of explanations as we have suggested in Chapter 4?

Negation

As we have already discussed in the previous chapter, under the Weak Completion Semantics, positive information is preferred over negative information. Consider, for example, the program $\mathcal{P} = \{q \leftarrow \top, q \leftarrow \bot\}$. The least model of $\mathsf{wc}\,\mathcal{P}$ is $\langle\{q\}, \emptyset\rangle$ and, hence, an agent reasoning with respect to this model will conclude q. Is this consistent with human reasoning? We can extend the notion of integrity constraints by

allowing them of the form $\bot \leftarrow q$. Any model for a program containing such an integrity constraint must map q to $\bot$. Is this adequate for human reasoning? If so, under which conditions shall such integrity constraints be added within the reasoning step towards an appropriate logical form?

Connectionist Realization

As shown in [Hölldobler and Kencana Ramli, 2009a], the computation of the least fixed point of the semantic operator $\Phi_{\mathcal{P}}$ associated with a program $\mathcal{P}$ can be realized within the core-method [Bader, Hitzler, Hölldobler, and Witzel, 2007]. In this connectionist realization, $\Phi_{\mathcal{P}}$ is computed by a feed-forward network, whose output units are recurrently connected to the input units. Whereas this network is trainable by backpropagation and, thus, $\Phi_{\mathcal{P}}$ can be learned by experience, there is no evidence whatsoever that backpropagation is biological plausible. Dietz Saldanha, Hölldobler, Kencana Ramli, and Palacios Medinacelli [2017a] present a connectionist realization of skeptical reasoning. However, in both settings, explanations are generated in a fixed, hard-wired sequence, which does not seem to be plausible either.

5.3.2. Summary

Originally Stenning and van Lambalgen suggested to model Byrne's suppression task under the Completion Semantics based on the three-valued logic used by Fitting [1985]. However, Hölldobler and Kencana Ramli [2009a,b] have shown that the three-valued logic proposed by Stenning and van Lambalgen is inadequate for the suppression task, and that the suppression task can be adequately modeled if the weak completion and the three-valued logic by Łukasiewicz [1920] is used instead. The first section of this chapter presents the adequate formalization of Byrne's suppression task based on Hölldobler's and Kencana Ramli's approach. We show that other approaches, such as completed logic programs, the Fitting Semantics, and credulous reasoning, and show that they do not lead to adequate results. Accordingly, we show which of Stenning and van Lambalgen's technical claims are wrong. Furthermore, we have shown here how these programs can be modified such that they achieve the same results under the Well-founded Semantics.

While Stenning and van Lambalgen proposed a formalization for Byrne's suppression task, they only analyzed Wason's selection task but did not attempt to formalize this task with their suggested approach. This is what we have done in the second part of this Chapter. We model Wason's selection task with the same approach we did for Byrne's suppression task, following Kowalski's proposal to distinguish both cases as follows: In order to solve the social case correctly, the conditional must be seen as a social constraint, whereas the abstract case is correctly represented when the conditional is seen as a belief. The second case can be modeled by extending the formalization to reason (either credulously or skeptically) within an abductive framework. Even though

Kowalski showed how to formalize the abstract and the social case of Wason's selection task, he did not propose a solution to Byrne's suppression task. Unlike Kowalski and Stenning and van Lambalgen who each proposed a formalization for only one task, we show here that the Weak Completion Semantics seems to adequately model both tasks.

However, there are still aspects we did not consider yet and which need to be further examined. Our approach does not deal with the so-called first step of modeling human reasoning: reasoning with respect to an adequate representation. We just assume that in the social case people take the conditional as a social constraint whereas they take it as a belief in the abstract case. These differences are modeled outside of the formal framework. Dawson and Regan [2002] show by some psychological experiments that the so-called *confirmation bias* plays an important role in the Wason selection task: if people disagree with the statement of the conditional, they are more likely to find the solution because they are motivated to search for a counterexample which refutes the conditional. On the other hand, people who agree with the statement of the conditional take it as a confirmation of their believes and therefore will not extensively search to falsify the conditional. An interesting observation discussed in [Stenning and van Lambalgen, 2008] is that similar to the *verification bias*, people might transfer the *truth of the card* to the *truth of the rule*. In the social case, this confusion cannot occur, because it is commonsense that the rule is true, independently from whether people behave accordingly. This leads to another phenomenon, namely that participants see a dependency between the card choices and might prefer to solve the problem by *reactive planning*. They would only like to decide what to do after they saw the outcome of the first card. For instance, if one turns over card D first and there is no 3 on the other side, no further cards needs to be examined, because the rule has been falsified. However, if there is a 3 on the other side, the other options need to be considered again. This kind of behavior could be described in a framework with belief change: Each card which is turned over is a piece of new information which needs to be integrated into the current knowledge base and updates new inferences accordingly.

6. Spatial Relations

In this Chapter, we present a new approach with respect to spatial reasoning problems using logic programs, where we formalize the Preferred Model Theory in a computational logic setting based on the Weak Completion Semantics. In order to do so, we first show a spatial relation task in human reasoning in Section 6.1. In Section 6.2, we briefly go through some spatial reasoning approaches, in particular, we will discuss the Preferred Model Theory. Thereafter, we show how the Preferred Model Theory can be implemented under the Weak Completion Semantics.[1]

6.1. Introduction

Given the information, that a Ferrari is to the left of a Porsche and to the right of the Porsche there is a Beetle, we can without difficulty conclude that the Ferrari is to the left of the Beetle. But how exactly do we come to this conclusion? What happens if we have a set of premises with which more than one arrangement is possible? Consider the following example taken from Ragni and Knauff [2013], which consists of four premises above the line and a conclusion below the line.

1. The Ferrari is left of the Porsche.
2. The Beetle is right of the Porsche.
3. The Porsche is left of the Hummer.
4. The Hummer is left of the dodge.

C. The Porsche is (necessarily)[2] left of the dodge.

Would a human immediately notice that the first two premises are not relevant and that the conclusion follows from the transitivity given the *left* relation of the third and the fourth premise?

The question which we shall be discussing is how to automatically construct what humans may have in mind while reading the premises. Accordingly, the conclusion should be evaluated in a similar way as humans do. For instance, Mental Model Theory assumes that humans construct one model, verify the conclusion and possibly construct the next

[1] The results of this chapter are originally based on the result by Raphael Höps, a student that we supervised for his bachelor thesis project, written in German [Höps, 2014]. The bachelor thesis has been heavily revised and published in English in [Dietz, Hölldobler, and Höps, 2015a].

[2] This means that the conclusion follows in all possible solutions.

model [Johnson-Laird, 1983]. On the other hand, the Preferred Model Theory claims that humans have exactly one so-called preferred model in mind [Ragni and Knauff, 2013]. This requires less effort than constructing all possible models and in most situations of our everyday life this one model is enough to reason with. According to Ragni and Knauff, people will only think of alternatives if they are asked to search for other models. Nevertheless, even then, they do not randomly construct them from scratch. Instead of that, they first start generating models which are most similar to the preferred one by changing them as little as possible. This theory seems to be very promising and is empirically supported. Furthermore, it can give us an explanation about how people deal with ambiguity when modeling spatial reasoning problems and that, while evaluating a conclusion, they might come to conclusions which are wrong according to classical logic.

6.2. Theories about Spatial Relations

We will first address the spatial reasoning problem following the introduction in [Ragni and Knauff, 2013]. We assume binary spatial relations between two objects and restrict ourselves in this paper to the *left* and *right* relations. We use the notation $left(X, Y)$ and $right(X, Y)$ to express that X is left of Y and X is right of Y, respectively. Reconsidering the example from the introduction we obtain

Example 1.

1.	$left(ferrari, porsche)$
2.	$right(beetle, porsche)$
3.	$left(porsche, hummer)$
4.	$left(hummer, dodge)$
$C.$	$left(porsche, dodge)$

Formally, a *spatial reasoning problem* consists of a finite list of *premises*, a *conclusion*, and the question whether the premises entail the conclusion. We assume that each premise is a ground atom of the form $p(a, b)$ specifying some spatial relation p between the *objects* a and b. The conclusion is also a ground atom of the form $p(c, d)$, where the objects c and d must occur in the premises.

6.2.1. Inference Rule Approach

The *inference rule approach*, as presented by Byrne and Johnson-Laird [1989], is based on following assumptions:

1. Humans know a set of inference rules, which they can apply to the premises of the spatial reasoning problem in order to derive new knowledge.

2. In case they encounter the conclusion, it is proven. The conclusion is only refuted, when all possibilities of applying the rules are exploited without proving the conclusion.

3. The order of the premises is not important.

A system with nine inference rules for spatial reasoning problems is specified, which does not only consider the *left* and *right* relations, but also the *front* relation. As we will restrict ourselves to the *left* and the *right* relation, we will only consider a simplified version of the rule system.

$$\forall X, Y, Z(\mathit{left}(X, Z) \leftarrow \mathit{left}(X, Y) \wedge \mathit{left}(Y, Z)) \quad (1)$$
$$\forall X, Y(\mathit{left}(X, Y) \leftrightarrow \mathit{right}(Y, X)) \quad (2)$$

(1) represents the transitivity of the *left* relations and (2) the symmetry between the *left* and *right* relation. In this system, Example 1 can be solved by considering premises 3 and 4 together with rule (1) specifying the transitivity of *left*, which allows one to infer the conclusion. These rules seem appropriate and are necessary to identify the relations between the objects. However, the major drawback of this approach is that it does not consider the order in which premises are read.

6.2.2. Mental Model Theory

Mental Model Theory does not assume that humans apply inference rules but that they construct so-called *mental models* [Johnson-Laird, 1983]. In spatial reasoning, a mental model is understood as the representation of the spatial arrangements between objects that correspond to the premises. Consider Example 2, again taken from [Ragni and Knauff, 2013], which is similar to Example 1 except for the third premise.

Example 2.

1. $\mathit{left}(\mathit{ferrari}, \mathit{porsche})$
2. $\mathit{right}(\mathit{beetle}, \mathit{porsche})$
3. $\mathit{left}(\mathit{beetle}, \mathit{hummer})$
4. $\mathit{left}(\mathit{hummer}, \mathit{dodge})$

C. $\mathit{left}(\mathit{porsche}, \mathit{dodge})$

A mental model corresponding to the premises is constructed by writing the objects next to each other exactly how one would imagine the arrangement in one's mind. In this case, there exists only one possible mental model.

ferrari *porsche* *beetle* *hummer* *dodge*

A spatial reasoning problem which has exactly one mental model is called a *deterministic* problem. On the other hand, Example 1 is a *non-deterministic* problem, for which we

have three mental models.

ferrari	*porsche*	*beetle*	*hummer*	*dodge*
ferrari	*porsche*	*hummer*	*beetle*	*dodge*
ferrari	*porsche*	*hummer*	*dodge*	*beetle*

Mental Model Theory assumes that when solving a spatial reasoning problem, humans behave as follows:

1. One of the mental models is constructed.
2. If the conclusion does not hold in this model, then it is refuted.
3. If the conclusion holds in this model then, one mental model after another is constructed and verified. If the conclusion holds in all these models, then it is proven.

In the original version of Mental Model Theory as well as in the inference rule approach, it is assumed that the order of the premises is irrelevant. Yet, Ragni, Knauff, and Nebel [2005] already refer to the studies made in [Manktelow, 2000], which show, that the order actually influences the construction of mental models and, in particular, the construction of the first mental model in Step 1.

6.2.3. Preferred Model Theory

In [Ragni and Knauff, 2013], the Preferred Model Theory is presented, which is based on Mental Model Theory. The major difference is the assumption that humans do not consider all but only certain mental models. These certain ones in turn depend on the order in which the premises are presented. In this section we will discuss the Preferred Model Theory together with the computational system PRISM, developed by Ragni and Knauff [2013] as well.

One should note that $\mathit{left}(a, b)$ or $\mathit{right}(a, b)$ denote that object a is left or right of object b, respectively, but this does not mean that a is a neighbor of b. There might be other objects between a and b. In case a is a neighbor of b, i.e. there is no other object between them, then we will refer to this relation as *left neighbor of* or *right neighbor of* and denote this as $\mathit{ln}(a, b)$ and $\mathit{ln}(b, a)$, respectively.

The Preferred Model Theory assumes that the phases in which humans solve spatial reasoning problems can be divided into a model construction, a model inspection and a model variation phase. Ragni and Knauff also present PRISM, an implementation of the Preferred Model Theory. It only allows the following four types of premises for a spatial reasoning problem:

Type 1 Exactly the first premise.

Type 2 Premises containing exactly one new object and one which occurs already in the model so far.

Type 3 Premises which contain two new objects, but which are not from type 1.

Type 4 Premises which relate two objects to each other, occurring in two different *submodels*, where submodels are arrangements of objects which are already formed but which are not yet in a relationship to each other.

Based on a list of premises, PRISM constructs the preferred mental model by stepwise adding new objects to an initially empty arrangement. For this purpose, one premise after another is read and, depending on its type, is processed as follows:

1. If the premise is of type 1, then the two objects will be placed directly next to each other.

2. If the premise is of type 2, then the new object will be inserted directly next to the already existing one, provided that the space next to the existing one is free. If this space is already occupied, then the new object is placed in the next available space. This is called *first free fit* or *3f-strategy*.

3. If the premise is of type 3, then a new arrangement, that is a new submodel, is constructed in which both objects are arranged directly next to each other.

4. If the premise is of type 4, the first arrangement will be placed directly next to the second arrangement; the objects within these arrangements do not change their places.

We illustrate PRISM by reconsidering Example 1. Reading the premises one by one, the preferred mental model is constructed as follows: After reading the first premise, $left(ferrari, porsche)$, which is of type 1, the Ferrari and the Porsche are placed next to each other.

$$ferrari \qquad porsche$$

After reading the second premise, $right(beetle, porsche)$, which is of type 2, the Beetle is added next to the Porsche as its right neighbor.

$$ferrari \qquad porsche \qquad beetle$$

After reading the third premise, $left(porsche, hummer)$, which is again of type 2, we notice that the Hummer cannot be placed directly right of the Porsche because this space is already occupied. Therefore, the Hummer will be placed on the first free space right of the Porsche.

$$ferrari \qquad porsche \qquad beetle \qquad hummer$$

Finally, after reading the forth premise, $left(hummer, dodge)$, the dodge is placed as right neighbor of the Hummer and we obtain the following preferred mental model:

ferrari *porsche* *beetle* *hummer* *dodge*

and then, in the second phase, we check whether the conclusion holds. As *porsche* is left of *dodge* in the above model, humans normally respond '*yes*'. Only if they are explicitly pointed to consider other models, the third phase starts and they try to change the model with the least possible number of operations. For Example 1, the models

ferrari	*porsche*	*hummer*	*beetle*	*dodge*
ferrari	*porsche*	*hummer*	*dodge*	*beetle*

are generated subsequently. The conclusion holds for all three models and indeed, the classical logically correct answer is '*yes*'. Interestingly, most humans appear to infer their answer immediately after generating the preferred mental model.

Let us consider yet another spatial reasoning problem, Example 3.

Example 3.

1.	$left(ferrari, porsche)$
2.	$right(beetle, hummer)$
3.	$left(ferrari, beetle)$
$C.$	$left(porsche, beetle)$

After reading the first premise, the Ferrari and the Porsche are placed next to each other.

ferrari *porsche*

The second premise is of type 3, that means both objects are placed next to each other in a new empty arrangement.

hummer *beetle*

Only after reading the third premise, which is of type 4, both submodels are put in relation to each other and we obtain the following preferred mental model:

ferrari *porsche* *hummer* *beetle*

and accordingly, the conclusion is true in the preferred mental model. Only in exceptional cases, humans try to do some model variation, and figure out that there is also another model which agrees with the premises.

ferrari *hummer* *beetle* *porsche*

In this model, the conclusion does not hold anymore. Nevertheless, the Preferred Model Theory assumes that the majority of the people believes that the conclusion holds. As

we aim at modeling human reasoning, the first model corresponds to the conclusion, which is what we intend to model.

6.3. Representation as Logic Programs

Let us recall the notation from Chapter 2 by applying them to Example 1. Consider $\mathcal{P}_{ex}$, the preliminary program representing Example 1, which consists of the following five clauses:

$$\begin{aligned} \mathit{left}(\mathit{ferrari}, \mathit{porsche}) &\leftarrow \top. \\ \mathit{left}(\mathit{porsche}, \mathit{beetle}) &\leftarrow \top. \\ \mathit{left}(\mathit{porsche}, \mathit{hummer}) &\leftarrow \top. \\ \mathit{left}(\mathit{hummer}, \mathit{dodge}) &\leftarrow \top. \\ \mathit{right}(X, Y) &\leftarrow \mathit{left}(Y, X). \end{aligned}$$

The first four clauses represent the four premises denoted as a set of facts about the *left* relation between objects. Note that instead of the second clause, we could have written

$$\mathit{right}(\mathit{beetle}, \mathit{porsche}) \leftarrow \top.$$

instead. In order to simplify the representation of the programs in the following, we will only allow *left* relations as facts or assumptions in the programs. Anyway, because of the last clause, we can still conclude the corresponding *right* relation, where the last clause is not a fact but simply a clause representing the symmetry between the *left* and *right* relations. Imagine that the Ferrari would not actually be left of the Porsche. We represent this as the assumption

$$\mathit{left}(\mathit{ferrari}, \mathit{porsche}) \leftarrow \bot.$$

The constants in $\mathcal{P}_{ex}$ are

$$\mathsf{constants}(\mathcal{P}_{ex}) = \{\mathit{ferrari}, \mathit{porsche}, \mathit{beetle}, \mathit{hummer}, \mathit{dodge}\}$$

and the weak completion of $\mathcal{P}_{ex}$ is

$$\begin{aligned} \mathsf{wc}\,\mathcal{P}_{ex} = \{ & \quad \mathit{left}(\mathit{ferrari}, \mathit{porsche}) \leftrightarrow \top, \\ & \quad \mathit{left}(\mathit{porsche}, \mathit{beetle}) \leftrightarrow \top, \\ & \quad \mathit{left}(\mathit{porsche}, \mathit{hummer}) \leftrightarrow \top, \\ & \quad \mathit{left}(\mathit{hummer}, \mathit{dodge}) \leftrightarrow \top \ \} \\ & \cup \ \{\mathit{right}(o_1, o_2) \leftrightarrow \mathit{left}(o_2, o_1) \mid o_1, o_2 \in \mathsf{constants}(\mathcal{P}_{ex})\}. \end{aligned}$$

The least model of the weak completion of $\mathcal{P}_{ex}$ is $\langle I^\top, \emptyset \rangle$, where

$$\begin{aligned} I^\top = \{ & \mathit{left}(\mathit{ferrari}, \mathit{porsche}), & & \mathit{left}(\mathit{porsche}, \mathit{beetle}), \\ & \mathit{left}(\mathit{porsche}, \mathit{hummer}), & & \mathit{left}(\mathit{hummer}, \mathit{dodge}), \\ & \mathit{right}(\mathit{porsche}, \mathit{ferrari}), & & \mathit{right}(\mathit{beetle}, \mathit{porsche}), \\ & \mathit{right}(\mathit{hummer}, \mathit{porsche}), & & \mathit{right}(\mathit{dodge}, \mathit{hummer})\}. \end{aligned}$$

6.4. Reasoning with Respect to Preferred Mental Models

Following the Preferred Model Theory, we show how the preferred mental model of a spatial reasoning problem can be computed by logic programs under the Weak Completion Semantics. This approach covers the model construction and the model inspection phase.

The running example in Section 6.3 shows us that relations between objects can be easily represented in logic programs. However, there is no straightforward way in which we can express the order in which the premises are read. But precisely this information is crucial if we want to formalize the Preferred Model Theory. For this purpose, in the approach we propose now, we explicitly express phases where each premise is read at one particular phase. This allows us to define the order in which the premises are processed. In contrast to PRISM, we do not distinguish between the model construction and model inspection phase, but process them at the same time.

Let $\mathcal{S}$ be a spatial reasoning problem. The following program $\mathcal{P}_\mathcal{S}$ represents the premises of $\mathcal{S}$ and the necessary background knowledge in order to construct the preferred mental model. Within $\mathcal{P}_\mathcal{S}$ we will use the following notation with informal meaning as follows:

$l(X, Y, i)$	in phase i, X is placed to the left of Y,
$ln(X, Y, i)$	in phase i, X is the left neighbor of Y,
$ol(X, i)$	in phase i, the space directly left of X is occupied,
$or(X, i)$	in phase i, the space directly right of X is occupied,

where i starts with 1. In the following, n indicates the number of premises processed so far. Given a spatial reasoning problem $\mathcal{S}$, the corresponding program $\mathcal{P}_\mathcal{S}$ is constructed as follows:

1. We start by reading the premises. For each premise do: If the ith premise is of the form $\mathit{left}(o_1, o_2)$ or $\mathit{right}(o_1, o_2)$, then add

   $$l(o_1, o_2, i) \leftarrow \top. \quad \text{or} \quad l(o_2, o_1, i) \leftarrow \top.$$

   respectively, to the (initially empty) program $\mathcal{P}_\mathcal{S}$, where o_1 and o_2 are assumed to be different objects. After that, we also know $\mathsf{constants}(\mathcal{P}_\mathcal{S})$, the set of constants in $\mathcal{P}_\mathcal{S}$.

2. We make a closed-world assumption for the l relation in phase 1 as initially nothing is known about the spatial relation of objects:

$$\{l(o_1, o_2, 1) \leftarrow \bot \mid o_1, o_2 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}) \text{ and } o_1 \neq o_2\}.$$

One should observe that programs are weakly completed, e.g. if the first premise of a spatial reasoning problem is of the form $\mathit{left}(\mathit{porsche}, \mathit{hummer})$ then the ground facts and assumptions

$$l(\mathit{porsche}, \mathit{hummer}, 1) \leftarrow \top \quad \text{and} \quad l(\mathit{porsche}, \mathit{hummer}, 1) \leftarrow \bot$$

are generated in the first two steps, respectively. Their weak completion is

$$l(\mathit{porsche}, \mathit{hummer}, 1) \leftrightarrow \top \vee \bot \equiv l(\mathit{porsche}, \mathit{hummer}, 1) \leftrightarrow \top.$$

Recall that by $\equiv$ we mean semantical equivalence. Under the Weak Completion Semantics facts override assumptions. In other words, there is no obligation to place o_1 to the left of o_2 in phase i unless explicitly stated in the ith premise.

3. As at the beginning no objects have been placed, the space to the left and to the right of each object is initially empty:

$$\{ol(o, 1) \leftarrow \bot \mid o \in \mathsf{constants}(\mathcal{P}_\mathcal{S})\} \cup \{or(o, 1) \leftarrow \bot \mid o \in \mathsf{constants}(\mathcal{P}_\mathcal{S})\}.$$

This corresponds to the closed-world assumption with respect to the ol relation, which needs to be explicitly made under the Weak Completion Semantics. In the running example, we find that the space to the left and to the right of both cars, the Porsche and the Hummer, are empty in phase 1. Accordingly, $\mathcal{P}_\mathcal{S}$ additionally consists of the following four clauses:

$$\begin{aligned} ol(\mathit{porsche}, 1) &\leftarrow \bot. \\ or(\mathit{porsche}, 1) &\leftarrow \bot. \\ ol(\mathit{hummer}, 1) &\leftarrow \bot. \\ or(\mathit{hummer}, 1) &\leftarrow \bot. \end{aligned}$$

4. We start to place objects. If in phase i object o_1 should be placed to the left of object o_2 and the space to the left of o_1 as well as the space to the right of o_1 are empty, then o_1 is placed as the left neighbor of o_2:

$$\begin{aligned} \{ln(o_1, o_2, i) \leftarrow\ & l(o_1, o_2, i) \wedge \neg ol(o_2, i) \wedge \neg or(o_1, i) \mid \\ & o_1, o_2 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}), \\ & o_1 \neq o_2 \text{ and } i \in \{1, ..., n\}\}. \end{aligned}$$

For the running example, we obtain, among others, in phase 1

$$\begin{aligned} ln(\mathit{porsche}, \mathit{hummer}, 1) \leftarrow\ & l(\mathit{porsche}, \mathit{hummer}, 1) \wedge \\ & \neg ol(\mathit{hummer}, 1) \wedge \neg or(\mathit{porsche}, 1). \end{aligned}$$

Given 1., 2. and 3., the body of this clause will be *true* and, consequently, the Porsche will be placed as the left neighbor of the Hummer in phase 1.

5. Once an object o_1 has become the left neighbor of another object o_2 in phase i, this relation holds until the preferred mental model is constructed:

$$\begin{aligned}\{ln(o_1, o_2, i+1) \quad \leftarrow \quad & ln(o_1, o_2, i) \mid \\ & o_1, o_2 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}), \\ & o_1 \neq o_2 \text{ and } i \in \{1, ..., n-1\}\}.\end{aligned}$$

6. If o_1 has become the left neighbor of o_2 in phase i, then the space to the left of o_2 as well as the space to the right of o_1 are occupied in phase $i+1$:

$$\begin{aligned}\{ol(o_2, i+1) \quad \leftarrow \quad & ln(o_1, o_2, i) \mid \\ & o_1, o_2 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}), \\ & o_1 \neq o_2 \text{ and } i \in \{1, ..., n-1\}\} \cup \\ \{or(o_1, i+1) \quad \leftarrow \quad & ln(o_1, o_2, i) \mid \\ & o_1, o_2 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}), \\ & o_1 \neq o_2 \text{ and } i \in \{1, ..., n-1\}\}.\end{aligned}$$

In combination with 5., the space to the left of o_2 and the space to the right of o_1 are occupied in all future phases. For example, after the Porsche has been placed as left neighbor of the Hummer in phase 1, the following two clauses determine that there is no space anymore immediately to the left of the hummer and immediately to the right of the Porsche at phase 2:

$$\begin{aligned}ol(hummer, 2) \quad &\leftarrow \quad ln(porsche, hummer, 1). \\ or(porsche, 2) \quad &\leftarrow \quad ln(porsche, hummer, 1).\end{aligned}$$

7. If o_1 should be placed to the left of o_2, but there is already a left neighbor o_3 of o_2, then o_1 is placed to the left of o_3:

$$\begin{aligned}\{l(o_1, o_3, i+1) \quad \leftarrow \quad & l(o_1, o_2, i+1) \wedge ln(o_3, o_2, i) \mid \\ & o_1, o_2, o_3 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}), \\ & \mathit{diff}(o_1, o_2, o_3) \text{ and } i \in \{1, ..., n-1\}\},\end{aligned}$$

where $\mathit{diff}(o_1, o_2, o_3)$ means that o_1, o_2 and o_3 are different objects. One should observe that this can only happen from phase 2 onwards, as in the first phase none of the objects has a left neighbor. This is the reason for writing $i+1$ in the atom $l(o_1, o_2, i+1)$ occurring in the bodies of the clauses.

8. Likewise, if o_1 should be placed to the left of o_2, but o_1 is already the left neighbor

of some other object o_3, then o_3 should be placed to the left of o_2:

$$\{l(o_3, o_2, i+1) \leftarrow l(o_1, o_2, i+1) \wedge ln(o_1, o_3, i) \mid o_1, o_2, o_3 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}), \mathit{diff}(o_1, o_2, o_3) \text{ and } i \in \{1, ..., n-1\}\}.$$

9. Finally, in order to determine whether the conclusion is true, we add the following clauses to $\mathcal{P}_\mathcal{S}$. If o_1 is the left neighbor of o_2 after processing all premises, then o_1 is to the left of o_2 in the preferred mental model:

$$\{\mathit{left}(o_1, o_2) \leftarrow ln(o_1, o_2, n) \mid o_1, o_2 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}) \text{ and } o_1 \neq o_2\}.$$

10. The *left* relation is transitive:

$$\{\mathit{left}(o_1, o_3) \leftarrow \mathit{left}(o_1, o_2) \wedge \mathit{left}(o_2, o_3) \mid o_1, o_2, o_3 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}) \text{ and } \mathit{diff}(o_1, o_2, o_3)\}.$$

11. The *right* relation is the inverse of the *left* relation:

$$\{\mathit{right}(o_1, o_2) \leftarrow \mathit{left}(o_2, o_1) \mid o_1, o_2 \in \mathsf{constants}(\mathcal{P}_\mathcal{S}) \text{ and } \mathit{diff}(o_1, o_2)\}.$$

In each phase, one premise is read and understood as a request to place the mentioned objects in the required order. Objects are placed in the first available space like in the PRISM approach, where again in each phase exactly one request to place objects is processed and the objects in the request are placed. Once the least fixed point of $\Phi_{\mathcal{P}_\mathcal{S}}$ has been reached, we can identify the preferred mental model: Given a problem $\mathcal{S}$, o_1 is left of o_2 iff $\mathit{left}(o_1, o_2)$ holds in the least fixed point. This will be illustrated by two examples in the next subsection.

6.5. Examples

We consider the spatial reasoning problem, Example 4.

Example 4.

1.	$\mathit{left}(\mathit{porsche}, \mathit{hummer})$
2.	$\mathit{left}(\mathit{dodge}, \mathit{hummer})$
C.	$\mathit{left}(\mathit{dodge}, \mathit{porsche})$

Let $\mathcal{P}_4$ be the logic program corresponding to Example 4 and $\Phi_{\mathcal{P}_4}$ the corresponding semantic operator.[3] We abbreviate the constants representing cars by their first letter, i.e.

[3] $g\mathcal{P}_4$ can be found in Appendix C.

Iteration	$I^\top$	$I^\bot$	#
$\Phi_{\mathcal{P}_4}\uparrow 0$	$\emptyset$	$\emptyset$	
$\Phi_{\mathcal{P}_4}\uparrow 1$	$l(p,h,1)$		1.
	$l(d,h,2)$		1.
		$l(d,h,1), l(d,p,1), l(h,d,1), l(h,p,1), l(p,d,1)$	2.
		$ol(d,1), ol(h,1), ol(p,1), or(d,1), or(h,1), or(p,1)$	3.
$\Phi_{\mathcal{P}_4}\uparrow 2$	$ln(p,h,1)$	$ln(d,h,1), ln(d,p,1), ln(h,d,1), ln(h,p,1), ln(p,d,1)$	4.
$\Phi_{\mathcal{P}_4}\uparrow 3$	$ln(p,h,2)$		5.
	$ol(h,2), or(p,2)$	$ol(d,2), ol(p,2), or(d,2), or(h,2)$	6.
	$l(d,p,2)$	$l(h,p,2), l(p,d,2), l(p,h,2)$	7.,8.
$\Phi_{\mathcal{P}_4}\uparrow 4$	$ln(d,p,2)$	$ln(d,h,2), ln(h,p,2), ln(p,d,2)$	4.
		$l(h,d,2)$	7.,8.
	$left(p,h)$		9.
$\Phi_{\mathcal{P}_4}\uparrow 5$		$ln(h,d,2)$	4.
	$\boldsymbol{left(d,p)}$		9.
	$right(h,p)$		11.
$\Phi_{\mathcal{P}_4}\uparrow 6$	$left(d,h)$		10.
	$right(p,d)$		11.
$\Phi_{\mathcal{P}_4}\uparrow 7$	$right(h,d)$		11.

Table 6.1.: The least model of the weak completion of $\mathcal{P}_4$ is computed by iterating $\Phi_{\mathcal{P}_4}$ until the least fixed point is reached. In each iteration only atoms are listed which appear in $I^\top$ and $I^\bot$ for the first time. # lists the clauses responsible for adding an atom to $I^\top$ or $I^\bot$. The atom in bold confirms the conclusion: The dodge is to the left of the Porsche.

d, h and p are abbreviations for *dodge*, *hummer* and *porsche*, respectively. In Table 6.1, we illustrate the computation of the least fixed point of $\Phi_{\mathcal{P}_4}$ step by step, where $\Phi_{\mathcal{P}_4}\uparrow n$ denotes I after the nth iteration of $\Phi_{\mathcal{P}_4}$. Focusing on atoms which are mapped to true, i.e. which are in $I^\top$, we find:

- In the first iteration of the $\Phi_{\mathcal{P}_4}$ operator ($\Phi_{\mathcal{P}_4}\uparrow 1$) the requests to place the Porsche to the left of the Hummer in phase 1 and the dodge to the left of the Hummer in phase 2 are recorded.
- In $\Phi_{\mathcal{P}_4}\uparrow 2$, the Porsche becomes the left neighbor of the Hummer in phase 1.
- In $\Phi_{\mathcal{P}_4}\uparrow 3$, we learn that the space to the left of the Hummer as well as the space to the right of the Porsche are occupied in phase 2. As the Porsche is the left neighbor of the Hummer in phase 1, this relationship is preserved in phase 2 and the dodge must be placed to the left of the Porsche in phase 2.
- In $\Phi_{\mathcal{P}_4}\uparrow 4$, the dodge becomes the left neighbor of the Porsche in phase 2 and we find that the Porsche and the Hummer are in the *left* relation.

- In $\Phi_{\mathcal{P}_4}\uparrow 5$, we find that the dodge and the Porsche are in the *left* relation, whereas the Hummer and the Porsche are in the *right* relation.
- In $\Phi_{\mathcal{P}_4}\uparrow 6$, we find by transitivity that the dodge and the Hummer are in the *left* relation, and the Porsche and the dodge are in the *right* relation.
- Finally, in $\Phi_{\mathcal{P}_4}\uparrow 7$, the Hummer and the dodge are in the *right* relation.

Indeed, as shown in bold in Table 6.1, the conclusion of Example 4 holds in the preferred mental model: The dodge is to the left of the Porsche.

We return to Example 3 from Section 6.2.3, which contains premises of type 3 and 4, i.e. premises that generate submodels. Let $\mathcal{P}_3$ be the logic program corresponding to Example 3 and $\Phi_{\mathcal{P}_3}$ be the corresponding semantic operator. We again abbreviate the constants representing *beetle*, *hummer*, *ferrari* and *porsche* by their first letter, i.e. b, h, f and p, respectively. In Table 6.2, we depict the computation of the least fixed point of $\Phi_{\mathcal{P}_3}$. For $I^\top$ we find:

- In the first iteration the three requests to place objects are recorded.
- In the second and the forth iteration, the Ferrari becomes the left neighbor of the Porsche and the Hummer becomes the left neighbor of the Beetle, respectively, thus generating two submodels which are not connected at this step.
- In the fifth and the sixth iteration, the request to place the Ferrari to the left of the Beetle ($l(f, b, 3)$) is processed. This generates $l(f, h, 3)$ and, thereafter $l(p, h, 3)$.
- The Porsche becomes the left neighbor of the Hummer in the seventh iteration leading to the preferred mental model.

Indeed, as shown in bold in Table 6.2, the conclusion of Example 3 holds in the preferred mental model: The Porsche is to the left of the Beetle.

6.6. Conclusion

We have shown that our computational logic approach based on the Weak Completion Semantics can compute preferred mental models for spatial reasoning problems. We have restricted our presentation to the *left* and *right* relation, but the formalization can be extended to include additional ones like the *front* or the *back* relations. Likewise, we should be able to handle the four cardinal directions. Different than other approaches such as described in [Goodwin and Johnson-Laird, 2005], the Preferred Model Theory explains how a model is constructed and seems to be able to predict conclusions humans make given a spatial reasoning problem. This allows us to understand how they influence the model construction, as we have shown by Example 3 and 4.

Iteration	$I^\top$	$I^\bot$	#
$\Phi_{\mathcal{P}_3}\uparrow 0$	$\emptyset$	$\emptyset$	
$\Phi_{\mathcal{P}_3}\uparrow 1$	$l(f,p,1), l(h,b,2), l(f,b,3)$		1.
		$l(b,f,1), l(b,h,1), l(b,p,1), l(f,b,1), l(p,b,1),$	2.
		$l(f,h,1), l(h,b,1)l(h,f,1), l(h,p,1), l(p,f,1), l(p,h,1)$	2.
		$ol(b,1), ol(f,1), ol(h,1), ol(p,1)$	3.
		$or(b,1), or(f,1), or(h,1), or(p,1)$	3.
$\Phi_{\mathcal{P}_3}\uparrow 2$	$ln(f,p,1)$	$ln(b,f,1), ln(b,h,1), ln(b,p,1), ln(f,b,1),$	4.
		$ln(f,h,1), ln(h,b,1), ln(h,f,1)ln(h,p,1)$	4.
		$ln(p,f,1), ln(p,h,1)$	4.
$\Phi_{\mathcal{P}_3}\uparrow 3$	$ln(f,p,2)$		5.
	$ol(p,2), or(f,2)$	$ol(b,2), ol(f,2), ol(h,2), or(b,2), or(h,2), or(p,2)$	6.
		$l(b,h,2), l(b,p,2), l(f,b,2), l(f,h,2)$	7.,8.
		$l(f,p,2), l(h,p,2), l(p,b,2)$	7.,8.
$\Phi_{\mathcal{P}_3}\uparrow 4$		$ln(b,h,2), ln(b,p,2), ln(f,b,2), ln(f,h,2)$	4.
	$ln(h,b,2)$	$ln(h,p,2), ln(p,b,2), ln(p,f,2)$	4.
	$ln(f,p,3)$		5.
	$ol(p,3), or(f,3)$		6.
		$l(b,f,2), l(h,f,2), l(p,h,2)$	7.,8.
$\Phi_{\mathcal{P}_3}\uparrow 5$	$ln(h,b,3)$	$ln(b,p,3), ln(f,b,3), ln(f,h,3), ln(h,p,3)$	5.
		$ln(h,d,2), ln(b,f,2), ln(h,f,2), ln(p,h,2)$	5.
	$ol(b,3), or(h,3)$		6.
	$l(f,h,3)$	$l(h,p,3), l(p,f,3)$	7.,8.
	$left(f,p)$		9.
$\Phi_{\mathcal{P}_3}\uparrow 6$		$ln(p,b,3), ln(p,f,3)$	5.
		$ol(f,3), ol(h,3), or(b,3), or(p,3)$	6.
	$l(p,h,3)$	$l(b,h,3), l(b,p,3), l(f,p,3),$	7,8.
		$l(h,b,3), l(h,f,3), l(p,b,3), ln(h,f,3)$	7.,8.
	$left(h,b)$		9.
	$right(p,f)$		11.
$\Phi_{\mathcal{P}_3}\uparrow 7$	$ln(p,h,3)$	$ln(b,h,3)$	4.
		$l(b,f,3)$	7.,8.
	$right(b,h)$		11.
$\Phi_{\mathcal{P}_3}\uparrow 8$		$ln(b,f,3)$	5.
	$left(p,h)$		9.
$\Phi_{\mathcal{P}_3}\uparrow 9$	$left(f,h)$, $\boldsymbol{left(p,b)}$		10.
	$right(h,p)$		11.
$\Phi_{\mathcal{P}_3}\uparrow 10$	$left(f,b)$		10.
	$right(b,p), right(h,f)$		11.
$\Phi_{\mathcal{P}_3}\uparrow 11$	$right(b,f)$		11.

Table 6.2.: The least model of the weak completion of $\mathcal{P}_3$ is computed by iterating $\Phi_{\mathcal{P}_3}$ until the least fixed point is reached. In each iteration only atoms are listed which appear in $I^\top$ and $I^\bot$ for the first time. # lists the clauses responsible for adding an atom to $I^\top$ or $I^\bot$. The atom in bold confirms the conclusion: The Porsche is to the left of the Beetle.

Höps [2014] has shown that although the logic programs here contain positive cycles, the correspondence between the Weak Completion Semantics and the Well-founded Semantics, which we discussed in Chapter 3, can be preserved and, hence, preferred mental models can also be computed within state-of-the-art reasoning systems based on answer set programming like CLINGO [Gebser, Kaminski, Kaufmann, and Schaub, 2014]. Thus, large scale applications seem to be feasible.

7. Quantified Statements

In a recent meta-analysis, Khemlani and Johnson-Laird [2012] showed that the conclusions drawn by humans in psychological experiments about syllogistic reasoning are not the conclusions predicted by classical first-order logic. We propose an alternative approach on modeling syllogisms. The chapter is structured as follows: After an introduction on syllogistic reasoning in the next section, we apply four principles in developing a logical form for the representation of syllogisms in Section 7.2. Section 7.3 shows how the four syllogistic moods can be represented in logic programs and how entailments can be understood under the Weak Completion Semantics. By means of three examples, we present in Section 7.4 the predictions under the Weak Completion Semantics. Finally, the last section compares these results with the results of FOL and three cognitive theories.[1]

7.1. Introduction

The way of how humans ought to reason correctly about syllogisms has already been investigated by Aristotle. A *syllogism* consists of two quantified statements using some of the four quantifiers *all* (A), *no* (E), *some* (I), and *some are not* (O) about sets of entities which we denote in the following by the predicate symbols a, b and c. The letters in brackets, A, E, I and O are the classical abbreviations derived from the first two vowels of the Latin words **a**ff**i**rmo and n**e**g**o** meaning affirm and deny, respectively. Consider the following two premises:

First Premise	'*some a are b*'	(IE1)
Second Premise	'*no b are c*'	

What can we conclude about the relation between a and c? The classical first-order logical consequence from these so-called premises is '*some a are not c*'. The first two premises together with a consequence that follows classical logically is called a *valid syllogism*. Otherwise it is called an *invalid syllogism*. The four quantifiers and their formalization in FOL are given in Table 7.1. The entities can appear in four different orders called *figures* as shown in Table 7.2. Hence, a problem consisting of two premises

[1] The original idea of this chapter has been published in [Dietz, Hölldobler, and Ragni, 2015d, Costa, Dietz, Hölldobler, and Ragni, 2016]. An extended version thereof is under review [Costa, Dietz, Hölldobler, and Ragni, 2017a].

Mood	Natural Language	FOL	Short
affirmative universal (A)	*all a are b*	$\forall X(a(X) \rightarrow b(X))$	Aab
affirmative existential (I)	*some a are b*	$\exists X(a(X) \wedge b(X))$	Iab
negative universal (E)	*no a are b*	$\forall X(a(X) \rightarrow \neg b(X))$	Eab
negative existential (O)	*some a are not b*	$\exists X(a(X) \wedge \neg b(X))$	Oab

Table 7.1.: The four syllogistic moods together with their logical formalization.

	Figure 1	Figure 2	Figure 3	Figure 4
First Premise	a-b	b-a	a-b	b-a
Second Premise	b-c	c-b	c-b	b-c

Table 7.2.: The four figures used in syllogistic reasoning.

can be completely specified by the quantifiers of the first and second premise and the figure. The example discussed above is denoted by IE1, where I stands for the mood of the first premise (some), E stands for the mood of the second premise (no) and 1 for its figure (a-b, b-c).

Altogether, there are 64 syllogisms and, if formalized in FOL, we can compute their classical logical consequence. Nevertheless, Khemlani and Johnson-Laird's [2012] meta-analysis based on six experiments has shown that humans do not only systematically deviate from the predictions of FOL, but from any other of at least 12 cognitive theories. In the case of IE1, besides the above mentioned logical consequence, a significant number of humans answered '*no a are c*', which does not follow from IE1 in FOL.

The predictions of the theories FOL, PSYCOP, Verbal, and Mental Models for the syllogisms OA4, EA2, and AA4 and those of the participants, taken from Khemlani and Johnson-Laird, are depicted in Table 7.3, where the participants were 156 high school to university students. FOL and the other three cognitive theories make different predictions. In particular, each theory provides at least one prediction which is correct with respect to classical FOL and provides an additional prediction, which is false with respect to classical FOL. Currently, the best overall results are achieved by the Verbal Models Theory [Polk and Newell, 1995], which predicts 84% of the participants responses, closely followed by the Mental Model Theory [Johnson-Laird, 1983] with 83%, whereas PSYCOP [Rips, 1994] only predicts 77% of the participants' responses. The conclusions depicted in the second column of Table 7.3 refer to the significant percentage of participants, which is the number of participants who chose the particular conclusion, which was too high for the conclusion to be chosen randomly. The threshold for the percentage to be significant is determined as follows: Given that there are nine different possible

	Participants	FOL	PSYCOP	Verbal Models	Mental Models
OA4	Oca	Oca	Oca, Ica, Iac	Ocs, NVC	Oca, Oac, NVC
EA2	Eac, Eca	Eac, Eca Oac, Oca	Eac, Eca Oac, Oca	Eca	Eac, Eca
AA4	Aac, NVC	Iac, Ica	Iac, Ica	NVC, Aca	Aca, Aac, Iac, Ica

Table 7.3.: The conclusions drawn by a significant percentage of participants are highlighted in gray and compared to the predictions of the theories FOL, PSYCOP, Verbal, and Mental Models for the syllogisms OA4, EA2, and AA4. NVC stands for *no valid conclusion*.

conclusions, the chance that a conclusion has been chosen randomly is $1/9 = 11.1\%$. A binomial test shows that if a conclusion is drawn in more than 16% of the cases by the participants it is unlikely that is has been chosen by just random guesses. The statistical analysis is elaborately explained in [Khemlani and Johnson-Laird, 2012].

In the sequel, we investigate whether the Weak Completion Semantics is competitive in syllogistic reasoning and how it performs with respect to the cognitive theories considered by Khemlani and Johnson-Laird. First, we develop four principles for the representation of syllogisms and show how to model them under the Weak Completion Semantics. Afterwards we compare our results with the results of FOL, the syntactic rule based theory PSYCOP [Rips, 1994], the Verbal Model Theory [Polk and Newell, 1995] and the Mental Model Theory [Johnson-Laird, 1983].[2] The last two theories are model-based and performed the best in Khemlani and Johnson-Laird's meta-analysis.

7.2. Five Principles

We will now introduce five principles, which we apply for developing a logical form for the representation of syllogisms. The first principle, licenses for inferences, has already been applied in the previously presented human reasoning tasks in Chapter 5. The second principle, negation by transformation, is an idea used in the area of Logic Programming as a mechanism to represent negative consequences. The last three principles, existential import and Gricean implicature, unknown generalization and blocking conclusions through double negation, are assumptions motivated from findings in Cognitive Science.

[2] `http://mentalmodels.princeton.edu/models/mreasoner/`

7.2.1. Licenses for Inferences (licenses)

As already discussed in Section 5.1.1, Stenning and van Lambalgen [2008] proposed to formalize conditionals in human reasoning not by inferences straight away, but rather by *licenses for inferences*. For instance, the conditional '*if* $p(X)$ *then* $q(X)$' is represented by the program, which consists of the following clauses:

$$\begin{array}{rcl} q(X) & \leftarrow & p(X) \wedge \neg ab_{pq}(X). \\ ab_{pq}(X) & \leftarrow & \bot. \end{array}$$

where the first clause states that '$q(X)$ *if* $p(X)$ *and* $\neg ab_{pq}(X)$'. The closed-world assumption with respect to the abnormality predicate, $ab_{pq}(X)$, is represented by the second clause. The assumption $ab_{pq}(X) \leftarrow \bot$ can be understood as '*nothing is abnormal for* X *with respect to the first clause*' (if nothing else is known).

We call this principle *licenses for inferences* and will refer to it by the following abbreviation in brackets: (licenses).

7.2.2. Negation by Transformation (transformation)

The logic programs we consider under the Weak Completion Semantics do not allow heads of clauses to be negative literals. In order to represent a negative conclusion $\neg p(X)$, we introduce an auxiliary formula $p'(X)$ together with the clause

$$p(X) \leftarrow \neg p'(X)$$

and the integrity constraint $\mathrm{U} \leftarrow p(X) \wedge p'(X)$. This is a widely used technique in logic programming. Together with the principle (licenses) introduced in Section 7.2.1, this additional clause is extended by the following two clauses:

$$\begin{array}{rcl} p(X) & \leftarrow & \neg p'(X) \wedge \neg ab_{npp}(X). \\ ab_{npp}(X) & \leftarrow & \bot. \end{array}$$

Note that the second clause represents the closed world assumption with respect to $ab_{npp}(X)$. The weak completion of both clauses is then

$$\begin{array}{rcl} p(X) & \leftrightarrow & \neg p'(X) \wedge \neg ab_{npp}(X). \\ ab_{npp}(X) & \leftrightarrow & \bot. \end{array}$$

Additionally, the integrity constraint

$$\mathrm{U} \leftarrow p(X) \wedge p'(X).$$

states that an object cannot belong to both, p and p'.

We call this principle *negation by transformation* and will refer to it by the following abbreviation in brackets: (transformation).

7.2.3. Existential Import and Gricean Implicature (import)

Humans understand quantifiers differently due to a pragmatic understanding of language. For instance, in natural language, we normally do not quantify over things that do not exist. Consequently, '*for all*' implies '*there exists*'. This appears to be in line with human reasoning and has been called the *Gricean implicature* [Grice, 1975]. This corresponds to what sometimes in literature is also called *existential import* and assumed by several theories like the theory of mental models [Johnson-Laird, 1983] or mental logic [Rips, 1994]. Likewise, Stenning and van Lambalgen [2008] have shown that humans require existential import for a conditional to be true.

Furthermore, as mentioned by Khemlani and Johnson-Laird [2012], the quantifier '*some a are b*' often implies that '*some a are not b*', which again is implied by the Gricean implicature: Someone would not state '*some a are b*' if that person knew that '*all a are b*'. As the person does not say '*all a are b*', but '*some a are b*' instead, we assume that '*not all a are b*', which in turn implies '*some a are not b*'.

We call this principle *existential import and Gricean implicature* and will refer to it by the following abbreviation in brackets: (import).

7.2.4. Unknown Generalization (unknownGen)

Humans seem to distinguish between '*some y are z*' and '*some z are y*', as the results reported by Khemlani and Johnson-Laird [2012] show. Nevertheless, if we would represent '*some y are z*' by $\exists X(y(X) \wedge z(X))$ then this is semantically equivalent to $\exists X(z(X) \wedge y(X))$ because conjunction is commutative in FOL. Likewise, humans seem to distinguish between '*some y are z*' and '*all y are z*', as we have already discussed in Section 7.2.3. Accordingly, if we only observe that an object o belongs to y and z then we do not want to conclude both, '*some y are z*' and '*all y are z*'.

In order to distinguish between '*some y are z*' and '*all y are z*', we introduce the following principle: If we know that '*some y are z*', then there must not only be an object o_1, which belongs to y and z (by Gricean implicature), but there must be another object o_2, which belongs to y and for which it is unknown whether it belongs to z .

We call this principle *unknown generalization* and will refer to it by the following abbreviation in brackets: (unknownGen).

7.2.5. No Derivation through Double Negation (doubleNeg)

Under Weak Completion Semantics, a positive conclusion can be derived from double negation within two conditionals. Consider the following two conditionals with each one having a negative premise:

If not a, then b.
If not b then c.

Additionally, assume that a is true. Let us encode the two conditionals and the fact that a is true as a program consisting of the following three clauses:

$$\begin{array}{lll} b & \leftarrow & \neg a. \\ c & \leftarrow & \neg b. \\ a & \leftarrow & \top. \end{array}$$

Its weak completion is

$$\begin{array}{lll} b & \leftrightarrow & \neg a. \\ c & \leftrightarrow & \neg b. \\ a & \leftrightarrow & \top. \end{array}$$

Its least model is

$$\langle \{a, c\}, \{b\} \rangle,$$

where a and c are true: a is true because it is a fact and c is true by the negation of b. b is derived false because the negation of a is false. This example shows that under the Weak Completion Semantics, a positive conclusion (c being true) can be derived from two clauses, through double negation. Yet, it appears to be the case that humans do not reason in such a way: Considering the results of the participants' responses in [Khemlani and Johnson-Laird, 2012],[3] they seem not to draw conclusions through double negatives. Accordingly, we block them through abnormalities, which we explain in more detail by an example in Section 7.3.2.

We call this principle *no derivation through double negation* and will refer to it by the following abbreviation in brackets: (doubleNeg).

7.3. Representation as Logic Programs

Based on the first five principles of the previous section, we encode the quantified statements in logic programs. The programs will be specified using the predicates y and z and depending on the figures shown in Table 7.2, where yz can be replaced by ab, ba, cb or bc.

[3] For instance, consider their results for the cases EE1, EE2, EE3 and EE4 in Appendix D.

7.3.1. All y are z (Ayz)

'*All y are z*' is represented by the program $\mathcal{P}_{Ayz}$, which consists of the following clauses:

$$\begin{array}{rcll} z(X) & \leftarrow & y(X) \wedge \neg ab_{yz}(X). & \text{(licenses)} \\ ab_{yz}(X) & \leftarrow & \bot. & \text{(licenses)} \\ y(o) & \leftarrow & \top. & \text{(import)} \end{array}$$

The first two clauses are obtained by applying the principle of using licenses for inferences. The last clause follows by the principle of existential import and Gricean implicature, where o is the object which we assume to exist for y. The least model of the weak completion of $\mathcal{P}_{Ayz}$ is

$$\langle \{y(o), z(o)\}, \{ab_{yz}(o)\} \rangle.$$

7.3.2. No y is z (Eyz)

According to Table 7.1, under FOL mood is represented as follows:

$$\forall X \quad (y(X) \quad \rightarrow \quad \neg z(X)).$$

As contraposition holds under the Weak Completion Semantics [Kencana Ramli, 2009], this formula is equivalent to

$$\forall X \quad (z(X) \quad \rightarrow \quad \neg y(X)).$$

Given this formula[4] and the three principles, licenses for inferences, existential import and Gricean implicature and negation by transformation, introduced in Section 7.2.1, 7.2.3 and 7.2.2, the logic program $\mathcal{P}_{Eyz}$ consists of the following clauses:

$$\begin{array}{rcll} y'(X) & \leftarrow & z(X) \wedge \neg ab_{zny}(X). & \text{(transformation \& licenses)} \\ ab_{zny}(X) & \leftarrow & \bot. & \text{(licenses)} \\ y(X) & \leftarrow & \neg y'(X) \wedge \neg ab_{nyy}(X). & \text{(transformation \& licenses)} \\ z(o) & \leftarrow & \top. & \text{(import)} \\ ab_{nyy}(o) & \leftarrow & \bot. & \text{(licenses \& doubleNeg)} \end{array}$$

In addition, we have the following integrity constraint:

$$\mathrm{U} \leftarrow y(X) \wedge y'(X). \qquad \text{(transformation)}$$

The first two clauses in $\mathcal{P}_{Eyz}$ are obtained by applying the principle of using licenses for inferences, where y' is an auxiliary predicate symbol used to denote the negation

[4] We decided to model the logic program according to the representation of $\forall X(z(X) \rightarrow \neg y(X))$, because we can use this same representation in the following Chapter, when abduction is required.

of y. This auxiliary predicate symbol is related to y by the third clause applying the principle of negation by transformation. In addition, this principle enforces the integrity constraint. The fourth clause of $\mathcal{P}_{Eyz}$ follows by the principle of Gricean implicature. The least model of the weak completion of $\mathcal{P}_{Eyz}$ is

$$\langle\{z(o), y'(o)\}, \{ab_{zny}(o), ab_{nyy}(o), y(o)\}\rangle.$$

Note that the last clause in $\mathcal{P}_{Eyz}$ cannot be generalized to all X, because otherwise we allow conclusions by double negatives. The fifth principle discussed in Section 7.2.5 states that we should block conclusions through double negatives.

The following example shows what exactly that means and why this is undesirable: Consider the case where additionally to the clauses in $\mathcal{P}_{Eyz}$, there is some o' for which $z(o')$ is false due to some clause being part of another premise. In this case, the first clause in $\mathcal{P}_{Eyz}$ will enforce the falsehood of $y'(o')$. Now, if $ab_{nyy}(X) \leftarrow \bot$ would be in $\mathcal{P}_{Eyz}$ (instead of $ab_{nyy}(o) \leftarrow \bot$), then also $ab_{nyy}(o')$ would be false and, consequently, by the third clause in $\mathcal{P}_{Eyz}$, $y(o')$ would be true. In other words, $y(o')$ would follow by the negation of $z(o')$, which in turn would be responsible for the negation of $y'(o')$. Even though this might be logically reasonable, the empirical results indicate that participants do not infer conclusions based on double negation. Therefore, we decide to restrict $ab_{nyy}(o) \leftarrow \bot$ to the objects occurring in $\mathcal{P}_{Eyz}$.

7.3.3. Some y are z (Iyz)

'*Some y are z*' is represented by the program $\mathcal{P}_{Iyz}$, which consists of the following clauses:

$$\begin{array}{rcll} z(X) & \leftarrow & y(X) \wedge \neg ab_{yz}(X). & \text{(licenses)} \\ ab_{yz}(o_1) & \leftarrow & \bot. & \text{(unknownGen \& licenses)} \\ y(o_1) & \leftarrow & \top. & \text{(import)} \\ y(o_2) & \leftarrow & \top. & \text{(unknownGen)} \end{array}$$

The first two clauses are again obtained by the principle of using licenses for inferences. The abnormality predicate is restricted to the object o_1, which is assumed to exist by the principle of Gricean implicature, represented by the third clause. The fourth clause is obtained by the principle of unknown generalization. The least model of $wc\mathcal{P}_{Iyz}$ is

$$\langle\{y(o_1), y(o_2), z(o_1)\}, \{ab_{yz}(o_1)\}\rangle.$$

Note that nothing about $ab_{yz}(o_2)$ is stated in $\mathcal{P}_{Iyz}$. Accordingly, $z(o_2)$ stays unknown in the least model.

7.3.4. Some y are not z (Oyz)

'*Some y are not z*' is represented by the program $\mathcal{P}_{Oyz}$ which consists of the following clauses:

$$
\begin{array}{rcll}
z'(X) & \leftarrow & y(X) \wedge \neg ab_{ynz}(X). & \text{(transformation \& licenses)} \\
ab_{ynz}(o_1) & \leftarrow & \bot. & \text{(unknownGen \& licenses)} \\
z(X) & \leftarrow & \neg z'(X) \wedge \neg ab_{nzz}(X). & \text{(transformation \& licenses)} \\
y(o_1) & \leftarrow & \top. & \text{(import)} \\
y(o_2) & \leftarrow & \top. & \text{(unknownGen)} \\
ab_{nzz}(o_1) & \leftarrow & \bot. & \text{(doubleNeg \& licenses)} \\
ab_{nzz}(o_2) & \leftarrow & \bot. & \text{(doubleNeg \& licenses)}
\end{array}
$$

In addition, we have the following integrity constraint:

$$U \leftarrow z(X) \wedge z'(X). \qquad \text{(transformation)}$$

The first four clauses as well as the integrity constraint are derived as in the program $\mathcal{P}_{Eyz}$ except that object o_1 is used instead of o and ab_{ynz} is restricted to o_1 like in $\mathcal{P}_{Iyz}$. The fifth clause of $\mathcal{P}_{Oyz}$ is obtained by the principle of unknown generalization. The last two clauses are again not generalized to all objects for the same reason as previously discussed in Section 7.3.2 for the representation of E: The generalization of ab_{nzz} to all objects can lead to conclusions through double negation, in case there is a second premise. The least model of $wc\mathcal{P}_{Oyz}$ is

$$\langle \{y(o_1), y(o_2), z'(o_1)\}, \{ab_{ynz}(o_1), ab_{nzz}(o_1), ab_{nzz}(o_2), z(o_1)\}\rangle.$$

7.3.5. Entailment of Quantified Statements

We have not yet defined when a syllogism holds given a program $\mathcal{P}$. These definitions will be developed in this section.

One should observe that Khemlani and Johnson-Laird [2012] do not propose a formal definition for the entailment of the moods. They use first-order theory as a normative theory, i.e. they test if the conclusions drawn by the participants are correct with respect to a first-order representation of a mood. In the following, we define the entailment of the moods under the Weak Completion Semantics, where yz will later be replaced by ac or ca.[5]

A straightforward definition of whenever '*All y are z*' holds given a program $\mathcal{P}$, is, whenever exactly the things, which have been stated in $\mathcal{P}_{Ayz}$, are true in its least model: $\mathcal{P} \models' Ayz$ iff there exists an object o such that $\mathcal{P} \models_{wcs} y(o)$ and for all objects o we find that if $\mathcal{P} \models_{wcs} y(o)$ then $\mathcal{P} \models_{wcs} z(o) \wedge \neg ab_{yz}(o)$. The existence of an object o belonging

[5] Participants were only asked to draw conclusions about the relation between a and c.

to y is due to the principle of Gricean implicature. Moreover, all objects belonging to y must belong to z. The requirement that $\neg ab_{yz}(o)$ is also entailed is a technical one which is based on the principle of licenses for inferences. Similarly, according to the specifications in Section 7.3, the other entailments for $\mathcal{P} \models' Eyz$, $\mathcal{P} \models' Iyz$ and $\mathcal{P} \models' Oyz$ could be defined as follows:

- $\mathcal{P} \models' Eyz$ iff there exists an object o such that $\mathcal{P} \models_{wcs} z(o)$ and for all objects o we find that if $\mathcal{P} \models_{wcs} z(o)$ then $\mathcal{P} \models_{wcs} y'(o) \wedge \neg y(o) \wedge \neg ab_{znu}(o) \wedge \neg ab_{nyy}(o)$.
- $\mathcal{P} \models' Iyz$ iff there exists an object o_1 such that $\mathcal{P} \models_{wcs} y(o_1) \wedge z(o_1) \wedge \neg ab_{yz}(o_1)$ and there exists an object o_2 such that $\mathcal{P} \models_{wcs} y(o_2)$ and $\mathcal{P} \not\models_{wcs} z(o_2) \wedge \neg ab_{yz}(o_2)$.
- $\mathcal{P} \models' Oyz$ iff there exists an object o_1 such that $\mathcal{P} \models_{wcs} y(o_1) \wedge z'(o_1) \wedge \neg z(o_1) \wedge \neg ab_{ynz}(o_1) \wedge \neg ab_{nzz}(o_1)$ and there exists an object o_2 such that $\mathcal{P} \models_{wcs} y(o_2)$ and $\mathcal{P} \not\models_{wcs} z'(o_2) \wedge \neg z(o_2) \wedge \neg ab_{y}nz(o_2) \wedge \neg ab_{nzz}(o_2)$.

These definitions are too strict as soon as we want to infer conclusions about relations, which are not explicitly stated. For instance, consider again Table 7.2, where the four possible figures are given. The first premise states a direct relation between a and b and the second premise states a direct relation between b and c. Yet, in the meta-analysis, participants were asked to draw conclusions about the relation between a and c.[6]

We obtain the following alternative definition when $\mathcal{P}$ entails a certain conclusion:

- $\mathcal{P} \models Ayz$ iff there exists an object o such that $\mathcal{P} \models_{wcs} y(o)$ and for all objects o we find that if $\mathcal{P} \models_{wcs} y(o)$ then $\mathcal{P} \models_{wcs} z(o)$.
- $\mathcal{P} \models Eyz$ iff there exists an object o such that $\mathcal{P} \models_{wcs} z(o)$ and for all objects o we find that if $\mathcal{P} \models_{wcs} z(o)$ then $\mathcal{P} \models_{wcs} \neg y(o)$.
- $\mathcal{P} \models Iyz$ iff there exists an object o_1 such that $\mathcal{P} \models_{wcs} y(o_1) \wedge z(o_1)$ and there exists an object o_2 such that $\mathcal{P} \models_{wcs} y(o_2)$ and $\mathcal{P} \not\models_{wcs} z(o_2)$.
- $\mathcal{P} \models Oyz$ iff there exists an object o_1 such that $\mathcal{P} \models_{wcs} y(o_1) \wedge \neg z(o_1)$ and there exists an object o_2 such that $\mathcal{P} \models_{wcs} y(o_2)$ and $\mathcal{P} \not\models_{wcs} \neg z(o_2)$.

In case we can not conclude any of these moods, then no valid conclusion is entailed, denoted as $\mathcal{P} \models \mathsf{NVC}$. Note that, by this last definition, we cannot entail NVC together with some other conclusion, i.e. NVC is disjoint with any other conclusion. In Section 7.5, where we present our results, we will discuss this aspect in more detail.

Note that yet an alternative definition could simply be the entailment under FOL:

- $\mathcal{P} \models_{\mathsf{FOL}} Ayz$ iff for all objects o we find that if $\mathcal{P} \models_{wcs} y(o)$ then $\mathcal{P} \models_{wcs} z(o)$.
- $\mathcal{P} \models_{\mathsf{FOL}} Eyz$ iff for all objects o we find that if $\mathcal{P} \models_{wcs} z(o)$ then $\mathcal{P} \models_{wcs} \neg y(o)$.
- $\mathcal{P} \models_{\mathsf{FOL}} Iyz$ iff there exists an object o such that $\mathcal{P} \models_{wcs} y(o) \wedge z(o)$.

[6] When we will discuss syllogism OA4 in Section 7.4.2, it will become clear why the $\models'$ entailment is not appropriate for the conclusions with respect to the relation between a and c.

- $\mathcal{P} \models_{\mathsf{FOL}} Oyz$ iff there exists an object o such that $\mathcal{P} \models_{wcs} y(o) \wedge \neg z(o)$

Similarly to the previous case, if we can not conclude any of these moods, then no valid conclusion is entailed, denoted as $\mathcal{P} \models_{\mathsf{FOL}} \mathsf{NVC}$. In Section 7.5, we will discuss whether the results of both entailments differ.

7.4. Predictions under the Weak Completion Semantics

We accomplished an average of 85% accuracy in our predictions, when we apply the representation for the quantified statements of Section 7.3 and the just introduced entailment rules. In nine cases we had a perfect match with the answers given by the participants. In 30 cases the match was 89% and in 20 cases the match was 78%. The five cases left had a match of 67%.

We explain now how the accuracy of the predictions is computed in general. After that, we will show the logic program representation of three syllogisms by combining the proposed representations with respect to the figures in Table 7.2. In doing so, we replace yz by ab, ba, cb or bc. In addition, we may need to rename objects such that different objects are referred to in the representations of different syllogisms. Thereafter, we compute the least model of the weak completion of the obtained programs and check which conclusions hold in this model. We compare our results with the data from the psychological experiments in [Khemlani and Johnson-Laird, 2012].

7.4.1. Accuracy of Predictions

Note that in Khemlani and Johnson-Laird's experiments, people were asked to infer conclusions about a and c from a syllogism built according to the figures in Table 7.2. Therefore, in our predictions we only consider entailment of syllogisms between a and c.

We have nine different answer possibilities for each of the 64 syllogisms:

$$\mathsf{Aac}, \quad \mathsf{Eac}, \quad \mathsf{Iac}, \quad \mathsf{Oac}, \quad \mathsf{Aca}, \quad \mathsf{Eca}, \quad \mathsf{Ica}, \quad \mathsf{Oca} \quad \text{and} \quad \mathsf{NVC}.$$

For every syllogism, we define a list of length 9 for the predictions of the Weak Completion Semantics, where the first element represents Aac, the second element represents Eac, and so forth. When Aac is predicted under the Weak Completion Semantics for a given syllogism, then the value of the first element of this list is a 1, otherwise it is a 0, and the same holds for the other eight elements in the list (representing the other eight answer possibilities). Analogously, for every syllogism we define a list of the participants' conclusions of length 9 containing either 1 or 0 for all nine answer possibilities, depending on whether the majority of the participants concluded Aac, Eac, and so forth.

For each syllogism we then simply compare each element of both lists as follows, where i is the ith element of both lists:

$$\text{COMP}(i) = \begin{cases} 1 & \text{if both lists have the same value for the } i\text{th element} \\ 0 & \text{otherwise} \end{cases}$$

The matching percentage of this syllogism is then computed by $\sum_{i=1}^{9} \text{COMP}(i)/9$. Note that the percentage of the match does not only take into account when the weak completion semantics correctly predicts a conclusion, but also whenever it correctly rejects a conclusion, i.e. when the Weak Completion Semantics does not predict a conclusion. The average percentage of accuracy in general is then simply the average of the matching percentage of all 64 syllogisms.

7.4.2. OA4 - Perfect Match (100%)

The syllogism OA4 is obtained by combining the last and the first mood in Table 7.1 according to figure 4 in Table 7.2. It can be read as:

First Premise	Oba	'*some b are not a*'
Second Premise	Abc	'*all b are c*'

The program $\mathcal{P}_{\mathsf{OA4}}$ representing the two premises is obtained as the union of the programs $\mathcal{P}_{Oba}$ (obtained from $\mathcal{P}_{Oyz}$ by replacing y and z by b and a, respectively) and $\mathcal{P}_{Abc}$ (obtained from $\mathcal{P}_{Ayz}$ by replacing y and z by b and c, respectively). In addition, the constant o occurring in $\mathcal{P}_{Abc}$ has been replaced by o_3. $\mathcal{P}_{\mathsf{OA4}}$ consists of the following clauses:[7]

$$\begin{aligned}
a'(X) &\leftarrow b(X) \wedge \neg ab_{bna}(X). && \text{(transformation \&licenses)} \\
ab_{bna}(o_1) &\leftarrow \bot. && \text{(unknownGen \& licenses)} \\
a(X) &\leftarrow \neg a'(X) \wedge \neg ab_{naa}(X). && \text{(transformation \& licenses)} \\
b(o_1) &\leftarrow \top. && \text{(import)} \\
b(o_2) &\leftarrow \top. && \text{(unknownGen)} \\
ab_{naa}(o_1) &\leftarrow \bot. && \text{(doubleNeg \& licenses)} \\
ab_{naa}(o_2) &\leftarrow \bot. && \text{(doubleNeg \& licenses)} \\
c(X) &\leftarrow b(X) \wedge \neg ab_{bc}(X). && \text{(licenses)} \\
ab_{bc}(X) &\leftarrow \bot. && \text{(licenses)} \\
b(o_3) &\leftarrow \top. && \text{(import)}
\end{aligned}$$

In addition, we have the following integrity constraint:

$$\text{U} \leftarrow a(X) \wedge a'(X) \qquad \text{(transformation)}$$

[7] We omit a, b and c in the index of $\mathcal{P}_{\mathsf{OA4}}$, as they are determined by the number of the figure.

The least model of the weak completion of $\mathcal{P}_{\mathsf{OA4}}$ is

$$\langle\{b(o_1),\ b(o_2),\ b(o_3),\ ab_{ca}(o_1),\ a'(o_1),\ c(o_1),\ c(o_2),\ c(o_3)\},\\ \{ab_{bna}(o_1),\ ab_{naa}(o_1),\ ab_{bc}(o_1),\ ab_{bc}(o_2),\ ab_{bc}(o_3),\ a(o_1)\}\rangle.$$

This model entails only the conclusion '*some c are not a*' (Oca): There exists an object, o_1, such that $\mathcal{P}_{\mathsf{OA4}} \models_{wcs} c(o_1) \wedge \neg a(o_1)$ and there exists an object, o_2, such that $\mathcal{P}_{\mathsf{OA4}} \models_{wcs} c(o_2)$ and $\mathcal{P}_{\mathsf{OA4}} \not\models_{wcs} \neg a(o_2)$.

Consider again the initial definition for the entailment of syllogisms as discussed in Section 7.3.5. Accordingly, $\mathcal{P} \models' Oca$ iff there exists an object o_1 such that $\mathcal{P} \models_{wcs} c(o_1) \wedge a'(o_1) \wedge \neg a(o_1) \wedge \neg ab_{cna}(o_1) \wedge \neg ab_{naa}(o_1)$ and there exists an object o_2 such that $\mathcal{P} \models_{wcs} c(o_2)$ and $\mathcal{P} \not\models_{wcs} a'(o_2) \wedge \neg a(o_2) \wedge \neg ab_{cna}(o_2) \wedge \neg ab_{naa}(o_2)$. This example shows that the entailment $\models'$ is not appropriate to derive some relation between a and c: $\mathcal{P}_{\mathsf{OA4}} \not\models' \mathsf{O}ac$ because $\mathcal{P}_{\mathsf{OA4}} \not\models_{wcs} \neg ab_{cna}(o_1)$.

7.4.3. EA2 - Worst Match (67%)

EA2 is one of the syllogisms with the lowest match (67%):

First Premise	Eab	'*No b are a*'
Second Premise	Acb	'*All c are b*'

The program $\mathcal{P}_{\mathsf{EA2}}$ representing the two premises is obtained as the union of the programs $\mathcal{P}_{\mathsf{E}ba}$ (obtained from $\mathcal{P}_{Eyz}$ by replacing y and z by b and a, respectively) and $\mathcal{P}_{\mathsf{A}cb}$ (obtained from $\mathcal{P}_{Eyz}$ by replacing y and z by c and b, respectively). In addition, the constant o occurring in $\mathcal{P}_{\mathsf{E}ba}$ and $\mathcal{P}_{\mathsf{A}cb}$ has been replaced by o_1 and o_2, respectively. $\mathcal{P}_{\mathsf{EA2}}$ consists of the following clauses:

$$\begin{aligned}
b'(X) &\leftarrow a(X) \wedge \neg ab_{anb}(X). && \text{(transformation \& licenses)}\\
ab_{anb}(X) &\leftarrow \bot. && \text{(licenses)}\\
b(X) &\leftarrow \neg b'(X) \wedge \neg ab_{nbb}(X). && \text{(transformation \& licenses)}\\
a(o_1) &\leftarrow \top. && \text{(import)}\\
ab_{nbb}(o_1) &\leftarrow \bot. && \text{(licenses \&doubleNeg)}\\
b(X) &\leftarrow c(X) \wedge \neg ab_{cb}(X). && \text{(licenses)}\\
ab_{cb}(X) &\leftarrow \bot. && \text{(licenses)}\\
c(o_2) &\leftarrow \top. && \text{(import)}
\end{aligned}$$

Furthermore, we have the following integrity constraint:

$$\mathrm{U} \leftarrow b(X) \wedge b'(X). \qquad \text{(transformation)}$$

The least model of the weak completion of $\mathcal{P}_{\mathsf{EA2}}$ is

$$\langle\{a(o_1),\ c(o_2),\ b'(o_1),\ b(o_2)\},\\ \{ab_{anb}(o_1),\ ab_{nbb}(o_1),\ ab_{anb}(o_2), ab_{nbb}(o_2),\ ab_{cb}(o_1),\ ab_{cb}(o_2)\}\rangle.$$

This model does not entail any conclusion between a and c, therefore our prediction is NVC. Participants concluded both Eac and Eca.

Even though none of the entailed conclusions predicted by the weak completion semantics matches the participants' answers, how could we possibly reach a match of 67%? The reason is that we also take into account the correct rejections, i.e. the conclusions that are not entailed by the model, as has been explained in Section 7.4.1

7.4.4. AA4 - Partial Match (78%)

For AA4 we got an accuracy of 78% in our prediction. This syllogism combines two premises in the first mood according to figure 4. It can be read as:

First Premise	Aba	'*All b are a*'
Second Premise	Abc	'*All b are c*'

The program $\mathcal{P}_{\mathsf{AA4}}$ representing the two premises is obtained as the union of programs $\mathcal{P}_{\mathsf{A}ba}$ (obtained from $\mathcal{P}_{Ayz}$ by replacing y and z by b and a, respectively) and $\mathcal{P}_{\mathsf{A}bc}$ (obtained from $\mathcal{P}_{Ayz}$ by replacing y and z by b and c, respectively). In addition, the constant o occurring in $\mathcal{P}_{\mathsf{A}ba}$ and $\mathcal{P}_{\mathsf{A}bc}$ have been replaced by o_1 and o_2, respectively. $\mathcal{P}_{\mathsf{AA4}}$ consists of the following clauses:

$$\begin{aligned}
a(X) &\leftarrow b(X) \wedge \neg ab_{ba}(X). && \text{(licenses)}\\
ab_{ba}(X) &\leftarrow \bot. && \text{(licenses)}\\
b(o_1) &\leftarrow \top. && \text{(import)}\\
c(X) &\leftarrow b(X) \wedge \neg ab_{bc}(X). && \text{(licenses)}\\
ab_{bc}(X) &\leftarrow \bot. && \text{(licenses)}\\
b(o_2) &\leftarrow \top. && \text{(import)}
\end{aligned}$$

The least model of the weak completion of $\mathcal{P}_{\mathsf{AA4}}$ is

$$\langle\{b(o_1),\ b(o_2),\ a(o_1),\ a(o_2), c(o_1),\ c(o_2)\},\\ \{ab_{ba}(o_1),\ ab_{ba}(o_2),\ ab_{bc}(o_1),\ ab_{bc}(o_2)\}\rangle.$$

This model entails both '*all a are c*' (Aac) and '*all c are a*' (Aca): There exists an object, o_1, such that $\mathcal{P}_{\mathsf{AA4}} \models_{wcs} a(o_1)$ and for all objects, o_1 and o_2, we find that if $\mathcal{P}_{\mathsf{AA4}} \models_{wcs} a(o_1)$ then $\mathcal{P} \models_{wcs} c(o_1)$ and if $\mathcal{P}_{\mathsf{AA4}} \models_{wcs} a(o_2)$ then $\mathcal{P} \models_{wcs} c(o_2)$.

	Participants	FOL	PSYCOP	Verbal Models	Mental Models	WCS
OA4	Oca	Oca	Oca, Ica, Iac	Oca, NVC	Oca, Oac, NVC	Oca
EA2	Eac, Eca	Eac, Eca Oac, Oca	Eac, Eca Oac, Oca	Eca	Eac, Eca	NVC
AA4	Aac, NVC	Iac, Ica	Iac, Ica	NVC, Aca	Aca, Aac, Iac, Ica	Aac, Aca
Overall results	100%		77%	84%	83%	85%

Table 7.4.: The conclusions drawn by a significant percentage of participants are highlighted in gray and compared to the predictions of the theories FOL, PSYCOP, Verbal, and Mental Models as well as WCS for the syllogisms OA4, EA2, and AA4.

Similarly, $\mathcal{P}_{\mathsf{AA4}} \models \mathsf{A}ca$ holds: There exists an object, o_1, such that $\mathcal{P}_{\mathsf{AA4}} \models_{wcs} c(o_1)$ and for all objects, o_1 and o_2, we find that if $\mathcal{P}_{\mathsf{AA4}} \models_{wcs} c(o_1)$ then $\mathcal{P} \models_{wcs} a(o_1)$ and if $\mathcal{P}_{\mathsf{AA4}} \models_{wcs} c(o_2)$ then $\mathcal{P} \models_{wcs} a(o_2)$. This prediction matches partially with the participants' answers who concluded Aac and NVC.

7.5. Conclusions

We discussed and formalized three examples under the Weak Completion Semantics. The results are summarized and compared to FOL, PSYCOP, the Verbal, and the Mental Model Theory in Table 7.4. The selected examples are typical in the sense that for OA4, the conclusions drawn by the participants and the Weak Completion Semantics are identical, for AA4, the conclusions drawn by the participants and the Weak Completion Semantics overlap, and for EA2, the conclusions drawn by the participants and the Weak Completion Semantics are disjoint. Overall, the Weak Completion Semantics differs from the other cognitive theories. The result with respect to the 64 syllogisms under the Weak Completion Semantics shows that we achieve the best results with a prediction of 85%. An overview can be found in Appendix D on page 205. In case, we evaluate the results with respect to the $\models_{\mathsf{FOL}}$ entailment, we can only predict 72% of the participants' responses. Compared to the other cognitive theories, we achieve together with the Verbal Models Theory (84%) the best performance, closely followed by the Mental Model Theory (83%). It seems natural to compare these theories in more detail and see where their similarities are and where they differ.

It might be interesting to investigate whether the combinations of the moods influence how the participants perceive the syllogisms. For instance, it seems that participants give different answers when they consider syllogistic premises of the same mood, especially in the cases for AA and EE. Yet, this is only an assumption and needs to be further investigated. The approach we propose in this chapter has one major drawback: We can never predict more than 89% of the participants responses in case participants have answered NVC together with some other conclusion. This is the case in 37 of in total 64 syllogism. For each of the 37 syllogisms, the Weak Completion Semantics can maximally predict 89% of the participants' responses, as NVC only follows in case no other conclusion follows, i.e. NVC is disjoint with any other conclusion. The current predictions under the Weak Completion Semantics with 85% are quite high compared to the maximal of 93.6%[8] we can reach in total, and it seems quite difficult to get better results within the current approach. Possibly, with help of abductive reasoning we could improve the results: Whenever the Weak Completion Semantics concludes NVC, we might find some explanations for the premises, which might yield some new relation with respect to a and c. How is it possible that PSYCOP, Mental Model Theory and Verbal Model theory derive NVC together with other conclusions? According to the results reported by Khemlani and Johnson-Laird in PSYCOP, the conclusion NVC is also disjoint with any other conclusions. Only the Mental Model Theory and the Verbal Model Theory can conclude NVC with some other conclusion: In the case of the Mental Model Theory, premises can have more than one mental model. Therefore, one model of some premises possibly confirms NVC whether some other model of the same premises yields some conclusion about the relation of a and c. Yet, in the case of the Verbal Model Theory, a program is implemented with three different versions which, given the premises, differ in the construction of the corresponding mental model. The predictions under the Verbal Model Theory are then the union of these three versions.

Summing up, what in this chapter is standing out: For the first time we are actually able to evaluate the performance of the Weak Completion Semantics and compare our predictions to other approaches. We predicted 64 syllogisms with only one logic programming representation for each of the four moods and show that the Weak Completion Semantics is indeed competitive compared to the other cognitive theories.

[8] $\frac{(89\% \times 37) + (100\% \times (64-37))}{64} = 93.6\%$

8. Belief-Bias Effect

As we have discussed in the previous chapter, psychological experiments on syllogistic reasoning have shown that participants did not always deduce the classical logically valid conclusions. We have developed five principles for the representation of syllogisms and showed that according to their encoding in logic programs, we can predict 85% of the participants' responses under the Weak Completion Semantics, which is, compared to other cognitive theories, quite competitive.

So far we haven't given any attention to the contextual setting of a syllogism and have only considered the 64 syllogisms in an abstract context. When we discussed the Wason Selection Task in in Chapter 5.2, we have seen that humans conclusions are different when a task is carried out in an abstract context or in a social context. Here, we are interested on whether a syllogistic reasoning task in a social context influences human reasoning. For this purpose, we will consider a syllogistic reasoning task in a social context, which has been proposed by Evans, Barston, and Pollard [1983]. They carried out a psychological experiment to investigate the belief-bias effect with respect to syllogisms. This task differs from the syllogisms in the previous chapter in the sense that now belief (or background knowledge) plays and important role.[1]

8.1. Introduction

Evans, Barston, and Pollard [1983] carried out a psychological study about deductive reasoning, which demonstrated possibly conflicting processes in human reasoning. Participants were presented different syllogisms for which they had to decide whether they accepted these syllogisms as valid. For instance, consider S_{vit}:

Premise 1	*No nutritional things are inexpensive.*
Premise 2	*Some vitamin tablets are inexpensive.*
Conclusion	*Therefore, some vitamin tablets are not nutritional.*

The Conclusion necessarily follows from the premises under classical logic. However, about half of the participants said that the syllogism was not valid. They were explicitly asked to logically validate or invalidate various syllogisms, but didn't seem to have the

[1] The original idea of the chapter has been published in [Dietz, 2015] and an improved version has been published in [Dietz, 2017].

	Type	Example	%
S_{dog}	valid and believable	*No police dogs are vicious.* *Some highly trained dogs are vicious.* *Therefore, some highly trained dogs are not police dogs.*	92
S_{vit}	valid and unbelievable	*No nutritional things are inexpensive.* *Some vitamin tablets are inexpensive.* *Therefore, some vitamin tablets are not nutritional.*	46
S_{rich}	invalid and unbelievable	*No millionaires are hard workers.* *Some rich people are hard workers.* *Therefore, some millionaires are not rich people.*	8
S_{cig}	invalid and believable	*No addictive things are inexpensive.* *Some cigarettes are inexpensive.* *Therefore, some addictive things are not cigarettes.*	92

Table 8.1.: Four examples of four types of syllogisms. The percentages of the participants that accepted the type of the syllogism as being valid are shown in the last column.

intellectual capability to do so. Even worse, they were not even aware about their inabilities. Participants reflectively read the instructions and understood well that they were required to reason logically from the premises to the conclusion. Nevertheless, the results show that their intuitions were stronger and delivered a tendency to say '*yes*' or '*no*' depending on whether the syllogism was believable [Evans, 2012].

Table 8.1 shows four examples of syllogisms, each one being of a different type, that have been evaluated by Evans, Barston, and Pollard [1983]. Note that the two premises of all four types are of the same form, namely *No A are B. Some C are B*, which corresponds to EI2. The first two types differ from the last two types with respect to the conclusions: In the first two types the conclusion of the examples, *Some highly trained dogs are not police dogs* and *Some vitamin tablets are not nutritional* correspond to the quantified statement *Some C are not A* (Oca), whereas in the last two types the conclusion of the examples, *Some millionaires are not rich people* and *Some addictive things are not cigarettes*, correspond to the quantified statement *Some A are not C* (Oac).[2]

If participants judged that 'the conclusion necessarily follows from the statements in the passage, [you]' they 'should answer *yes*, otherwise *no*.' The column on the right side shows the percentage of the participants that validated the syllogism. Note that the participants could only answer '*yes*' or '*no*', and did not have the chance to answer '*I don't know*' or similar. A detailed description of the experimental method of a sample experiment on the belief bias can be found in [Evans, 2012].

[2]See Chapter 7 for the meaning of EI2 and Oac. Under FOL and the Weak Completion Semantics, EI2 entails Oca. The majority of participants in [Khemlani and Johnson-Laird, 2012], concluded Oca and NVC.

Let us show that the first two syllogisms are indeed valid under classical logic and that the last two are not. Consider S_{dog} and S_{vit}, both of which have the same form, and assume that the first two premises hold. The formulas $p(X)$, $v(X)$ and $h(X)$ can either mean that X is a police dog, vicious and highly trained or that X is a nutritional thing, inexpensive and a vitamin tablet, respectively. The first two premises of S_{dog} and S_{vit} can be formulated as the following logical formula:

$$((\forall X)(p(X) \rightarrow \neg v(X)) \wedge (\exists Y)(h(Y) \wedge v(Y))),$$

where the formula on the left hand side of the outer conjunction represents the first premise, in the example of S_{dog}: *No police dogs are vicious.* The right hand side of the outer conjunction represents the second premise, in the example of S_{dog}: *Some highly trained dogs are vicious.* The Conclusion, in the example of S_{dog}: *Some highly trained dogs are not police dogs.* can be formulated as follows:

$$(\exists Z)(h(Z) \wedge \neg p(Z)).$$

We prove that the Conclusion follows from the premises, by showing that

$$(((\forall X)(p(X) \rightarrow \neg v(X)) \wedge (\exists Y)(h(Y) \wedge v(Y))) \rightarrow ((\exists Z)(h(Z) \wedge \neg p(Z))))$$

is valid. Figure E.1 in Appendix E shows the proof within the calculus of natural deduction by following the notation and the definitions in [Hölldobler, 2009].

On the other hand, S_{rich} and S_{cig} are not valid, which can best be shown by the Venn diagram in Figure 8.1. The first premise *No addictive things are inexpensive* leads us to conclude that the intersection between addictive things and inexpensive is empty, i.e. there is nothing which is both addictive and inexpensive (denoted by the hatched lines). From the second premise we know that there exists something, which is inexpensive and a cigarette, for instance some constant o. This is represented by the o drawn in the intersection of inexpensive and cigarettes. We try to verify the Conclusion, *Some addictive things are not cigarettes*, by examining whether there is at least one o in the part of the addictive things that is not in the part of the cigarettes. This is actually not the case, as all addictive things could simply be cigarettes, denoted by the dots in the intersecting area of addictive things and cigarettes. Consequently, the syllogism is not valid. Analogously, we try to verify S_{rich}.

8.2. Theories about the Belief-Bias Effect

Evans, Barston and Pollard asserted that the participants were influenced by their own beliefs, their so-called belief bias. We can distinguish between the negative and the positive belief bias [Evans, Handley, and Harper, 2001]. The negative belief bias, i.e. when a support for the unbelievable conclusion is suppressed, happens for about half of the participants such as for the example of S_{vit}. A positive belief bias, i.e. when

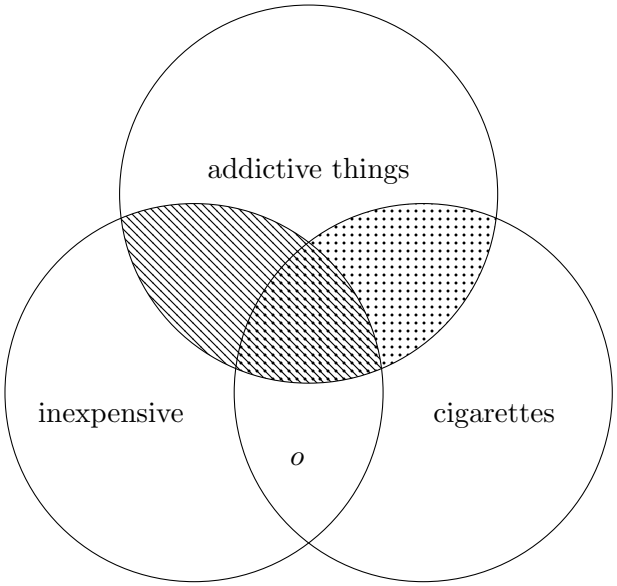


Figure 8.1.: Venn diagram showing that syllogism S_{cig} and S_{rich} are invalid.

the acceptance for the believable conclusion is raised, happens for the majority of the participants such as for the example of S_{cig}. As pointed out in [Garnham and Oakhill, 1994], Wilkins [1928] already observed that syllogisms which conflict with our beliefs are more difficult to solve.

Various theories have tried to explain why humans deviate from the classical logically valid answers. Some conclusions can be explained by converting the premises [Chapman and Chapman, 1959] or by assuming that the type of the premises create an atmosphere which influences the acceptance for the conclusion [Woodworth and Sells, 1935, Garnham and Oakhill, 1994]. The atmosphere hypothesis states the following:

1. Any negative premise (e.g. *no a are b* or *some a are not b*) creates a negative atmosphere, in which negative conclusions tend to be more easily accepted than other ones.

2. Any particular premise (e.g. *some a are not b* or *some a are not b*) creates a particular atmosphere, in which conclusions of the particular form tend to be more easily accepted than other ones.

The syllogisms in Figure 8.1 contain premises imposing a negative and particular atmosphere. This has been done on purpose in order to avoid any possible bias based on the atmosphere.

Johnson-Laird and Byrne [1991] proposed the mental model theory [Johnson-Laird, 1983], which additionally supposes the search for counterexamples when validating the

conclusion. Later, Stenning and van Lambalgen [2008] explain why certain aspects influence the interpretations made by humans when evaluating syllogisms and discuss this in the context of mental models. Evans, Barston, and Pollard [1983] and Evans [1989] proposed a theory, which in the literature is sometimes referred to as the selective scrutiny model [Garnham and Oakhill, 1994, Adler and Rips, 2008]. First, humans heuristically accept any syllogism having a believable conclusion, and only proceed with a logical evaluation if the conclusion contradicts their belief.

Adler and Rips [2008] claim that this behavior is rational in the sense of efficient belief maintenance. It results in a normal adaptive process for which we only make an effort towards a logical evaluation when the conclusion is unbelievable. It would take a lot of effort if we would constantly verify conclusions even though there is no reason to question them. As people generally intend to keep their beliefs as consistent as possible, they invest more effort in examining statements that contradict their beliefs, than the ones that comply with them. Yet, this theory cannot fully explain all classical logical errors in the reasoning process. Yet another approach, the selective processing model, accounts only for a single preferred model [Evans, 2000]. If the conclusion is neutral or believable, humans try to construct a model that supports it. Otherwise, they attempt to construct a model that rejects it.

According to Garnham and Oakhill [1994] the belief-bias effect can take place at several stages: First, beliefs can influence our understanding of the premises. Second, in case a statement contradicts our belief, we might search for alternative models and check whether the conclusion is plausible. This seems to comply with Stenning and van Lambalgen's proposal to model human reasoning by a two step procedure, which we have discussed in the introduction of this thesis and again in Chapter 5: The first step, the representational part, determines how our beliefs influence the understanding of the premises. The second step, the procedural part, determines whether we search for alternative models based on the plausibility of the conclusion. Here, we will follow up on this distinction when modeling the belief-bias effect and show how we can model this syllogistic task under the Weak Completion Semantics together with the non-classical logical conclusions made by the participants.

In the following section, we model all four syllogisms of Table 8.1. They are typical in the sense that, in S_{dog} there is no belief bias, in S_{vit} we identify a belief bias in the representational part, in S_{cig} we identify is a belief bias in the procedural part and in S_{rich} we identify a belief bias in the representational and in the reasoning part.

8.3. Two Additional Principles

If the belief bias occurs in the representational part, we can model it with help of the principle licenses, i.e. with help of abnormalities. For the belief bias that occurs

during the reasoning process, we introduce a new principle, *Search for Alternative Models* (searchAlt), which we model with help of skeptical abduction.

8.3.1. Background Knowledge

Recall principle (licenses) from Section 7.2.1: The way of representing quantified statements as '$q(X)$ *if* $p(X)$ *and* $\neg ab_{pq}(X)$', where $ab_{pq}(X)$ is an abnormality predicate, allows us to express additional background knowledge, which possibly influences our beliefs about this statement. For instance, this can be done by extending the program with the additional statement $ab_{pq}(X) \leftarrow r(X)$, where $r(X)$ stands for the additional belief. If the belief-bias effect occurs on the representational part, we will encode the belief bias with help of abnormality predicates.

8.3.2. Search for Alternative Models

Consider again S_{rich} and S_{add}: The premises are about things which contradict the conclusion. We assume that in case there seems no conclusion possible, humans might try to search for alternative models by perceiving the first part of the conclusion as an observation, that needs to be explained. We assume that the belief-bias effect occurs during the reasoning process. We call this principle *Search for Alternative Models* (searchAlt). We will model this principle with the help of abduction, formally introduced in Chapter 2.5. Recall, that given a knowledge base and an observation, the goal of abduction is to compute a minimal explanation that entails the observation.

8.4. Representation as Logic Programs

According to the observations made in Section 8.2, we model the belief-bias effect when (1) the belief can influence the representation, i.e. how the given information is understood, and when (2) the belief can influence the reasoning, i.e. how new information is gained, if nothing can be derived. In the following, we model (1) with the help of abnormalities, motivated by principle (licenses). (2) is modeled by means of skeptical abduction, motivated by the principle (searchAlt).

8.4.1. No Belief-Bias Effect

According to Section 7.3.2 of the previous chapter, Premise 1 in S_{dog}, *No police dogs are vicious*, is encoded by the following five clauses, where the terms in brackets refer to

the respective principles introduced in Section 7.2:[3]

$$\begin{array}{rcll} \mathit{police\_dog}'(X) & \leftarrow & \mathit{vicious}(X) \wedge \neg \mathit{ab}_{\mathit{police\_dog}'}(X). & (\textsf{transformation \& licenses}) \\ \mathit{ab}_{\mathit{police\_dog}'}(X) & \leftarrow & \bot. & (\textsf{licenses}) \\ \mathit{police\_dog}(X) & \leftarrow & \neg \mathit{police\_dog}'(X) \wedge \neg \mathit{ab}_{\mathit{police\_dog}}(X). & (\textsf{transformation \& licenses}) \\ \mathit{vicious}(o_1) & \leftarrow & \top. & (\textsf{import}) \\ \mathit{ab}_{\mathit{police\_dog}}(o_1) & \leftarrow & \bot. & (\textsf{licenses \& doubleNeg}) \end{array}$$

In addition, we have the following integrity constraint:

$$\mathrm{U} \leftarrow \mathit{police\_dog}(X) \wedge \mathit{police\_dog}'(X). \qquad (\textsf{transformation})$$

$\mathit{police\_dog}(X)$ and $\mathit{police\_dog}'(X)$ denote that X is a police dog and not a police dog, respectively. According to Section 7.3.3, PREMISE 2 in S_{dog}, *Some highly trained dogs are vicious*, is represented by the following four clauses:[4]

$$\begin{array}{rcll} \mathit{vicious}(X) & \leftarrow & \mathit{highly\_trained}(X) \wedge \neg \mathit{ab}_{\mathit{vicious}}(X). & (\textsf{licenses}) \\ \mathit{ab}_{\mathit{vicious}}(o_2) & \leftarrow & \bot. & (\textsf{unknownGen \& licenses}) \\ \mathit{highly\_trained}(o_2) & \leftarrow & \top. & (\textsf{import}) \\ \mathit{highly\_trained}(o_3) & \leftarrow & \top. & (\textsf{unknownGen}) \end{array}$$

$\mathcal{P}_{dog}$ represents the first two premises of S_{dog} and consists of

$$\begin{array}{rcl} \mathit{police\_dog}'(X) & \leftarrow & \mathit{vicious}(X) \wedge \neg \mathit{ab}_{\mathit{police\_dog}'}(X). \\ \mathit{ab}_{\mathit{police\_dog}'}(X) & \leftarrow & \bot. \\ \mathit{police\_dog}(X) & \leftarrow & \neg \mathit{police\_dog}'(X) \wedge \neg \mathit{ab}_{\mathit{police\_dog}}(X). \\ \mathit{vicious}(o_1) & \leftarrow & \top. \\ \mathit{ab}_{\mathit{police\_dog}}(o_1) & \leftarrow & \bot. \\ \mathit{vicious}(X) & \leftarrow & \mathit{highly\_trained}(X) \wedge \neg \mathit{ab}_{\mathit{vicious}}(X). \\ \mathit{ab}_{\mathit{vicious}}(o_2) & \leftarrow & \bot. \\ \mathit{highly\_trained}(o_2) & \leftarrow & \top. \\ \mathit{highly\_trained}(o_3) & \leftarrow & \top. \end{array}$$

The weak completion of $\mathcal{P}_{dog}$ is shown in Box 1. Its least model, $\langle I^{\top}, I^{\bot} \rangle$, is as follows:

$$\begin{array}{rcl} I^{\top} & = & \{\mathit{highly\_trained}(o_2), \mathit{highly\_trained}(o_3), \mathit{police\_dog}'(o_1), \\ & & \ \mathit{police\_dog}'(o_2), \mathit{vicious}(o_1), \mathit{vicious}(o_2)\}, \\ I^{\bot} & = & \{\mathit{police\_dog}(o_2), \mathit{police\_dog}(o_1), \\ & & \ \mathit{ab}_{\mathit{police\_dog}'}(o_1), \mathit{ab}_{\mathit{police\_dog}'}(o_2), \mathit{ab}_{\mathit{police\_dog}'}(o_3), \mathit{ab}_{\mathit{police\_dog}}(o_1), \mathit{ab}_{\mathit{vicious}}(o_2)\}, \end{array}$$

[3] Note that o, y, y', z, ab_{zny} and ab_{nyy} in $\mathcal{P}_{Eyz}$ in Section 7.3.2 are replaced here by o_1, $\mathit{police\_dog}$, $\mathit{police\_dog}'$, $\mathit{vicious}$, $\mathit{ab}_{\mathit{police\_dog}'}$ and $\mathit{ab}_{\mathit{police\_dog}}$, respectively.

[4] Note that o_1, o_2, y, z and ab_{yz} in $\mathcal{P}_{Iyz}$ in Section 7.3.3 are replaced here by o_2, o_3, $\mathit{highly\_trained}$, $\mathit{vicious}$ and $\mathit{ab}_{\mathit{vicious}}$, respectively.

Box 1. $\mathsf{wc}\,\mathcal{P}_{dog}$ consists of the following clauses[a]:

$$
\begin{array}{rcl}
police\_dog'(o_1) & \leftrightarrow & vicious(o_1) \wedge \neg ab_{police\_dog'}(o_1). \\
police\_dog'(o_2) & \leftrightarrow & vicious(o_2) \wedge \neg ab_{police\_dog'}(o_2). \\
police\_dog'(o_3) & \leftrightarrow & vicious(o_3) \wedge \neg ab_{police\_dog'}(o_3). \\
police\_dog(o_1) & \leftrightarrow & \neg police\_dog'(o_1) \wedge \neg ab_{police\_dog}(o_1). \\
police\_dog(o_2) & \leftrightarrow & \neg police\_dog'(o_2) \wedge \neg ab_{police\_dog}(o_2). \\
police\_dog(o_3) & \leftrightarrow & \neg police\_dog'(o_3) \wedge \neg ab_{police\_dog}(o_3). \\
vicious(o_1) & \leftrightarrow & highly\_trained(o_1) \wedge \neg ab_{vicious}(o_1). \\
vicious(o_2) & \leftrightarrow & highly\_trained(o_2) \wedge \neg ab_{vicious}(o_2). \\
vicious(o_3) & \leftrightarrow & highly\_trained(o_3) \wedge \neg ab_{vicious}(o_3). \\
vicious(o_1) & \leftrightarrow & \top. \\
ab_{police\_dog}(o_1) & \leftrightarrow & \bot. \\
highly\_trained(o_2) & \leftrightarrow & \top. \\
highly\_trained(o_3) & \leftrightarrow & \top. \\
ab_{vicious}(o_2) & \leftrightarrow & \bot. \\
ab_{police\_dog'}(o_1) & \leftrightarrow & \bot. \\
ab_{police\_dog'}(o_2) & \leftrightarrow & \bot. \\
ab_{police\_dog'}(o_3) & \leftrightarrow & \bot.
\end{array}
$$

[a]Note that here and in the following, the only purpose for the clauses highlighted in white is a better readability.

Indeed, this model entails the Conclusion of S_{dog} that *Some highly trained dogs are not police dogs*: There exists an object, namely o_2, such that

$$\mathcal{P}_{dog} \models_{wcs} highly\_trained(o_2) \wedge \neg police\_dog(o_2)$$

and there exists another object, namely o_3, such that

$$\mathcal{P}_{dog} \models_{wcs} highly\_trained(o_3) \quad \text{and} \quad \mathcal{P}_{dog} \not\models_{wcs} \neg police\_dog(o_3).$$

According to Evans, Barston, and Pollard [1983], this type of syllogism is logically valid and psychologically believable. No conflict arises either at the psychological or at the logical level. The majority validated the syllogism, which complies with what is entailed by $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{dog}$.

8.4.2. Belief-Bias Effect in Representation

Premise 1 and Premise 2 in S_{vit}, *Some vitamin tablets are inexpensive*, can be modeled analogously to Premise 1 and Premise 2 in S_{dog}. $\mathcal{P}_{vit}$ represents the two premises

of S_{vit} and consists of[5]

$$
\begin{array}{rcll}
nutritional'(X) & \leftarrow & inex(X) \wedge \neg ab_{nutritional'}(X). & \text{(transformation \& licenses)} \\
ab_{nutritional'}(X) & \leftarrow & \bot. & \text{(licenses)} \\
nutritional(X) & \leftarrow & \neg nutritional'(X) \wedge \neg ab_{nutritional}(X). & \text{(transformation \& licenses)} \\
inex(o_1) & \leftarrow & \top. & \text{(import)} \\
ab_{nutritional}(o_1) & \leftarrow & \bot. & \text{(licenses \& doubleNeg)} \\
inex(X) & \leftarrow & vitamin(X), \neg ab_{inex}(X). & \text{(licenses)} \\
ab_{inex}(o_2) & \leftarrow & \bot. & \text{(unknownGen \& licenses)} \\
vitamin(o_2) & \leftarrow & \top. & \text{(import)} \\
vitamin(o_3) & \leftarrow & \top. & \text{(unknownGen)}
\end{array}
$$

In addition, we have the following integrity constraint:

$$\mathrm{U} \leftarrow nutritional'(X) \wedge nutritional(X). \qquad \text{(transformation)}$$

The weak completion of $\mathcal{P}_{vit}$ is shown in Box 2. The corresponding least model, $\langle I^{\top}, I^{\bot} \rangle$, is as follows:

$$
\begin{array}{rcl}
I^{\top} & = & \{vitamin(o_2), vitamin(o_3), inex(o_1), inex(o_2), \\
 & & \ nutritional'(o_1), nutritional'(o_2)\} \\
I^{\bot} & = & \{nutritional(o_1), nutritional(o_2), ab_{inex}(o_2), ab_{nutritional}(o_1), \\
 & & \ ab_{nutritional'}(o_1), ab_{nutritional'}(o_2), ab_{nutritional'}(o_3)\},
\end{array}
$$

Indeed, this model entails the Conclusion of S_{vit} that *Some vitamin tablets are not nutritional*: There exists an object, namely o_2, such that

$$\mathcal{P}_{vit} \models_{wcs} vitamin(o_2) \wedge \neg nutritional(o_2)$$

and there exists another object, namely o_3, such that

$$\mathcal{P}_{vit} \models_{wcs} vitamin(o_3) \quad \text{and} \quad \mathcal{P}_{vit} \not\models_{wcs} \neg nutritional(o_3).$$

46% of the participants validated the syllogism, which complies with what is entailed by $\mathsf{lm\,wc}\ \mathcal{P}_{vit}$. As Table 8.1 shows, the psychological results of the second syllogism, S_{vit}, indicate that there seemed to be two groups of participants where each group had a different understanding of the premises. The group that validated the syllogism was not influenced by some bias with respect to vitamin tablets. Their understanding of the syllogism is reflected by $\mathcal{P}_{vit}$ and their conclusion complies with what is entailed by $\mathsf{lm\,wc}\,\mathcal{P}_{vit}$. The participants who chose to invalidate the syllogism belong to the other group that has apparently been influenced by the belief. The belief bias occurred in the the representational part of the syllogism. This aspect will be modeled as discussed in Section 8.3.1 with help of abnormality predicates.

[5] $nutritional(X)$, $nutritional'(X)$ denote that X is nutritional, not nutritional, respectively.

Box 2. $\mathsf{wc}\,\mathcal{P}_{vit}$ consists of the following clauses:

$$
\begin{array}{rcl}
nutritional'(o_1) & \leftrightarrow & inex(o_1) \wedge \neg ab_{nutritional'}(o_1). \\
nutritional'(o_2) & \leftrightarrow & inex(o_2) \wedge \neg ab_{nutritional'}(o_2). \\
nutritional'(o_3) & \leftrightarrow & inex(o_3) \wedge \neg ab_{nutritional'}(o_3). \\
nutritional(o_1) & \leftrightarrow & \neg nutritional'(o_1) \wedge \neg ab_{nutritional}(o_1). \\
nutritional(o_2) & \leftrightarrow & \neg nutritional'(o_2) \wedge \neg ab_{nutritional}(o_2). \\
nutritional(o_3) & \leftrightarrow & \neg nutritional'(o_3) \wedge \neg ab_{nutritional}(o_3). \\
inex(o_1) & \leftrightarrow & vitamin(o_1) \wedge \neg ab_{inex}(o_1). \\
inex(o_2) & \leftrightarrow & vitamin(o_2) \wedge \neg ab_{inex}(o_2). \\
inex(o_3) & \leftrightarrow & vitamin(o_3) \wedge \neg ab_{inex}(o_3). \\
inex(o_1) & \leftrightarrow & \top. \\
ab_{nutritional}(o_1) & \leftrightarrow & \bot. \\
vitamin(o_2) & \leftrightarrow & \top. \\
vitamin(o_3) & \leftrightarrow & \top. \\
ab_{inex}(o_2) & \leftrightarrow & \bot. \\
ab_{nutritional'}(o_1) & \leftrightarrow & \bot. \\
ab_{nutritional'}(o_2) & \leftrightarrow & \bot. \\
ab_{nutritional'}(o_3) & \leftrightarrow & \bot.
\end{array}
$$

Regarding both premises, someone might observe that it is commonly known that

The purpose of vitamin tablets is to aid nutrition.

This belief in the context of Premise 1 leads to

If something is a vitamin tablet, then it is abnormal.
(regarding Premise 1 *of* S_{vit}*)*

We extend $\mathcal{P}_{vit}$ according to this new information, resulting in

$$\mathcal{P}_{vit}^{\mathsf{bias}} \;=\; \mathcal{P}_{vit} \;\cup\; \{ab_{nutritional'}(X) \leftarrow vitamin(X)\},$$

The interpretation of S_{vit} together with the belief-bias effect is represented by $\mathcal{P}_{vit}^{\mathsf{bias}}$. Observe that $ab_{nutritional'}(X) \leftarrow vitamin(X)$ overrides $ab_{nutritional'}(X) \leftarrow \bot(X)$ under the weak completion of $\mathcal{P}_{vit}^{\mathsf{bias}}$. The weak completion of $\mathcal{P}_{vit}^{\mathsf{bias}}$ differs with respect to the last three clauses in $\mathsf{wc}\,\mathcal{P}_{vit}$. The last three clauses in $\mathsf{wc}\,\mathcal{P}_{vit}^{\mathsf{bias}}$ are as follows:

$$
\begin{array}{rcl}
ab_{nutritional'}(o_1) & \leftrightarrow & \bot \vee vitamin(o_1). \\
ab_{nutritional'}(o_2) & \leftrightarrow & \bot \vee vitamin(o_2). \\
ab_{nutritional'}(o_3) & \leftrightarrow & \bot \vee vitamin(o_3).
\end{array}
$$

Its least model, $\langle I^\top, I^\bot \rangle$ is

$$I^\top = \{inex(o_1), inex(o_2), vitamin(o_2), vitamin(o_3), ab_{nutritional'}(o_2), ab_{nutritional'}(o_3)\},$$
$$I^\bot = \{nutritional'(o_2), nutritional'(o_3), ab_{nutritional}(o_1), ab_{inex}(o_2)\}.$$

In this case, the Conclusion of S_{vit}, that *Some vitamin tablets are not nutritional*, is not entailed. Actually, nothing is stated about the relation between vitamin tablets and them (not) being nutritional. Yet, we can derive from the model that some vitamin tablets exist, which are inexpensive, therefore principle (searchAlt) does not apply and we are done. According to Evans, Barston, and Pollard [1983], type of syllogism is logically valid but psychologically unbelievable. There arises a conflict at the psychological level, because we generally assume that the purpose of vitamin tablets is to aid nutrition. The participants who have been influenced by this belief did not validate the syllogism, which complies to the result above, as the Conclusion is not entailed by $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{vit}^{\mathsf{bias}}$ either.

8.4.3. Belief-Bias Effect in Reasoning

$\mathcal{P}_{rich}$ represents Premise 1 and Premise 2 of S_{rich} and consists of[6]

$$\begin{array}{rcll}
mil'(X) & \leftarrow & hard_worker(X) \wedge \neg ab_{mil'}(X). & \text{(transformation \& licenses)} \\
ab_{mil'}(X) & \leftarrow & \bot. & \text{(licenses)} \\
mil(X) & \leftarrow & \neg mil'(X) \wedge ab_{mil}(X). & \text{(transformation \& licenses)} \\
hard_worker(o_1) & \leftarrow & \top. & \text{(import)} \\
ab_{mil}(o_1) & \leftarrow & \bot. & \text{(licenses \& doubleNeg)} \\
hard_worker(X) & \leftarrow & rich(X) \wedge \neg ab_{hard_worker}(X). & \text{(licenses)} \\
ab_{hard_worker}(o_2) & \leftarrow & \bot. & \text{(unknownGen \& licenses)} \\
rich(o_2) & \leftarrow & \top. & \text{(import)} \\
rich(o_3) & \leftarrow & \top. & \text{(unknownGen)}
\end{array}$$

In addition, we have the following integrity constraint:

$$\mathrm{U} \leftarrow mil(X) \wedge mil'(X). \qquad \text{(transformation)}$$

The weak completion of $\mathcal{P}_{mil}$ is shown in Box 3. Its least model, $\langle I^\top, I^\bot \rangle$, is as follows:

$$I^\top = \{hard\_worker(o_1), hard\_worker(o_2), mil'(o_1), mil'(o_2), rich(o_2), rich(o_3)\},$$
$$I^\bot = \{mil(o_1), mil(o_2), ab_{hard\_worker}(o_2), ab_{mil}(o_1), ab_{mil'}(o_1), ab_{mil'}(o_2), ab_{mil'}(o_3)\}$$

This model does not confirm the Conclusion of S_{rich} that *some millionaires are not rich people*. The Conclusion in S_{rich} states something which contradicts Premise 2 and cannot be about any of the previously introduced constant o_1, o_2 or o_3. As nothing can be derived about the relation between *mil* and *hard_worker* nor between *mil* and

[6] $mil(X)$ and $mil'(X)$ denote that X is a millionaire and not a millionaire, respectively.

Box 3. $\mathsf{wc}\,\mathcal{P}_{mil}$ consists of the following clauses:

$$
\begin{array}{rcl}
mil'(o_1) & \leftrightarrow & hard\_worker(o_1) \wedge \neg ab_{mil'}(o_1). \\
mil'(o_2) & \leftrightarrow & hard\_worker(o_2) \wedge \neg ab_{mil'}(o_2). \\
mil'(o_3) & \leftrightarrow & hard\_worker(o_3) \wedge \neg ab_{mil'}(o_3). \\
mil(o_1) & \leftrightarrow & \neg mil'(o_1) \wedge \neg ab_{mil}(o_1). \\
mil(o_2) & \leftrightarrow & \neg mil'(o_2) \wedge \neg ab_{mil}(o_2). \\
mil(o_3) & \leftrightarrow & \neg mil'(o_3) \wedge \neg ab_{mil}(o_3). \\
hard\_worker(o_1) & \leftrightarrow & rich(o_1) \wedge \neg ab_{hard\_worker}(o_1). \\
hard\_worker(o_2) & \leftrightarrow & rich(o_2) \wedge \neg ab_{hard\_worker}(o_2). \\
hard\_worker(o_3) & \leftrightarrow & rich(o_3) \wedge \neg ab_{hard\_worker}(o_3). \\
hard\_worker(o_1) & \leftrightarrow & \top. \\
ab_{mil}(o_1) & \leftrightarrow & \bot. \\
rich(o_2) & \leftrightarrow & \top. \\
rich(o_3) & \leftrightarrow & \top. \\
ab_{hard\_worker}(o_2) & \leftrightarrow & \bot. \\
ab_{mil'}(o_1) & \leftrightarrow & \bot. \\
ab_{mil'}(o_2) & \leftrightarrow & \bot. \\
ab_{mil'}(o_3) & \leftrightarrow & \bot.
\end{array}
$$

rich, principle (searchAlt) of Section 8.3.2 applies: According to our background knowledge, we know that 'normal' millionaires exist, i.e. millionaires for whom we do not assume anything abnormal with respect to them being millionaires. Further, we cannot be sure that all millionaires are normal, i.e. we know that millionaires exist for whom we don't know whether they are normal. This is as an observation about two newly introduced constants, let's say o_4, representing a normal millionaire,[7] and o_5, representing a millionaire for whom it is unknown whether he or she is normal:

$$\mathcal{O} \quad = \quad \{mil(o_4), \neg ab_{mil'}(o_4), \neg ab_{mil}(o_4), mil(o_5)\}.$$

If we want to find an explanation for $\mathcal{O}$ with respect to $\mathcal{P}_{mil}$, we can no longer assume that $\mathcal{C} = \mathsf{constants}(\mathcal{P}_{mil})$, as $\mathcal{A}_{\mathcal{P}_{mil}}$ does not contain any facts or assumptions about o_4 and o_5. We specify $\mathcal{C} = \{o_1, o_2, o_3, o_4, o_5\}$. Additionally to the previously listed clauses

[7] This implies that all abnormalities about *mil* or *mil'* are false with respect to o_4.

in $g\mathcal{P}_{mil}$, $\mathcal{P}_{mil}$ ground with respect to $\mathcal{C}$, $g\mathcal{P}^{\mathcal{C}}_{mil}$, consists of the following eight clauses:

$$\begin{aligned}
mil'(o_4) &\leftarrow hard_worker(o_4) \wedge \neg ab_{mil'}(o_4).\\
ab_{mil'}(o_4) &\leftarrow \bot.\\
mil(o_4) &\leftarrow \neg mil'(o_4) \wedge \neg ab_{mil}(o_4).\\
hard_worker(o_4) &\leftarrow rich(o_4) \wedge \neg ab_{hard_worker}(o_4).\\
mil'(o_5) &\leftarrow hard_worker(o_5) \wedge \neg ab_{mil'}(o_5).\\
ab_{mil'}(o_5) &\leftarrow \bot.\\
mil(o_5) &\leftarrow \neg mil'(o_5) \wedge \neg ab_{mil}(o_5).\\
hard_worker(o_5) &\leftarrow rich(o_5) \wedge \neg ab_{hard_worker}(o_5).
\end{aligned}$$

Given that $\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}_{mil}) = \langle I^{\top}, I^{\bot}\rangle$ as defined above, $\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}^{\mathcal{C}}_{mil})$ is as follows:

$$\langle I^{\top}, I^{\bot} \cup \{ab_{mil'}(o_4), ab_{mil'}(o_5)\}\rangle.$$

The set of abducibles, $\mathcal{A}_{\mathcal{P}^{\mathcal{C}}_{mil}}$, has now six facts and assumptions about o_4 and o_5:

$$\begin{array}{llllllllll}
rich(o_4) & \leftarrow & \top. & & ab_{mil}(o_4) & \leftarrow & \top. & & ab_{hard_worker}(o_4) & \leftarrow & \top.\\
rich(o_4) & \leftarrow & \bot. & & ab_{mil}(o_4) & \leftarrow & \bot. & & ab_{hard_worker}(o_4) & \leftarrow & \bot.\\
rich(o_5) & \leftarrow & \top. & & ab_{mil}(o_5) & \leftarrow & \top. & & ab_{hard_worker}(o_5) & \leftarrow & \top.\\
rich(o_5) & \leftarrow & \bot. & & ab_{mil}(o_5) & \leftarrow & \bot. & & ab_{hard_worker}(o_5) & \leftarrow & \bot.
\end{array}$$

We find six explanations for $\mathcal{O} = \{mil(o_4), \neg ab_{mil'}(o_4), \neg ab_{mil}(o_4), mil(o_5)\}$:

$$\begin{aligned}
\mathcal{E}_1 &= \{rich(o_4) \leftarrow \bot, ab_{mil}(o_4) \leftarrow \bot, ab_{mil}(o_5) \leftarrow \top\},\\
\mathcal{E}_2 &= \{rich(o_4) \leftarrow \bot, ab_{mil}(o_4) \leftarrow \bot, rich(o_5) \leftarrow \bot\},\\
\mathcal{E}_3 &= \{rich(o_4) \leftarrow \bot, ab_{mil}(o_4) \leftarrow \bot, ab_{hard_worker}(o_5) \leftarrow \top\},\\
\mathcal{E}_4 &= \{ab_{hard_worker}(o_4) \leftarrow \top, ab_{mil}(o_4) \leftarrow \bot, ab_{mil}(o_5) \leftarrow \top\},\\
\mathcal{E}_5 &= \{ab_{hard_worker}(o_4) \leftarrow \top, ab_{mil}(o_4) \leftarrow \bot, rich(o_5) \leftarrow \bot\},\\
\mathcal{E}_6 &= \{ab_{hard_worker}(o_4) \leftarrow \top, ab_{mil}(o_4) \leftarrow \bot, ab_{hard_worker}(o_5) \leftarrow \top\}.
\end{aligned}$$

The least models of the weak completion of $\mathcal{P}^{\mathcal{C}}_{mil}$ together with the six corresponding explanations, are shown in Box 4. The atoms highlighted in white in Box 4 are the ones which follow from all explanations, that means, these are the skeptically entailed atoms. The Conclusion of S_{rich}, *Some millionaires are not rich people*, does not follow skeptically from $\mathcal{P}^{\mathcal{C}}_{mil}$ and the observation $\mathcal{O}$. According to the definition for skeptical abduction in Section 2.5, one explanation for which the Conclusion of S_{rich}, *Some millionaires are not rich people*, does not follow is enough to show that the Conclusion does not follow skeptically from $\mathcal{P}^{\mathcal{C}}_{mil}$, $\mathcal{IC}$ and $\mathcal{O}$: Consider $\mathcal{E}_4$, where we cannot derive that *Some millionaires are not rich people* in order to conclude that the Conclusion does not follow skeptically from $\mathcal{P}^{\mathcal{C}}_{mil}$ and the observation $\mathcal{O}$. According to Evans, Barston, and Pollard [1983], this type of syllogism is neither logically valid nor believable. Almost no one validated S_{rich}, which complies to the result above, as the Conclusion is not skeptically entailed by $\mathcal{P}^{\mathcal{C}}_{mil}$, $\mathcal{IC}$ and $\mathcal{O}$.

Box 4. Given that $\mathsf{lm\,wc}\,(\mathcal{P}_{mil}) = \langle I^\top, I^\perp \rangle$, the least models of the weak completion of $\mathcal{P}^{\mathcal{C}}_{mil}$ together with the six corresponding explanations, are as follows:

$\mathsf{lm\,wc}\,(\mathcal{P}^{\mathcal{C}}_{mil} \cup \mathcal{E}_1)$
$= \langle I^\top \cup \{\ mil(o_4)\ ,\ mil(o_5)\ , ab_{mil}(o_5)\},$
$\quad I^\perp \cup \{\ ab_{mil}(o_4)\ ,\ ab_{mil'}(o_4)\ ,\ hard\_worker(o_4)\ ,\ mil'(o_4)\ , rich(o_4),\ ab_{mil'}(o_5)\ \}\rangle,$

$\mathsf{lm\,wc}\,(\mathcal{P}^{\mathcal{C}}_{mil} \cup \mathcal{E}_2)$
$= \langle I^\top \cup\ mil(o_4)\ ,\ mil(o_5)\ \},$
$\quad I^\top \cup \{\ ab_{mil}(o_4)\ ,\ ab_{mil'}(o_4)\ ,\ hard\_worker(o_4)\ ,\ mil'(o_4)\ , rich(o_4),\ ab_{mil'}(o_5)\ ,$
$\quad hard\_worker(o_5), mil'(o_5), rich(o_5)\}\rangle,$

$\mathsf{lm\,wc}\,(\mathcal{P}^{\mathcal{C}}_{mil} \cup \mathcal{E}_3)$
$= \langle I^\top \cup \{\ mil(o_4)\ ,\ mil(o_5)\ , ab_{hard\_worker}(o_5)\},$
$\quad I^\perp \cup \{\ ab_{mil}(o_4)\ ,\ ab_{mil'}(o_4)\ ,\ hard\_worker(o_4)\ ,\ mil'(o_4)\ , rich(o_4),\ ab_{mil'}(o_5)\ ,$
$\quad hard\_worker(o_5), mil'(o_5)\}\rangle,$

$\mathsf{lm\,wc}\,(\mathcal{P}^{\mathcal{C}}_{mil} \cup \mathcal{E}_4)$
$= \langle I^\top \cup \{ab_{hard\_worker}(o_4),\ mil(o_4)\ ,\ mil(o_5)\ , ab_{mil}(o_5)\},$
$\quad I^\perp \cup \{\ ab_{mil}(o_4)\ ,\ ab_{mil'}(o_4)\ ,\ hard\_worker(o_4)\ ,\ mil'(o_4)\ ,\ ab_{mil'}(o_5)\ \}\rangle,$

$\mathsf{lm\,wc}\,(\mathcal{P}^{\mathcal{C}}_{mil} \cup \mathcal{E}_5)$
$= \langle I^\top \cup \{ab_{hard\_worker}(o_4),\ mil(o_4)\ ,\ mil(o_5)\ \},$
$\quad I^\perp \cup \{\ ab_{mil}(o_4)\ ,\ ab_{mil'}(o_4)\ ,\ hard\_worker(o_4)\ ,\ mil'(o_4)\ ,\ ab_{mil'}(o_5)\ ,$
$\quad hard\_worker(o_5), mil'(o_5), rich(o_5)\}\rangle,$

$\mathsf{lm\,wc}\,(\mathcal{P}^{\mathcal{C}}_{mil} \cup \mathcal{E}_6)$
$= \langle I^\top \cup \{ab_{hard\_worker}(o_4),\ mil(o_4)\ ,\ mil(o_5)\ , ab_{hard\_worker}(o_5)\},$
$\quad I^\perp \cup \{\ ab_{mil}(o_4)\ ,\ ab_{mil'}(o_4)\ ,\ hard\_worker(o_4)\ ,\ mil'(o_4)\ ,\ ab_{mil'}(o_5)\ ,$
$\quad hard\_worker(o_5), mil'(o_5)\}\rangle.$

8.4.4. Belief-Bias Effect in Representation and Reasoning

$\mathcal{P}_{cig}$ represents Premise 1 and Premise 2 of S_{cig} and consists of

$$
\begin{array}{rcll}
addictive'(X) & \leftarrow & inex(X) \wedge \neg ab_{addictive'}(X). & (\mathsf{transformation}\ \&\ \mathsf{licenses}) \\
ab_{addictive'}(X) & \leftarrow & \bot. & (\mathsf{licenses}) \\
addictive(X) & \leftarrow & \neg addictive'(X) \wedge \neg ab_{addictive}(X). & (\mathsf{transformation}\ \&\ \mathsf{licenses}) \\
inex(o_1) & \leftarrow & \top. & (\mathsf{import}) \\
ab_{addictive}(o_1) & \leftarrow & \bot. & (\mathsf{licenses}\ \&\ \mathsf{doubleNeg}) \\
inex(X) & \leftarrow & cig(X) \wedge \neg ab_{inex}(X). & (\mathsf{licenses}) \\
ab_{inex}(o_2) & \leftarrow & \bot. & (\mathsf{unknownGen}\ \&\ \mathsf{licenses}) \\
cig(o_2) & \leftarrow & \top. & (\mathsf{import}) \\
cig(o_3) & \leftarrow & \top. & (\mathsf{unknownGen})
\end{array}
$$

In addition, we have the following integrity constraint:

$$\mathrm{U} \leftarrow addictive(X) \wedge addictive'(X). \qquad (\mathsf{transformation})$$

$addictive(X)$ and $addictive'(X)$ denote that X is addictive and not addictive, respectively. Regarding the first and the second premise, it is commonly known that

Cigarettes are addictive.

This belief in the context of Premise 1 leads to

If something is a cigarette, then it is abnormal. (regarding Premise 1 *of* S_{cig}*)*

$\mathcal{P}_{cig}$ is extended accordingly. The new program is

$$\mathcal{P}_{cig,\mathsf{bias}} \quad = \quad \mathcal{P}_{cig} \cup \{ab_{addictive'}(X) \leftarrow cig(X)\}.$$

The interpretation of S_{cig} together with the belief-bias effect is represented by $\mathcal{P}_{cig}^{\mathsf{bias}}$. Observe that $ab_{addictive'}(X) \leftarrow cig(X)$ overrides $ab_{addictive'}(X) \leftarrow \bot(X)$ under the weak completion of $\mathcal{P}_{cig}^{\mathsf{bias}}$. The weak completion of $\mathcal{P}_{cig}^{\mathsf{bias}}$ is shown in Box 5. Its least model is

$$\langle \{cig(o_2), cig(o_3), inex(o_1), inex(o_2)\}, \{ab_{addictive}(o_1), ab_{inex}(o_2)\} \rangle.$$

This model does not state anything about the Conclusion, that *some addictive things are not cigarettes*. Again, the Conclusion of S_{cig} is about something, which cannot be o_1, o_2 or o_3. As nothing can be derived about the relation between *addictive* and *inex* nor between *addictive* and *cig*, principle (searchAlt) of Section 8.3.2 applies: According to our background knowledge, we know that 'normal' addictive things exist, i.e. addictive things for which we do not assume anything abnormal with respect to them being addictive things. Additionally, we cannot be sure that all addictive things are normal, i.e. we know that addictive things exist for which we simply don't know whether they

Box 5. $\mathcal{P}_{cig}^{\mathsf{bias}}$ consists of the following clauses:

$$
\begin{array}{rcl}
addictive'(o_1) & \leftrightarrow & inex(o_1) \wedge \neg ab_{addictive'}(o_1). \\
addictive'(o_2) & \leftrightarrow & inex(o_2) \wedge \neg ab_{addictive'}(o_2). \\
addictive'(o_3) & \leftrightarrow & inex(o_3) \wedge \neg ab_{addictive'}(o_3). \\
addictive(o_1) & \leftrightarrow & \neg addictive'(o_1) \wedge \neg ab_{addictive}(o_1). \\
addictive(o_2) & \leftrightarrow & \neg addictive'(o_2) \wedge \neg ab_{addictive}(o_2). \\
addictive(o_3) & \leftrightarrow & \neg addictive'(o_3) \wedge \neg ab_{addictive}(o_3). \\
inex(o_1) & \leftrightarrow & cig(o_1) \wedge \neg ab_{inex}(o_1). \\
inex(o_2) & \leftrightarrow & cig(o_2) \wedge \neg ab_{inex}(o_2). \\
inex(o_3) & \leftrightarrow & cig(o_3) \wedge \neg ab_{inex}(o_3). \\
inex(o_1) & \leftrightarrow & \top. \\
ab_{addictive}(o_1) & \leftrightarrow & \bot. \\
cig(o_2) & \leftrightarrow & \top. \\
cig(o_3) & \leftrightarrow & \top. \\
ab_{inex}(o_2) & \leftrightarrow & \bot. \\
ab_{addictive'}(o_1) & \leftrightarrow & \bot \vee cig(o_1). \\
ab_{addictive'}(o_2) & \leftrightarrow & \bot \vee cig(o_2). \\
ab_{addictive'}(o_3) & \leftrightarrow & \bot \vee cig(o_3).
\end{array}
$$

are normal. We formulate this as an observation about two newly introduced constants, let's say o_4, representing normal addictive things[8] and o_5 representing addictive things for which it is unknown whether they are normal:

$$\mathcal{O} = \{addictive(o_4), \neg ab_{addictive'}(o_4), \neg ab_{addictive}(o_4), addictive(o_5)\},$$

In order to generate an explanation for $\mathcal{O}$, let us define $\mathcal{C} = \{o_1, o_2, o_3, o_4, o_5\}$. In addition to the previously listed clauses in $g\mathcal{P}_{cig}^{\mathsf{bias}}$, $\mathcal{P}_{cig,\mathsf{bias}}$ ground with respect to $\mathcal{C}$, denoted as $g\mathcal{P}_{cig,\mathsf{bias}}^{\mathcal{C}}$, consists now of the following ten clauses:

$$
\begin{array}{rcl}
addictive'(o_4) & \leftarrow & inex(o_4) \wedge \neg ab_{addictive'}(o_4). \\
ab_{addictive'}(o_4) & \leftarrow & \bot. \\
ab_{addictive'}(o_4) & \leftarrow & cig(o_4). \\
addictive(o_4) & \leftarrow & \neg addictive'(o_4) \wedge \neg ab_{addictive}(o_4). \\
inex(o_4) & \leftarrow & cig(o_4) \wedge \neg ab_{inex}(o_4). \\
addictive'(o_5) & \leftarrow & inex(o_5) \wedge \neg ab_{addictive'}(o_5). \\
ab_{addictive'}(o_5) & \leftarrow & \bot. \\
ab_{addictive'}(o_5) & \leftarrow & cig(o_5). \\
addictive(o_5) & \leftarrow & \neg addictive'(o_5) \wedge \neg ab_{addictive}(o_5). \\
inex(o_5) & \leftarrow & cig(o_5) \wedge \neg ab_{inex}(o_5).
\end{array}
$$

[8] This implies that all abnormalities about *addictive* or *addictive'* are false with respect to o_4.

Box 6. Given that $\mathsf{lm\,wc}\,\mathcal{P}_{cig,\mathsf{bias}} = \langle I^\top, I^\perp\rangle$, the least models of the weak completion of $\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}}$ together with the corresponding explanations are as follows:

$$
\begin{aligned}
&\mathsf{lm\,wc}\,(\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}} \cup \mathcal{E}_1)\\
&= \langle I^\top \cup \{\, addictive(o_4)\,,\ addictive(o_5)\,\},\\
&\quad I^\perp \cup \{\, cig(o_4)\,,\ inex(o_4)\,,\ ab_{addictive}(o_4)\,,\ ab_{addictive'}(o_4)\,,\ addictive'(o_4)\,,\\
&\quad ab_{addictive}(o_5)\,, cig(o_5), ab_{addictive'}(o_5), inex(o_5), addictive'(o_5)\}\rangle,\\
&\mathsf{lm\,wc}\,(\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}} \cup \mathcal{E}_2)\\
&= \langle I^\top \cup \{\, addictive(o_4)\,,\ addictive(o_5)\,, ab_{inex}(o_5)\},\\
&\quad I^\perp \cup \{\, cig(o_4)\,,\ inex(o_4)\,,\ ab_{addictive}(o_4)\,,\ ab_{addictive'}(o_4)\,,\ addictive'(o_4)\,,\\
&\quad ab_{addictive}(o_5)\,, inex(o_5), addictive'(o_5)\}\rangle,\\
&\mathsf{lm\,wc}\,(\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}} \cup \mathcal{E}_3)\\
&= \langle I^\top \cup \{\, addictive(o_4)\,,\ addictive(o_5)\,, cig(o_5), ab_{addictive'}(o_5)\},\\
&\quad I^\perp \cup \{\, cig(o_4)\,,\ inex(o_4)\,,\ ab_{addictive}(o_4)\,,\ ab_{addictive'}(o_4)\,,\ addictive'(o_4)\,,\\
&\quad ab_{addictive}(o_5)\,, addictive'(o_5)\}\rangle.
\end{aligned}
$$

$\mathsf{lm\,wc}\,\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}}$ does not state anything about o_4 nor o_5: All atoms about o_4 and o_5 are unknown in this least model. Given $g\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}}$, the set of abducibles, $\mathcal{A}_{\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}}}$ contains six clauses about o_4 and six clauses about o_5:

$$
\begin{array}{lll}
cig(o_4) \leftarrow \top. & ab_{addictive}(o_4) \leftarrow \top. & ab_{inex}(o_4) \leftarrow \top.\\
cig(o_4) \leftarrow \bot. & ab_{addictive}(o_4) \leftarrow \bot. & ab_{inex}(o_4) \leftarrow \bot.\\
cig(o_5) \leftarrow \top. & ab_{addictive}(o_5) \leftarrow \top. & ab_{inex}(o_5) \leftarrow \top.\\
cig(o_5) \leftarrow \bot. & ab_{addictive}(o_5) \leftarrow \bot. & ab_{inex}(o_5) \leftarrow \bot.
\end{array}
$$

The only three (minimal) explanations for $\mathcal{O}$ are

$$
\begin{aligned}
\mathcal{E}_1 &= \{cig(o_4) \leftarrow \bot, ab_{addictive}(o_4) \leftarrow \bot, ab_{addictive}(o_5) \leftarrow \bot, cig(o_5) \leftarrow \bot\},\\
\mathcal{E}_2 &= \{cig(o_4) \leftarrow \bot, ab_{addictive}(o_4) \leftarrow \bot, ab_{addictive}(o_5) \leftarrow \bot, ab_{inex}(o_5) \leftarrow \bot\},\\
\mathcal{E}_3 &= \{cig(o_4) \leftarrow \bot, ab_{addictive}(o_4) \leftarrow \bot, ab_{addictive}(o_5) \leftarrow \bot, cig(o_5) \leftarrow \top\}.
\end{aligned}
$$

The least models of the weak completion of $\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}}$ together with the corresponding explanations are shown in Box 6. The atoms highlighted in white in Box 6 are the ones which follow from all explanations, that means, these are the skeptically entailed atoms. The Conclusion of S_{add}, *Some addictive things are not cigarettes*, follows skeptically

from $\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}}$ and the observation $\mathcal{O}$: There exists an object, namely o_4, such that

$$\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}}, \mathcal{O} \models^{s}_{wcs} addictive(o_4) \wedge \neg cig(o_4)$$

and there exists another object, namely o_5, such that

$$\mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}}, \mathcal{O} \models^{s}_{wcs} addictive(o_5) \quad \text{and} \quad \mathcal{P}^{\mathcal{C}}_{cig,\mathsf{bias}}, \mathcal{O} \not\models^{s}_{wcs} cig(o_5).$$

According to Evans, Barston, and Pollard [1983], this type of syllogism is classical logically invalid, but psychologically believable and therefore causes a conflict: S_{cig} does not follow logically from the premises. Nevertheless, people are biased and search for a model that confirms their beliefs. This complies with what is entailed skeptically by $\mathcal{P}^{\mathsf{bias},\mathcal{C}}_{cig}$, $\mathcal{IC}$ and $\mathcal{O}$. Note that in the formalization of this example, the original restriction that explanations have to minimal is necessary. Consider for instance

$$\{cig(o_4) \leftarrow \bot, ab_{addictive}(o_4) \leftarrow \bot, ab_{addictive}(o_5) \leftarrow \bot, ab_{inex}(o_4) \leftarrow \top, cig(o_5) \leftarrow \bot\},$$

where, if it would be an explanation for $\mathcal{O}$, the Conclusion would not follow skeptically. However, it cannot be an explanation for $\mathcal{O}$ because it is not minimal, i.e. it is a superset of $\mathcal{E}_1$.

8.5. Conclusion

By taking the principles presented in the previous Chapter as starting point and extending them with two additional principles *background knowledge* and *search for alternative models*, we show that they can be applied to model the belief-bias effect in syllogistic human reasoning. For this purpose, we model the four examples of Evans, Barston and Pollard's [1983] syllogistic reasoning task. The belief-bias effect can be modeled in two stages: The first stage is where the belief bias seems to occur in the representational part of the syllogism, for instance in S_{vit}. In this case, the belief bias can be modeled by means of abnormality predicates. The belief bias in S_{cig} seems to occur in the representational and the reasoning part of the syllogism. The reasoning part can be modeled with skeptical abduction. Additionally, as the last example shows, explanations are required to be minimal.

One of the properties of the Weak Completion Semantics, which is different than other logic programming semantics, is that undefined atoms stay unknown instead of being false. To the best of our knowledge, the syllogistic reasoning tasks discussed so far in the literature have never accounted for providing the option 'I don't know' to the participants. As has been discussed in [Newstead, Handley, and Buck, 1999], participants who say that no valid conclusion follows might have problems to find a conclusion easily, possibly meaning that they do not know the answer. The authors also point to [Polk and Newell, 1995], who suggested that if a conclusion is stated as being not valid this could

mean that the reasoning process is exhausted. An experimental study which allows the participants to distinguish between 'I don't know' and 'not valid' might give us more insights about their reasoning processes and identify where exactly the belief bias takes effect.

Part III.

On Conditionals

9. Conditionals Evaluation System

In this chapter we present an approach, formalized in terms of an abstract reduction system, which allows us to evaluate conditionals by various procedural steps, including revision and abduction. Our focus is on conditionals in human reasoning, i.e. we want to predict the conclusions drawn by humans. We assume that in certain cases humans apply abduction and only apply revision if contradictory information is given. However, we are not aware of any approach which explicitly considers the case of a condition being *unknown*.[1]

9.1. Introduction

Consider the following scenario [Adams, 1970]: President Kennedy was killed. There was a lengthy investigation about the question whether Oswald or somebody else shot the president. In the end, it was determined that Oswald did it. Which of the following two conditionals do we easily accept?

> *If Kennedy is dead and Oswald did not shoot Kennedy,*
> *then someone else did.*[2]

and

> *If Oswald had not shot Kennedy, then someone else would have.*

According to Adams [1970] people easily accept the first conditional, whereas they reject the second conditional. *Conditionals* are statements of the form *if condition then consequence*. *Indicative conditionals* are conditionals whose condition may or may not be known to be *true*, in the sense that the condition is *true* or else *unknown*, and consequently, whose consequence also may or may not be *true*. Yet, the consequence is asserted to be *true* if the condition is *true*. On the other hand, the condition of a *subjunctive* or *counterfactual conditional* needs to be *false* [Lewis, 1973]. Only in the counterfactual circumstance of the condition being *true*, the consequence is asserted to be *true*. In the sequel, we distinguish subjunctive from indicative conditionals only by

[1] The original idea of this chapter has been published in [Dietz, Hölldobler, and Pereira, 2015c,b]. Some results in Section 9.3 have not been published and are contributions of this thesis.

[2] In the original version, the conditional does not contain the part 'If Kennedy is dead and'. Instead, it is additionally assumed that everyone knows that Kennedy was killed. For clarity about what is assumed to be known, we included this knowledge in the premise of the conditional.

the truth of their condition. If the condition of a conditional is *false*, then it is a subjunctive conditional, but not an indicative one. Otherwise, it is an indicative conditional. We distinguish between both types by grammatically expressing them in their indicative or subjunctive mood, respectively. Other than that, we do not distinguish between a subjunctive and counterfactual conditional.[3]

Note that our view is a very simplified understanding of conditionals. There are controversial discussions within the fields of philosophy and psychology where conditionals might be understood with either a narrow or a broad notion [Hoerl, McCormack, and Beck, 2011]. The narrow notion of conditionals imposes a strict distinction between counterfactual and indicative conditionals, whereas the broad notion doesn't: An indicative conditional can turn into a counterfactual conditional, depending on the context [Edgington, 2011]. Some require that counterfactuals must be in the subjunctive mood or can only be evaluated in a state that is different with respect to the current one [Woodward, 2011].

It is generally accepted that conditionals in natural language do not have the same interpretation as material or truth functional conditionals [Edgington, 1995]. Some theories have been proposed, but there is no agreement on a general one [Evans and Over, 2004]. We briefly discuss some of them. For a recent survey see [Byrne, 2016].

Ramsey [1931] proposed to test conditionals by assuming the condition hypothetically and verify whether the consequence follows. This approach is problematic in case that the current state is inconsistent with the condition. Stalnaker [1968] extended Ramsey's approach and suggested minimal revision, which can be applied for both, indicative and counterfactual conditionals. Lewis [1976] showed that Stalknaker and Thomason's [1970] counterfactual theory of possible worlds had some technical problems and developed an approach of maximal world-similarity [Lewis, 1973, 1986]. Ginsberg's [1986] possible worlds approach towards counterfactuals might be one of the first in the field of Artificial Intelligence. It has been improved by requiring relevancy [Pereira and Aparício, 1989]. The notion of relevancy will be discussed in more detail in Section 9.6. Other early approaches have been proposed in [Bench-Capon, 1989, Routen and Bench-Capon, 1991]. The logic programming approaches presented in [Baral and Hunsaker, 2007, Baral, Gelfond, and Rushton, 2009, Vennekens, Denecker, and Bruynooghe, 2009, Vennekens, Bruynooghe, and Denecker, 2010, Pereira and Saptawijaya, 2016a] are inspired by Pearl's [2000, 2011] structural theory of counterfactuals in Bayesian networks. The distinction between causal and counterfactual reasoning, based on Pearl's theory, has been extensively discussed by Sloman [2005]. Rescher [2005, 2007] presented a systematic reconstruction of the belief system, which only requires to consider immediately relevant beliefs.

[3] In English, grammatical moods can be expressed by the form of the verb in the sentence. The first conditional above (on page 155) is expressed in the indicative mood, whereas the second conditional (on page 155) is expressed in the subjunctive mood.

Yet, in the area of Cognitive Science, quite different theories of reasoning with conditionals have been presented. Oberauer [2006] compares the four most dominant ones: The Mental Model Theory [Johnson-Laird and Byrne, 2002], a dual-process variant thereof [Verschueren, Schaeken, and d'Ydewalle, 2005], the Suppositional Theory [Evans and Over, 2004] and the Probabilistic Theory presented by Oaksford, Chater, and Larkin [2000]. Oberauer observes that none of these theories seems to be more favorable than the others. Nevertheless, he claims that the following two components are essential for a model for reasoning with conditionals: The reasoning should be based on models and forward reasoning should be preferred over backward reasoning.

In the following, the methodology of our approach applied to reasoning is inspired by Pearl [2011], but does not involve probabilities, and we agree with Rescher's [2005, 2007] view to concentrate on relevant knowledge and minimally revising the current state. Before we present the central achievement of this Chapter in Section 9.3 and Section 9.5, we will first introduce a revision operator. In Section 9.4, we state several questions, which we think should be answered by psychological experiments in order to understand better how humans reason. Our hypothesis is that humans prefer a certain evaluation strategy, which we present in Section 9.5. Section 9.6 focuses on the concept of relevance in the context of conditionals.

9.2. Revision Operator

The revision operator is a concept by which we modify a given program with respect to a set of literals. We use it during the evaluation of subjunctive conditionals to revise background knowledge such that previously *false* conditions are mapped to *true* under the revised program. Somehow surprisingly, it will turn out that revision is also needed in the evaluation of indicative conditionals.

Let $\mathcal{L}$ be a finite and consistent set of ground literals. Recall that $\mathcal{L}$ is consistent iff it does not contain a pair of complementary literals. Then, given that $\mathcal{P}$ is a program, the *revision of $\mathcal{P}$ with respect to $\mathcal{L}$* is formally defined as:

$$rev(\mathcal{P}, \mathcal{L}) = (\mathcal{P} \setminus \mathsf{def}(\mathcal{L}, \mathcal{P})) \cup \{A \leftarrow \top \mid A \in \mathcal{L}\} \cup \{A \leftarrow \bot \mid \neg A \in \mathcal{L}\}.$$

When writing sets of literals, we will omit curly brackets if the set has only one element. Example 9.1 demonstrates how the revision operator works and Example 9.2 shows how it is related to Pearl's intervention in the Do-Calculus [Pearl, 2000].

Example 9.1. Consider the program $\mathcal{P}$ consisting of the following clauses:

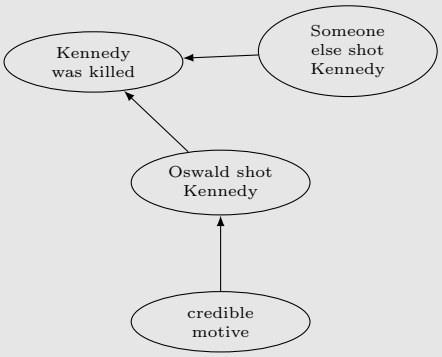
$$p \leftarrow q. \qquad q \leftarrow \top.$$

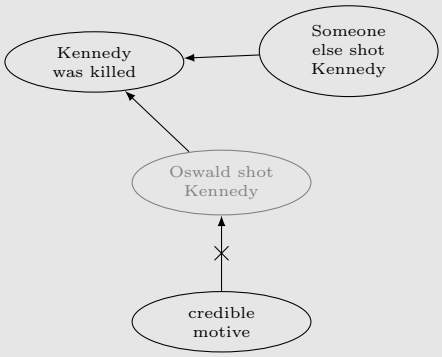
The least model of the weak completion of $\mathcal{P}$, $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}$, is $\langle\{q,p\},\emptyset\rangle$. Note that $\mathsf{def}(q,\mathcal{P}) = \{q \leftarrow \top\}$. The revision of $\mathcal{P}$ with respect to $\neg q$ is

$$rev(\mathcal{P},\neg q) \;=\; (\{p \leftarrow q, q \leftarrow \top\} \setminus \{q \leftarrow \top\}) \cup \{q \leftarrow \bot\} \;=\; \{p \leftarrow q, q \leftarrow \bot\}.$$

Accordingly, $\mathsf{lm}\,\mathsf{wc}\,rev(\mathcal{P},q)$ is $\langle\emptyset,\{p,q\}\rangle$.

Example 9.2. A counterfactual requires a hypothetical modification of the current situation according to Pearl [2000]. It is accepted if "the consequent follows after adding the antecedent hypothetically to the beliefs and the minimal required adjustments to maintain consistency of the model are made". Pearl's interventions, where the antecedent node is isolated from its parent nodes in the network and imposed to be true or false, can be understood analogously to the revision operator. Consider again: *If Oswald had not shot Kennedy, then someone else would have.* A representation of a causal relations between Kennedy was killed, Oswald shot Kennedy and someone else shot Kennedy in a bayesian network before intervention is shown on the left side, and after the intervention is shown on the right side: Here, the parent leaves are cut from the premise, the premises' truth value is imposed and the network is computed again.





Credible motive	Oswald killed Kennedy T	F
F	0.25	0.75
T	0.99	0.01

Credible motive	Oswald shot Kennedy T	F
F	0	1
T	0	1

Proposition 9.1. *Given a program $\mathcal{P}$, a finite and consistent set of ground literals $\mathcal{L}$ and a formula F, the following holds:*

1. *rev is non-monotonic: There are $\mathcal{P}$, $\mathcal{L}$, F s.t. $\mathcal{P} \models_{wcs} F$ and $rev(\mathcal{P}, \mathcal{L}) \not\models_{wcs} F$.*
2. *If for all $L \in \mathcal{L}$, $\mathsf{Im}\,\mathsf{wc}\,\mathcal{P}(L) = U$,*
   then rev is monotonic: $\mathsf{Im}\,\mathsf{wc}\,\mathcal{P} \subseteq \mathsf{Im}\,\mathsf{wc}\,rev(\mathcal{P}, \mathcal{L})$.[4]
3. *For all $L \in \mathcal{L}$, $\mathsf{Im}\,\mathsf{wc}\,rev(\mathcal{P}, \mathcal{L})(L) = \top$.*

Proof.

1. Consider Example 9.1: $\mathcal{P} \models_{wcs} p$, but $rev(\mathcal{P}, q) \not\models_{wcs} p$.
2. $\mathsf{Im}\,\mathsf{wc}\,\mathcal{P}$ and $\mathsf{Im}\,\mathsf{wc}\,rev(\mathcal{P}, \mathcal{L})$ can be computed by iterating $\Phi_{\mathcal{P}}$ and $\Phi_{rev(\mathcal{P},\mathcal{L})}$, respectively.
   We show that for all $n \in \mathbb{N}$ the relationship $\Phi_{\mathcal{P}} \uparrow n \subseteq \Phi_{rev(\mathcal{P},\mathcal{L})} \uparrow n$ holds by induction on n. If $n = 0$ we find $\Phi_{\mathcal{P}}\uparrow 0 = \langle \emptyset, \emptyset \rangle = \Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow 0$. We assume that the result holds for n and turn to the induction step:

$$\Phi_{\mathcal{P}}\uparrow(n+1) = \Phi_{\mathcal{P}}(\Phi_{\mathcal{P}}\uparrow n) = \langle I^{\top}, I^{\bot} \rangle, \tag{9.1}$$

   where

$$\begin{aligned} I^{\top} &= \{A \mid A \leftarrow body \in \mathsf{def}(A, \mathcal{P}) \text{ and } \Phi_{\mathcal{P}}\uparrow n(body) = \top\}, \\ I^{\bot} &= \{A \mid \mathsf{def}(A, \mathcal{P}) \neq \emptyset \text{ and} \\ &\qquad \text{for all } A \leftarrow body \in \mathsf{def}(A, \mathcal{P}) \text{ we find that } \Phi_{\mathcal{P}}\uparrow n(body) = \bot\}. \end{aligned}$$

   If $\mathsf{Im}\,\mathsf{wc}\,\mathcal{P}(L) = \mathrm{U}$ for all $L \in \mathcal{L}$, then, given that either $L = A$ or $L = \neg A$, A is neither in $I^{\top}$ nor in $I^{\bot}$. By the definition of the revision operator, however, A is either in $J^{\top}$ or in $J^{\bot}$, where

$$\begin{aligned} J^{\top} &= \{A \mid A \leftarrow body \in \mathsf{def}(A, rev(\mathcal{P}, \mathcal{L})) \text{ and } \Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow n(body) = \top\}, \\ J^{\bot} &= \{A \mid \mathsf{def}(A, rev(\mathcal{P}, \mathcal{L})) \neq \emptyset \text{ and} \\ &\qquad \text{for all } A \leftarrow body \in \mathsf{def}(A, rev(\mathcal{P}, \mathcal{L})) \\ &\qquad \text{we find that } \Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow n(body) = \bot\}. \end{aligned}$$

   Additionally, because $\mathcal{P}$ and $rev(\mathcal{P}, \mathcal{L})$ contain identical definitions for atoms not occurring in $\mathcal{L}$, we conclude by the induction hypothesis that $I^{\top} \subseteq J^{\top}$, $I^{\bot} \subseteq J^{\bot}$ and

$$\begin{aligned} \langle I^{\top}, I^{\bot} \rangle \subseteq \langle J^{\top}, J^{\bot} \rangle &= \Phi_{rev(\mathcal{P},\mathcal{L})}(\Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow n) \\ &= \Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow(n+1) \end{aligned} \tag{9.2}$$

   The result follows by combining (9.1) and (9.2).

[4] Proposition 5.1.9. in [Philipp, 2010] shows a related result with respect to facts and assumptions that are not in $\mathcal{P}$.

3. By the definition of the $\Phi_{\mathcal{P}}$ and the *rev* operator, after the first iteration on $\Phi_{rev(\mathcal{P},\mathcal{L})}$ for interpretation $I_1 = \langle I_1^\top, I_1^\bot \rangle$ it holds that

$$I_1^\top \supseteq \{A \mid A \in \mathcal{L}\} \quad \text{and} \quad I_1^\bot \supseteq \{A \mid \neg A \in \mathcal{L}\}.$$

As shown by Hölldobler and Kencana Ramli [2009b], there exists a least fixed point of $\Phi_{rev(\mathcal{P},\mathcal{L})}$ such that $\Phi_{rev(\mathcal{P},\mathcal{L})}(I_n) = I_n$, where $n \in \mathbb{N}$. As the $\Phi_{\mathcal{P}}$ operator is monotonic, for $I_n = \langle I_n^\top, I_n^\bot \rangle$ it holds that

$$I_n^\top \supseteq I_1^\top \quad \text{and} \quad I_n^\bot \supseteq I_1^\bot.$$

This implies that for every $A \in \mathcal{L}$, $\mathsf{lm\,wc}\, rev(\mathcal{P},\mathcal{L})(A) = \top$ and, likewise, for every $\neg A \in \mathcal{L}$, $\mathsf{lm\,wc}\, rev(\mathcal{P},\mathcal{L})(\neg A) = \top$. □

9.3. ARSC – Abstract Reduction System

The abstract reduction system for conditionals, ARSC, is a general characterization for deriving the truth value of a certain conditional, possibly by means of transforming the program with respect to the condition of this conditional. We consider conditionals of the form *if* $\mathcal{C}$ *then* $\mathcal{D}$, denoted as $cond(\mathcal{C},\mathcal{D})$, where the condition $\mathcal{C}$ and the consequence $\mathcal{D}$ are finite and consistent sets of literals. Recall that a set of literals is consistent, when it does not contain a pair of complementary literals. Conditionals are evaluated with respect to some background information specified as a program and a set of integrity constraints. In the sequel, let $\mathcal{P}$ be a program, $\mathcal{IC}$ be a finite set of integrity constraints, and $\mathsf{lm\,wc}\,\mathcal{P}$ be the least model of the weak completion of $\mathcal{P}$. For simplicity, if not stated otherwise, we will assume that $\mathcal{IC} = \emptyset$. In the following, $\mathsf{lm\,wc}\,\mathcal{P}$ always satisfies $\mathcal{IC}$.

An *abstract reduction system* is defined as a pair $(A, \{\longrightarrow_\alpha | \ \alpha \in I\})$, where the *reduction* (or *relation*) $\longrightarrow_\alpha$ is a binary relation over the set A, i.e. $\longrightarrow_\alpha \subseteq A \times A$, indexed by a set I [Baader and Nipkow, 1998, Klop, Bezem, and Vrijer, 2001]. We write $a \longrightarrow_\alpha a'$ instead of $(a,a') \in \longrightarrow_\alpha$. The union $\bigcup\{\longrightarrow_\alpha | \ \alpha \in I\}$ is written as $\longrightarrow_I$. If there is just one reduction, we simply write $(A, \longrightarrow)$. The composition of two reductions $\longrightarrow_\alpha$ and $\longrightarrow_\beta$, is denoted by $\longrightarrow_\alpha \cdot \longrightarrow_\beta$. $a \longrightarrow_\alpha \cdot \longrightarrow_\beta a''$ iff there is some $a' \in A$, such that $a \longrightarrow_\alpha a' \longrightarrow_\beta a''$. We express the i-fold composition of some reduction $\longrightarrow_\alpha$ by $\xrightarrow{i}_\alpha$, where $i \geq 0$. $a \xrightarrow{0}_\alpha a'$ iff $a = a'$. If we want to specify that the composition of $\longrightarrow_\alpha$ can be either i-fold or j-fold, we write $\xrightarrow{i/j}_\alpha$, where $i, j \geq 0$.

The initial and intermediate states in ARSC are tuples of the form $S = \langle \mathcal{P}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle$, where $\mathcal{P}$ is a program, $\mathcal{IC}$ a set of integrity constraints and $cond(\mathcal{C},\mathcal{D})$ the conditional under consideration. The final states are of the form $\top$, $\bot$ and U, corresponding to *true*, *false* and *unknown*, respectively.

The reductions in ARSC are indexed by the set $\mathcal{R} = \{a, r, s, c\}$, where a stands for abduction, r for revision, s for the final state and c for counterfactual. When the condition can be explained, then abduction, $\longrightarrow_a$, is applied. The reduction $\longrightarrow_r$ leads to revision with respect to unknown literals. When the condition is true, then $\longrightarrow_s$ is applied and leads to one of the final states. We can handle counterfactuals by the reduction $\longrightarrow_c$: More precisely, in case the condition of the conditional is *false*, we revise the program in order to satisfy the condition of the conditional:

- $\langle \mathcal{P}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle \longrightarrow_a \langle \mathcal{P} \cup \mathcal{E}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle$
  iff $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \mathrm{U}$ and there exists $\mathcal{O} \subseteq \mathcal{C}$ where $\mathcal{O} \neq \emptyset$,
  such that for each $L \in \mathcal{O}$ we find $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(L) = \mathrm{U}$
  and $\mathcal{E}$ explains $\mathcal{O}$ in the abductive framework $\langle \mathcal{P}, \mathcal{A}_\mathcal{P}, \mathcal{IC}, \models_{wcs} \rangle$.

- $\langle \mathcal{P}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle \longrightarrow_r \langle rev(\mathcal{P}, \mathcal{L}), \mathcal{IC}, \mathcal{C} \setminus \mathcal{L}, \mathcal{D} \rangle$
  iff $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \mathrm{U}$ and there exists $\mathcal{L} \subseteq \mathcal{C}$, where $\mathcal{L} \neq \emptyset$,
  such that for each $L \in \mathcal{L}$ we find $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(L) = \mathrm{U}$.

- $\langle \mathcal{P}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle \longrightarrow_s \mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{D})$ iff $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \top$.

- $\langle \mathcal{P}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle \longrightarrow_c \langle rev(\mathcal{P}, \mathcal{L}), \mathcal{IC}, \mathcal{C} \setminus \mathcal{L}, \mathcal{D} \rangle$
  iff $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \bot$, where $\mathcal{L} = \{L \in \mathcal{C} \mid \mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(L) = \bot\}$.

Finally ARSC is defined as $(\mathcal{S}, \longrightarrow_\mathcal{R})$ where the set of states $\mathcal{S}$ is $\{\top, \bot, \mathrm{U}\} \cup \{\langle \mathcal{P}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle \mid \mathcal{P}, \mathcal{IC}, \mathcal{C}$ and $\mathcal{D}$ as defined above.$\}$ and the set of reductions $\longrightarrow_\mathcal{R}$ is $\bigcup\{\longrightarrow_\alpha \mid \alpha \in \mathcal{R}\}$. Recall that $\mathcal{R} = \{a, r, s, c\}$. Note that the reduction $\longrightarrow_c$ revises the program non-monotonically (see Proposition 9.1), therefore explanations generated by the reduction $\longrightarrow_a$ may not persist and, hence, observations cannot be deleted from $\mathcal{C}$ even if they are explained. Example 9.3 discusses such a case.

9.3.1. Properties

Abstract reduction systems can have various properties, among others, confluence and termination. An abstract reduction system is said to be confluent if all its state can be reduced to the same (successor) state. Termination means that we can not apply the reductions infinitely many times. We will show whether these properties hold in ARSC.

Theorem 9.2. *Let $\langle \mathcal{P}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle$ be a state in $(\mathcal{S}, \longrightarrow_\mathcal{R})$ and $\langle rev(\mathcal{P}, \mathcal{L}), \mathcal{IC}, \mathcal{C} \setminus \mathcal{L}, \mathcal{D} \rangle$ its successor state by applying $\longrightarrow_c$ or $\longrightarrow_r$. If $\langle \mathcal{P}', \mathcal{IC}', \mathcal{C}', \mathcal{D}' \rangle$ is a successor state of $\langle rev(\mathcal{P}, \mathcal{L}), \mathcal{IC}, \mathcal{C} \setminus \mathcal{L}, \mathcal{D} \rangle$, then for all $L \in \mathcal{L}$, $L \notin \mathcal{C}'$.*

Proof.
We need to distinguish between the following two cases:

1. If $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \bot$, then $\longrightarrow_c$ is applied and the definitions for $\mathcal{L} = \{L \in \mathcal{C} \mid \mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(L) = \bot\}$ have been replaced by facts or assumptions such that the least model of the weak completion of the revised program maps each literal occurring in $\mathcal{L}$ to $\top$. As $\mathcal{C}$ is consistent, these new facts or assumptions will never be revised again.

2. If $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \mathrm{U}$ and $\longrightarrow_r$ was applied, then the definitions for $\mathcal{L} \subseteq \mathcal{C}$ have been replaced by facts or assumptions such that the least model of the weak completion of the revised program maps each literal occurring in $\mathcal{L}$ to $\top$, where for each $L \in \mathcal{L}$ we find $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(L) = \mathrm{U}$. As $\mathcal{C}$ is consistent, the new facts or assumptions will never be revised again. □

As Example 9.3 shows, Theorem 9.2 does not extend to $\longrightarrow_a$. Let us recall again the property of monotonicity with respect to the reductions in ARSC. As shown in Proposition 9.1, we can easily see that applications of $\longrightarrow_c$ are non-monotonic, i. e., after applying $\longrightarrow_c$ to a state, a previously entailed formula, is possibly not entailed anymore. On the other hand, applications of $\longrightarrow_a$ are monotonic:

Proposition 9.3. *Let $\langle \mathcal{P}, \mathcal{A}_\mathcal{P}, \mathcal{IC}, \models_{wcs} \rangle$ be an abductive framework and let $\mathcal{E} \subseteq \mathcal{A}_\mathcal{P}$, then*

$$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P} \subseteq \mathsf{lm}\,\mathsf{wc}\,(\mathcal{P} \cup \mathcal{E}).^5$$

Proof.
As $\mathcal{E} \subseteq \mathcal{A}_\mathcal{P}$, $\mathcal{E}$ is unknown in $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}$. By induction on $n \in \mathbb{N}$ one can show that $\Phi_\mathcal{P} \uparrow n \subseteq \Phi_{\mathcal{P}\cup\mathcal{E}} \uparrow n$, where $\Phi_\mathcal{P} \uparrow 0 = \langle \emptyset, \emptyset \rangle$ and $\Phi_\mathcal{P} \uparrow (n+1) = \Phi_\mathcal{P}(\Phi_\mathcal{P} \uparrow n)$. The result follows immediately. □

Lemma 9.4. *Let $\langle \mathcal{P}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle$ be a state in $(\mathcal{S}, \longrightarrow_\mathcal{R})$ and $\langle \mathcal{P} \cup \mathcal{E}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle$ its successor state by applying $\longrightarrow_a$. The set of abducibles decreases after the application of $\longrightarrow_a$, i.e. $\mathcal{A}_{\mathcal{P}\cup\mathcal{E}} \subset \mathcal{A}_\mathcal{P}$.*

Proof.
According to Proposition 9.3, $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P} \subseteq \mathsf{lm}\,\mathsf{wc}\,(\mathcal{P} \cup \mathcal{E})$, thus the number of atoms which are unknown in the least model of the weak completion of $\mathcal{P}$ are reduced in each application of $\longrightarrow_a$. Because $\mathcal{E} \neq \emptyset$ and by the definition of the set of abducibles, we find that $\mathcal{A}_{\mathcal{P}\cup\mathcal{E}} \subset \mathcal{A}_\mathcal{P}$. □

Similarly, as only unknown literals are revised when applying $\longrightarrow_r$, applications on $\longrightarrow_r$ are monotonic as well.

Lemma 9.5. *Let $\langle \mathcal{P}, \mathcal{IC}, \mathcal{C}, \mathcal{D} \rangle$ be a state in $(\mathcal{S}, \longrightarrow_\mathcal{R})$. $\mathcal{C}$ decreases after each application of $\longrightarrow_r$ or $\longrightarrow_c$.*

[5] This corresponds to Proposition 5.1.9 in [Philipp, 2010].

Example 9.3. Let $\mathcal{P}_1$ consist of the following two clauses:

$$\begin{array}{lcl} a & \leftarrow & b. \\ b & \leftarrow & c. \end{array}$$

Let us assume that $\mathcal{IC} = \emptyset$, $\mathcal{C} = \{a, \neg b\}$ and $\mathcal{D} = c$. As

$$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_1 = \langle \emptyset, \emptyset \rangle,$$

$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_1(\mathcal{C}) = \mathrm{U}$ and thus we may apply $\longrightarrow_a$ with $\mathcal{O} = a$.
$\mathcal{A}_{\mathcal{P}_1}$ consists of the following two clauses:

$$\begin{array}{lcl} c & \leftarrow & \top. \\ c & \leftarrow & \bot. \end{array}$$

Accordingly, $\mathcal{O}$ can be explained by $\mathcal{E} = \{c \leftarrow \top\}$:

$$\langle \mathcal{P}_1, \emptyset, \{a, \neg b\}, c \rangle \longrightarrow_a \langle \mathcal{P}_2, \emptyset, \{a, \neg b\}, c \rangle,$$

where $\mathcal{P}_2 = \mathcal{P}_1 \cup \mathcal{E}$. We find

$$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_2 = \langle \{a, b, c\}, \emptyset \rangle$$

and thus $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_2(\{a, \neg b\}) = \bot$. Now we can only apply $\longrightarrow_c$ with $\mathcal{C} = \{\neg b\}$, which is mapped to *false* under $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_2$:

$$\langle \mathcal{P}_2, \emptyset, \{a, \neg b\}, c \rangle \longrightarrow_c \langle \mathcal{P}_3, \emptyset, a, c \rangle,$$

where $\mathcal{P}_3 = \mathit{rev}(\mathcal{P}_2, \neg b) = \{a \leftarrow b, b \leftarrow \bot, c \leftarrow \top\}$. We find

$$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_3 = \langle c, \{b, a\} \rangle$$

and, hence, $a \in \mathcal{C}$ is no longer assigned to $\top$ and must be re-considered:

$$\langle \mathcal{P}_3, \emptyset, a, c \rangle \longrightarrow_c \langle \{a \leftarrow \top, b \leftarrow \bot, c \leftarrow \top\}, \emptyset, \emptyset, c \rangle \longrightarrow_s \top,$$

where the corresponding least model of the weak completion is

$$\langle \{a, c\}, \{b\} \rangle.$$

Proof.
Recall that $\mathcal{C}$ is finite and no reduction increases $\mathcal{C}$. Accordingly, there cannot be an infinite chain of applications on $\longrightarrow_r$ or $\longrightarrow_c$. The resulting state after applying $\longrightarrow_r$, is $\langle rev(\mathcal{P}, \mathcal{L}), \mathcal{IC}, \mathcal{C} \setminus \mathcal{L}, \mathcal{D} \rangle$. As $\mathcal{L}$ is never empty and by Theorem 9.2, each $L \in \mathcal{C}$ only occurs once in $\mathcal{L}$, applications of $\longrightarrow_r$ or $\longrightarrow_c$, reduce the number of literals occurring in $\mathcal{C}$. □

ARSC terminates if the set of reductions $\longrightarrow_{\mathcal{R}}$, is not applicable infinitely many times. In order to show this, we rely on a result shown in [Baader and Nipkow, 1998]. Accordingly, it is sufficient to show that there is a monotone embedding of $(\mathcal{S}, \longrightarrow_{\mathcal{R}})$ into $(\mathbb{N}, >)$. Given the two sets $\mathcal{S}$ and $\mathbb{N}$, a *monotone embedding* of $(\mathcal{S}, \longrightarrow_{\mathcal{R}})$ into $(\mathbb{N}, >)$ is a mapping $\varphi : \mathcal{S} \longrightarrow \mathbb{N}$ such that $s \longrightarrow_{\mathcal{R}} s'$ implies that $\varphi(s) > \varphi(s')$, where $s, s' \in \mathcal{S}$ (in [Baader and Nipkow, 1998], φ is also called measure function). If there exists such a mapping φ from $(\mathcal{S}, \longrightarrow_{\mathcal{R}})$ into $(\mathbb{N}, >)$, then $\longrightarrow_{\mathcal{R}}$ terminates.

Lemma 9.6 (Lemma 2.3.3 in [Baader and Nipkow, 1998])**.** *A finitely branching reduction terminates iff there is a monotone embedding into* $(\mathbb{N}, >)$.

A reduction is finitely branching if each state has only finitely many immediate successor states [Baader and Nipkow, 1998]. For ARSC, $(\mathcal{S}, \longrightarrow_{\mathcal{R}})$, that means the following: For each reductions $\longrightarrow_\alpha$, $\alpha \in \mathcal{R}$, for each state $s \in \mathcal{S}$, the set $\{s' \in \mathcal{S} \mid s \longrightarrow_\alpha s'\}$ of immediate successor states of s (or one-step reducts of s [Klop, Bezem, and Vrijer, 2001]) is finite. It is easy to see that the set of reductions $\longrightarrow_{\mathcal{R}}$ is finitely branching.

Lemma 9.7. *There is a monotone embedding from* $(\mathcal{S}, \longrightarrow_{\mathcal{R}})$ *into* $(\mathbb{N}, >)$.

Proof.
We define the following mapping $\varphi : S \to \mathbb{N}$:

$$\varphi(S) \quad = \quad \begin{cases} 0 & \text{if } S \in \{\top, \bot, \mathrm{U}\}, \\ 1 + |\mathcal{A}_{\mathcal{P}}| + |\mathcal{C}| & \text{otherwise.} \end{cases}$$

Applications of the reduction $\longrightarrow_s$ yield a final state. By Lemma 9.4, $\mathcal{A}_{\mathcal{P}}$ decreases after each application of $\longrightarrow_a$ and by Lemma 9.5, $\mathcal{C}$ decreases after each application of $\longrightarrow_r$ or $\longrightarrow_c$ and the set of abducibles does not increase. As there is no reduction which either increases $\mathcal{A}_{\mathcal{P}}$ or $\mathcal{C}$, φ is a monotone mapping into $\mathbb{N}$, i.e. $S \longrightarrow_{\mathcal{R}} S'$ implies $\varphi(S) > \varphi(S')$. □

Corollary 9.8. ARSC *is a finitely branching reduction and terminates.*

Proof.
Follows immediately from Lemma 9.6 and Lemma 9.7. □

Corollary 9.9. *Derivations in* ARSC *are of the form*

$$\{\longrightarrow_a, \longrightarrow_r, \longrightarrow_c\}^n \cdot \longrightarrow_s$$

Example 9.4. Reconsider Example 9.3, but select $\mathcal{O} = \{\neg b\}$ in the first step. $\mathcal{O}$ can be explained by $\mathcal{E}_2 = \{c \leftarrow \bot\}$ and we obtain

$$\langle \mathcal{P}_1, \emptyset, \{a, \neg b\}, c\rangle \longrightarrow_a \langle \mathcal{P}_4, \emptyset, \{a, \neg b\}, c\rangle,$$

where $\mathcal{P}_4 = \mathcal{P}_1 \cup \mathcal{E}_2 = \{a \leftarrow b, b \leftarrow c, c \leftarrow \bot\}$. We find

$$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_4 = \langle \emptyset, \{a, b, c\}\rangle$$

and apply $\longrightarrow_c$ by revising the definition of a:

$$\langle \mathcal{P}_4, \emptyset, \{a, \neg b\}, c\rangle \longrightarrow_c \langle \{a \leftarrow \top, b \leftarrow c, c \leftarrow \bot\}, \emptyset, \neg b, c\rangle \longrightarrow_s \bot.$$

Nevertheless, it is also possible to reduce the initial state to *unknown*:

$$\langle \mathcal{P}_1, \emptyset, \{a, \neg b\}, c\rangle \longrightarrow_r \langle \{a \leftarrow \top, b \leftarrow \bot\}, \emptyset, \emptyset, c\rangle \longrightarrow_s \mathrm{U}.$$

Theorem 9.10. ARSC *is not confluent.*

Proof.
Consider Example 9.4: As $\bot$ and U are not further reducible, it follows that ARSC is not confluent. □

The *Firing Squad* example in the following section shows that the same conditional can be evaluated to *true*, *false* or *unknown*, even if the program stays the same and only the order in which the reductions are applied changes.

9.3.2. Modeling Well-known Examples

We will now discuss two examples which have been extensively discussed in the literature and show how we can evaluate them with ARSC.

Shooting of Kennedy Let us reconsider the example from the introduction of this Chapter. The scenario is represented by program $\mathcal{P}_5$, which consists of the following five clauses:

$$\begin{aligned} k &\leftarrow os \wedge \neg ab_1.\\ ab_1 &\leftarrow \bot.\\ k &\leftarrow ses \wedge \neg ab_2.\\ ab_2 &\leftarrow \bot.\\ os &\leftarrow \top. \end{aligned}$$

The abbreviations k, os and ses mean *Kennedy was killed*, *Oswald shot Kennedy* and *Someone else shot Kennedy*, respectively. ab_1 and ab_2 are the abnormality predicates.

The least model of the weak completion of $\mathcal{P}$, $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_5$ is

$$\langle\{os, k\}, \{ab_1, ab_2\}\rangle.$$

Consider again the second counterfactual conditional from the introduction:

If Oswald had not shot Kennedy, then someone else would have.

Its condition $\neg os$ is *false* under $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_5$ and, hence, we view it as a counterfactual:

$$\langle\mathcal{P}_5, \emptyset, \neg os, ses\rangle \longrightarrow_c \langle\mathcal{P}_6, \emptyset, \emptyset, ses\rangle \longrightarrow_s \mathit{unknown},$$

where $\mathcal{P}_6$ is

$$rev(\mathcal{P}_5, \neg os) \quad = \quad (\mathcal{P}_5 \setminus \{os \leftarrow \top\}) \cup \{os \leftarrow \bot\}.$$

As *ses* is mapped to *unknown* under $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_6 = \langle\emptyset, \{os, ab_1, ab_2\}\rangle$. The conditional is *unknown* as well, which in this case, is the only possible reduction. Now consider the conditional

If Kennedy is dead and Oswald did not shot Kennedy,
then someone else did.

Its condition $\{k, \neg os\}$ is still *false* under $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_5$ and we obtain

$$\langle\mathcal{P}_5, \emptyset, \{k, \neg os\}, ses\rangle \longrightarrow_c \langle\mathcal{P}_6, \emptyset, k, ses\rangle.$$

Because $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_6(k) = \mathrm{U}$ we may try to explain k in the abductive framework $\langle\mathcal{P}_6, \{ses \leftarrow \top, ses \leftarrow \bot\}, \emptyset, \models_{wcs}\rangle$ and find that

$$\mathcal{E}_3 \quad = \quad \{ses \leftarrow \top\}$$

is the only minimal explanation for k:

$$\langle\mathcal{P}_6, \emptyset, k, ses\rangle \longrightarrow_a \langle\mathcal{P}_6 \cup \mathcal{E}_3, \emptyset, k, ses\rangle \longrightarrow_s \top,$$

where *ses* is mapped to *true* under $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_6 \cup \mathcal{E}_3 = \langle\{ses, k\}, \{os, ab_1, ab_2\}\rangle$. Instead of abduction we could have applied revision:

$$\langle\mathcal{P}_6, \emptyset, k, ses\rangle \longrightarrow_r \langle\mathcal{P}_7, \emptyset, \emptyset, ses\rangle \longrightarrow_s \mathit{unknown},$$

where $\mathcal{P}_7$ is

$$rev(\mathcal{P}_6, k) \quad = \quad \{k \leftarrow \top, os \leftarrow \bot, ab_1 \leftarrow \bot, ab_2 \leftarrow \bot\}$$

and *ses* is mapped to *unknown* under $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_7 = \langle k, \{os, ab_1, ab_2\}\rangle$.

Firing Squad The following example shows that in ARSC it is possible to evaluate exactly the same conditional to *unknown*, *true* or *false*, even though the program stays the same, and only the order in which the reductions are applied changes. We assume

that humans prefer a certain evaluation strategy with respect to conditionals, which we will then propose in the next section. Pearl [2000] presents the so-called *Firing Squad example: If the court orders an execution (e), then the captain will give the signal (s) upon which rifleman A will shoot the prisoner (ra) and rifleman B will shoot the prisoner (rb). Consequently, the prisoner will be dead (d).* We assume that the court's decision is *unknown*, that the captain is law-abiding, that both riflemen are accurate, alert and law-abiding, and that the prisoner is unlikely to die from any other causes. Altogether, we obtain the program $\mathcal{P}_8$:

$$\begin{array}{rcl} s & \leftarrow & e \wedge \neg ab_1. \\ ra & \leftarrow & s \wedge \neg ab_2. \\ rb & \leftarrow & s \wedge \neg ab_3. \\ d & \leftarrow & ra \wedge \neg ab_4. \\ d & \leftarrow & rb \wedge \neg ab_5. \\ ab_1 & \leftarrow & \bot. \\ ab_2 & \leftarrow & \bot. \\ ab_3 & \leftarrow & \bot. \\ ab_4 & \leftarrow & \bot. \\ ab_5 & \leftarrow & \bot. \end{array}$$

$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_8$ is

$$\langle \emptyset, \{ab_1, ab_2, ab_3, ab_4, ab_5\}\rangle.$$

Consider the conditional

> *If the captain gave no signal and rifleman A decides to shoot, then the court did not order an execution.*

Its condition $\{\neg s, ra\}$ is *unknown* under $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_8$ and, hence, we view it as an indicative conditional. We can revise $\mathcal{P}_8$ with respect to $\{\neg s, ra\}$ to obtain

$$\langle \mathcal{P}_8, \emptyset, \{\neg s, ra\}, \neg e\rangle \longrightarrow_r \langle rev(\mathcal{P}_8, \{\neg s, ra\}), \emptyset, \emptyset, \neg e\rangle \longrightarrow_s unknown, \tag{9.3}$$

where $\mathsf{lm}\,\mathsf{wc}\,rev(\mathcal{P}_8, \{\neg s, ra\})$ is

$$\langle \{ra, d\}, \{s, ab_1, ab_2, ab_3, ab_4, ab_5\}\rangle$$

and, hence, $\neg e$ is *unknown*. Alternatively, we can revise $\mathcal{P}_8$ with respect to $\neg s$ first to obtain

$$\begin{array}{rcll} \langle \mathcal{P}_8, \emptyset, \{\neg s, ra\}, \neg e\rangle & \longrightarrow_r & \langle rev(\mathcal{P}_8, \neg s), \emptyset, ra, \neg e\rangle & \\ & \longrightarrow_c & \langle rev(rev(\mathcal{P}_8, \neg s), ra), \emptyset, \emptyset, \neg e\rangle & (9.4) \\ & \longrightarrow_s & unknown, & \end{array}$$

where $\mathsf{lm}\,\mathsf{wc}\,rev(\mathcal{P}_8, \neg s)$ is

$$\langle \emptyset, \{s, ra, rb, d, ab_1, ab_2, ab_3, ab_4, ab_5\}\rangle$$

and, hence, *ra* is *false*. The remaining conditional $cond(ra, \neg e)$ has become a counterfactual with respect to the program $rev(\mathcal{P}_8, \neg s)$ and, consequently, the definition for *ra* is revised. As another alternative, we can revise $\mathcal{P}_8$ with respect to *ra* first to obtain

$$\begin{aligned} \langle \mathcal{P}_8, \emptyset, \{\neg s, ra\}, \neg e\rangle &\longrightarrow_r \langle rev(\mathcal{P}_8, ra), \emptyset, \neg s, \neg e\rangle \\ &\longrightarrow_a \langle rev(\mathcal{P}_8, ra) \cup \{e \leftarrow \bot\}, \emptyset, \neg s, \neg e\rangle \\ &\longrightarrow_s \top, \end{aligned} \tag{9.5}$$

where $\mathsf{lm}\,\mathsf{wc}\, rev(\mathcal{P}_8, ra)$ is

$$\langle \{ra, d\}, \{ab_1, ab_2, ab_3, ab_4, ab_5\}\rangle$$

and, hence, $\neg s$ remains *unknown*. We could apply again revision leading to the same result as in the previous cases, but we apply abduction to explain $\neg s$ by $\{e \leftarrow \bot\}$, which yields a *true* conditional. The condition $\{\neg s, ra\}$ cannot be explained in the abductive framework $\langle \mathcal{P}_8, \{e \leftarrow \top, e \leftarrow \bot\}, \emptyset, \models_{wcs}\rangle$. But $\{e \leftarrow \bot\}$ explains $\neg s$ and we obtain

$$\begin{aligned} \langle \mathcal{P}_8, \emptyset, \{\neg s, ra\}, \neg e\rangle &\longrightarrow_a \langle \mathcal{P}_8 \cup \{e \leftarrow \bot\}, \emptyset, \{\neg s, ra\}, \neg e\rangle \\ &\longrightarrow_c \langle rev(\mathcal{P}_8 \cup \{e \leftarrow \bot\}, ra), \emptyset, \neg s, \neg e\rangle \\ &\longrightarrow_s \top, \end{aligned} \tag{9.6}$$

where $\mathsf{lm}\,\mathsf{wc}\, \mathcal{P}_8 \cup \{e \leftarrow \bot\}$ is

$$\langle \emptyset, \{e, s, ra, rb, d, ab_1, ab_2, ab_3, ab_4, ab_5\}\rangle$$

and, hence, *ra* is *false*. As final alternative, we observe that $\{e \leftarrow \top\}$ explains *ra* and we obtain

$$\begin{aligned} \langle \mathcal{P}_8, \emptyset, \{\neg s, ra\}, \neg e\rangle &\longrightarrow_a \langle \mathcal{P}_8 \cup \{e \leftarrow \top\}, \emptyset, \{\neg s, ra\}, \neg e\rangle \\ &\longrightarrow_c \langle rev(\mathcal{P}_8 \cup \{e \leftarrow \top\}, \neg s), \emptyset, ra, \neg e\rangle \\ &\longrightarrow_c \langle rev(rev(\mathcal{P}_8 \cup \{e \leftarrow \top\}, \neg s), ra\}, \emptyset, \emptyset, \neg e\rangle \\ &\longrightarrow_s \bot, \end{aligned} \tag{9.7}$$

where $\mathsf{lm}\,\mathsf{wc}\, \mathcal{P}_8 \cup \{e \leftarrow \top\}$ is

$$\langle \{e, s, ra, rb, d\}, \{ab_1, ab_2, ab_3, ab_4, ab_5\}\rangle$$

and, hence, $\neg s$ is *false*. After revising the program with respect to $\neg s$, *ra* is *false* and we need to revise the program once more.

The least models of the weak completion of the last programs in the various reduction sequences are shown in Table 9.1. The first row of Table 9.1 indicates which reduction has been applied with respect to which set of literals. Consider the first reduction, $\rightarrow_{r\{\overline{s}, ra\}}$: $\mathcal{P}_8$ is revised with respect to $\{\overline{s}, ra\}$. We have omitted the final application of $\longrightarrow_s$ and have indexed the remaining reductions by the conditions that were revised or explained.

	$\rightarrow_{r\{\overline{s},ra\}}$	$\rightarrow_{r\{\overline{s}\}} \rightarrow_{c\{ra\}}$	$\rightarrow_{r\{ra\}} \rightarrow_{a\{\overline{s}\}}$	$\rightarrow_{a\{\overline{s}\}} \rightarrow_{c\{ra\}}$	$\rightarrow_{a\{ra\}} \rightarrow_{c\{\overline{s}\}} \rightarrow_{c\{ra\}}$
s	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
ra	$\top$	$\top$	$\top$	$\top$	$\top$
d	$\top$	$\top$	$\top$	$\top$	$\top$
rb	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
e	U	U	$\bot$	$\bot$	$\top$

Table 9.1.: The least models of the weak completion in the last non-final states in the *Firing Squad* example. The derivation shown in the grey box is our preferred one; it is computed by MRFA which will be discussed in Section 9.5.

The atoms s and ra are always *false* and *true*, respectively. d is always *true* as it depends on ra (and rb). rb is always *false* as it depends on s. Yet, e may take any truth value depending on the sequence in which the conditions are considered and on the reductions that are applied.

The conditional

> *if the captain gave no signal and rifleman A decides to shoot,*
> *then rifleman B will not shoot and the prisoner will be dead.*

will always be evaluated as *true*. The situation will change if it becomes known that *a broken firing pin leads to a malfunctioning rifle*. In this case, $\mathcal{P}_8$ is updated by replacing the definition of ab_4 with $ab_4 \leftarrow b$.[6] If rifleman A decides to shoot now, then it is *unknown* whether the prisoner will die as b is *unknown*. If $b \leftarrow \top$ is added to the program, then the prisoner will not die. Section 9.6 discusses an extension, which allows to abduce *unknown* consequences.

Table 9.2 shows the dependency graph of the program $\mathcal{P}_8$. Revision cuts the dependencies from a particular node and assigns *true* or *false* to the node. Abduction assigns *true* or *false* to the node marked e.

9.4. Need for Experimental Data

Although many papers and books have been written about conditionals, we are unaware of psychological experiments that help us to identify adequate strategies for the application of reductions in ARSC and to determine how humans evaluate conditionals in examples like the *Shooting of Kennedy* or the *Firing Squad*. As we have discussed in Chapter 5, experimental data are available for examples for psychological tasks such as the Suppression Task or the Selection Task, but these tasks are considerably simpler

[6] We could update the definition of ab_5 as well, but we should then identify the firing pins of the different rifles.

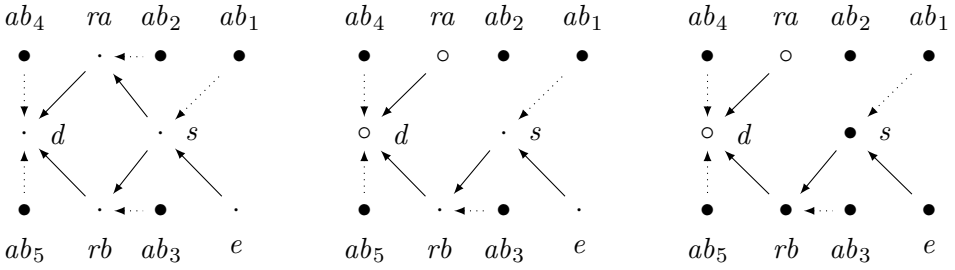


Table 9.2.: Positive dependencies are depicted by solid arrows, negative dependencies by dotted arrows. •, ·, and ∘ denote nodes, which are mapped to ⊥, U and ⊤, respectively.
(Left) The dependency graph of $\mathcal{P}_8$. The leaf node marked e is undefined, whereas all other nodes are defined. (Middle) The dependency graph of $rev(\mathcal{P}_8, ra)$: ra does not depend on s and ab_2 anymore and is mapped to ⊤. (Right) The dependency graph of $rev(\mathcal{P}_8, ra) \cup \{e \leftarrow \bot\}$.

than the conditionals discussed in the *Firing Squad* example. From our perspective, the following questions should be evaluated:

- *Do humans reason with multi-valued logics and, if they do, which multi-valued logic are they using? Can an answer 'I don't know' be qualified as a truth value assignment or is it a meta-remark?*
- *What do we have to tell humans such that they fully understand the background information including, e.g., the dependency graph in the Firing Squad example?*
- *Do humans apply abduction and/or revision if the condition of a conditional is unknown and, if they apply both, do they prefer one over the other? Do they prefer skeptical over credulous abduction? Do they prefer minimal revision?*
- *How important is the order in which multiple conditions of a conditional are considered?*
- *Do humans consider abduction and/or revision steps, which turn an indicative conditional into a subjunctive one like in the second, fourth and fifth reduction sequence of the Firing Squad example?*

In order to explore these questions, it is unavoidable to actually carry out reasoning experiments with humans. Unfortunately this task is beyond the scope of this thesis. Usually the composition of such an experiment requires a prior hypothesis, which thereafter should to be tested. In the following, we will develop such a hypothesis.

We believe that humans do reason with a third truth value. As we have shown in Chapter 5, 6 and 8, various episodes from human reasoning can be adequately modeled under the Weak Completion Semantics and, moreover, in some of these tasks, skeptical

abduction had to be applied. We believe that *minimal revision followed by abduction* is applied if the conditions of a conditional are *unknown*. Finally, we believe that humans do not consider abduction and/or revision steps that turn an indicative conditional into a subjunctive one. Altogether, we believe that humans prefer a particular strategy in evaluating conditionals: They do not consider derivations as stated in Corollary 9.9, but rather search for derivations of the form $\xrightarrow{n}_c \cdot \xrightarrow{0/1}_r \cdot \xrightarrow{0/1}_a \cdot \xrightarrow{0/1}_s$, where $n \in \mathbb{N}^0$ and $\longrightarrow_c$ is only applied if needed. In other words, the reduction $\longrightarrow_c$ is only applied if the given conditional is a counterfactual, in which case the reduction may be applied several times because *unknown* conditions may be turned into *false* ones by applications of $\longrightarrow_c$. However, as soon as the condition of a conditional is mapped to *true* or *unknown*, $\longrightarrow_c$ will not be applied anymore. Additionally, because $rev(rev(\mathcal{P}, \mathcal{L}), \mathcal{L}') = rev(\mathcal{P}, \mathcal{L} \cup \mathcal{L}')$ and $(\mathcal{P} \cup \mathcal{E}) \cup \mathcal{E}' = \mathcal{P} \cup (\mathcal{E} \cup \mathcal{E}')$, $\longrightarrow_r$ and $\longrightarrow_a$ need to be applied at most once. If the condition of a conditional is *unknown*, then a final state can always be reached by applying the revision reduction to all *unknown* conditions (Equation (9.3) on page 167 in the *Firing Squad* example), but this is usually not a derivation where minimal revision is applied (Equation (9.4) on page 167 , where only $\neg s$ is revised and Equation (9.5) on page 168, where only *ra* is revised).

In the context of the Mental Model Theory, our assumption of *minimal revision followed by abduction* seems to go along with the idea in [Johnson-Laird, Khemlani, and Goodwin, 2015]: The authors state that when humans generate a mental model and need to explain some inconsistencies, they rate explanations as more probable than minimal changes.

9.5. Minimal Revision followed by Abduction

Our belief expressed in the last section, namely that humans search for derivations of the form $\xrightarrow{n}_c \cdot \xrightarrow{0/1}_r \cdot \xrightarrow{0/1}_a \cdot \xrightarrow{0/1}_s$, allows us to redefine the evaluation of a conditional:

1. If $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \top$ then evaluate $cond(\mathcal{C}, \mathcal{D})$ with respect to $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{D})$.
2. If $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \bot$ then evaluate $cond(\mathcal{C}, \mathcal{D})$ with respect to $\mathsf{lm}\,\mathsf{wc}\,rev(\mathcal{P}, \mathcal{L})$ where $\mathcal{L} = \{L \in \mathcal{C} \mid \mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(L) = \bot\}$.
3. If $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \mathrm{U}$ then evaluate $cond(\mathcal{C}, \mathcal{D})$ with respect to $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}'$ where
   - $\mathcal{P}' = rev(\mathcal{P}, \mathcal{L}) \cup \mathcal{E}$,
   - $\mathcal{L}$ is the smallest (possibly empty) subset of $\mathcal{C}$ and $\mathcal{E} \subseteq \mathcal{A}_{rev(\mathcal{P},\mathcal{L})}$ is a minimal explanation for $\mathcal{C} \setminus \mathcal{L}$ such that $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}'(\mathcal{C}) = \top$.

If the condition $\mathcal{C}$ of a conditional is *true*, then the conditional is an indicative one and is evaluated as implication under Łukasiewicz semantics. If $\mathcal{C}$ is *false*, then the conditional is a counterfactual one and revision is applied in order to reverse the truth value of those literals, which are mapped to *false*. By Proposition 9.1.1, this case is non-monotonic. If

$\mathcal{C}$ is *unknown*, then we propose to split $\mathcal{C}$ into two disjoint subsets $\mathcal{L}$ and $\mathcal{C} \setminus \mathcal{L}$, where the former is treated by revision and the latter by abduction. In case $\mathcal{C}$ contains some literals, which are *true* and some, which are *unknown* under $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}$, then the former will be part of $\mathcal{C} \setminus \mathcal{L}$ because the empty explanation explains them. Furthermore, as all revised or explained literals were assigned to *unknown*, by Proposition 9.1.2 this case is monotonic. As we assume $\mathcal{L}$ to be minimal, this approach is called *minimal revision followed by abduction* (MRFA).

Firing Squad Reconsidering the *Firing Squad* example we find that Equation (9.6), which corresponds to the derivation shown in gray in Table 9.1 is the only evaluation with respect to MRFA.

Forest Fire As another example, consider the *Forest Fire* example taken from Byrne [2007]: *Lightning (l) causes a forest fire (f) if nothing abnormal is taking place (ab_1), lightning happened, the absence of dry leaves (d) is an abnormality, and dry leaves are present.* We obtain $\mathcal{P}_9$:

$$\begin{array}{rcl} f & \leftarrow & l \wedge \neg ab_1. \\ l & \leftarrow & \top. \\ ab_1 & \leftarrow & \neg d. \\ d & \leftarrow & \top. \end{array}$$

and $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_9$ is

$$\langle \{d, l, f\}, \{ab_1\} \rangle.$$

Now consider the conditional

> *If there had not been so many dry leaves on the forest floor ($\neg d$), then the forest fire would not have occurred ($\neg f$).*

As $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_9(\neg d) = \bot$, the conditional is a counterfactual and we consider $rev(\mathcal{P}_9, \neg d)$. As $\mathsf{lm}\,\mathsf{wc}\,rev(\mathcal{P}_9, \neg d) = \langle \{l, ab_1\}, \{d, f\} \rangle$ maps $\neg f$ to *true*, the conditional is *true*. Suppose we additionally learn that *arson may cause a forest fire.* The corresponding program, $\mathcal{P}_{10}$ is defined as follows:

$$\mathcal{P}_9 \cup \{f \leftarrow a \wedge \neg ab_2,\ ab_2 \leftarrow \bot\}.$$

We find that $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{10}$ is

$$\langle \{d, l, f\}, \{ab_1, ab_2\} \rangle$$

and $\mathsf{lm}\,\mathsf{wc}\,rev(\mathcal{P}_{10}, \neg d)$ is

$$\langle \{l, ab_1\}, \{d, ab_2\} \rangle,$$

where $rev(\mathcal{P}_{10}, \neg d)$ consists of the following clauses:

$$\begin{array}{rcl} f & \leftarrow & l \wedge \neg ab_1. \\ l & \leftarrow & \top. \\ ab_1 & \leftarrow & \neg d. \\ f & \leftarrow & a \wedge \neg ab_2. \\ ab_2 & \leftarrow & \bot. \\ d & \leftarrow & \bot. \end{array}$$

Under $\mathsf{lm}\,\mathsf{wc}\; rev(\mathcal{P}_{10}, \neg d)$, f is *unknown* and, consequently, the conditional is *unknown*.

9.6. Relevance

We will discuss relevance in the context of the evaluation of conditionals and present two notions of relevance, both indirectly inspired by Anderson and Belnap [1975], Anderson, Belnap, and Dunn [1992] applied to our framework.[7] It seems to be natural and is widely assumed in the literature, that humans evaluate the truth of the consequence based on whether it is supported on something in common with the support of the condition, i.e. their supports must not be completely disjoint or irrelevant to one another [Mares, 2004].

So far, a conditional $cond(\mathcal{C}, \mathcal{D})$ is mapped to *unknown* if $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \top$ and $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{D}) =$ U. We have seen several examples like the first derivation in the *Shooting of Kennedy* example or the last derivation discussed in the *Forest Fire* example. These conclusions are due to the fact that we are using the Weak Completion Semantics, which adopts an open-world assumption and assigns *unknown* to undefined atoms. If we reason with respect to the well-founded model or the stable model instead, *false* would have been assigned to all undefined atoms, because a closed-world assumption has been adopted. This would have led to a positive evaluation in the last derivation of the *Forest Fire* example: Because of the absence of dry leaves, lightning could not have caused the forest fire and, since arson being assigned to *false* by default, arson could not have caused the forest fire either. Also in the *Shooting of Kennedy* example, by assigning *false* to *ses* by default, the conditional would be *false*, which is a rather unexpected result. It seems that by the closed-world assumption, the particular reason why the conditional is evaluated *true*, is lost. Under the Weak Completion Semantics, it seems natural to explicitly state the context in which the conditional is true:

> *'If there had not been so many dry leaves on the forest floor,*
> *then the forest fire would not have occurred' is true*
> *in the context of arson being false.*

[7] For yet another approach, which also considers unknown values and does not use probabilities, consider [Pereira and Saptawijaya, 2016b,a], who apply their system to human moral reasoning.

We can construct the context under which the conditional might be true by allowing abduction with respect to consequences if the condition of a conditional is mapped to *true*. Consider again $rev(\mathcal{P}_{10}, \neg d)$ (page 173): If we allow to abduce an explanation for $\neg f$, then $\{a \leftarrow \bot\}$ is its only minimal explanation in the abductive framework $\langle rev(\mathcal{P}_{10}, \neg d), \{a \leftarrow \top, a \leftarrow \bot\}, \emptyset, \models_{wcs}\rangle$. Hence, the conditional $cond(\neg d, \neg f)$ is *true* in the context of a being *false*.[8]

In the case of $\mathcal{P}_6$ (page 166) we find that $\{ses \leftarrow \top\}$ is a minimal explanation for *ses* in the abductive framework $\langle \mathcal{P}_6, \{ses \leftarrow \top, ses \leftarrow \bot\}, \emptyset, \models_{wcs}\rangle$. Hence, the conditional $cond(\neg os, ses)$ is *true* in the context of *ses* being *true*.

In fact, any conditional whose conditions are *true* and whose consequences are *unknown* can be mapped to *true* in the context of its consequences being *true*. As an example consider the conditional

> *If Oswald had not shot Kennedy,*
> *then lightning would have occurred.*

and suppose that this conditional, $cond(\neg os, l)$, is evaluated with respect to $\mathcal{P}_5$ (page 165). We have to revise $\mathcal{P}_5$ with respect to $\neg os$ and obtain $\mathcal{P}_6$. After that we can explain l by $\{l \leftarrow \top\}$. For the evaluation of conditionals this does not seem to be very helpful as it does not include any relevant information provided by the conditional itself. This brings us to two new aspects that need to be taken into account: Firstly, we need to restrict the set of abducibles such that the consequence cannot abduce itself and, secondly, we need to check whether the condition of a true conditional is relevant to its consequence.

9.6.1. Weak Relevance

One way to define relevance is exclusively through dependencies as follows: atom B is *relevant to* atom A iff A depends on B. Recall that the *depends on* relation is the transitive closure of the following relation: Atom A *depends on* atom B if $\mathcal{P}$ contains a clause of the form $A \leftarrow body$ and B occurs (positively or negatively) in *body*. Let $\mathcal{P}_{11}$ consist of the following two clauses:

$$\begin{array}{lcl} a & \leftarrow & b. \\ c & \leftarrow & b. \end{array}$$

Is c relevant to a? As a does not depend on c, the answer is *no*. Assume that we would like to evaluate $cond(c, a)$ with respect to $\mathcal{P}_{11}$ using MRFA: c will be *true* by abducing the explanation $\mathcal{E}_4 = \{b \leftarrow \top\}$ and, consequently, a will be true as well. Thus, c indirectly determines the truth value of a with respect to $\mathcal{P}_{11} \cup \mathcal{E}_4$. Yet, a still does not depend on c in $\mathcal{P}_{11} \cup \mathcal{E}_4$. Therefore, this notion of relevance does not seem to be adequate.

[8] Recall that d, f and a stand for dry leaves, forest fire and arson, respectively.

Example 9.5. Applied to the program $\mathcal{P}_{11} \cup \mathcal{E}_4$ and the conditional $cond(a, c)$, we find that

$$(\{a \leftarrow \top\} \cup \{b \leftarrow \top\}) \cap (\{c \leftarrow \top\} \cup \{b \leftarrow \top\}) = \{b \leftarrow \top\} \neq \emptyset$$

and therefore, c and a are weakly relevant to one another. Consider yet another example with $cond(b, a)$ where $\mathcal{P}_{12}$ consists of the following clauses:

$$\begin{array}{lcl} a & \leftarrow & b. \\ b & \leftarrow & \top. \\ a & \leftarrow & \top. \end{array}$$

Even though a depends on b, the truth of b has no influence on a: a is even *true* in $\mathsf{lm}\,\mathsf{wc}\; rev(\mathcal{P}_{12}, \neg b)$.
Our notion of weak relevance does not help here: a and b are weakly relevant to one another, because

$$(\{a \leftarrow \top\} \cup dep(a, \mathcal{P}_{12})) \cap (\{b \leftarrow \top\} \cup dep(b, \mathcal{P}_{12})) = \{b \leftarrow \top\} \neq \emptyset.$$

Let $\mathcal{L}$ be a set of literals: $dep(\mathcal{L}, \mathcal{P})$ is the set of facts and assumptions in $\mathcal{P}$, on which $\mathcal{L}$ depends on:

$$\begin{array}{lcl} dep(\mathcal{L}, \mathcal{P}) & = & \{A' \leftarrow body \in \mathcal{P} \mid body \in \{\top, \bot\} \text{ and} \\ & & \text{there exists } A \in \mathcal{L} \text{ or } \neg A \in \mathcal{L} \text{ such that } A \text{ depends on } A'\}. \end{array}$$

Consider the following weak notion of relevance:
$\mathcal{C}$ and $\mathcal{D}$ are *weakly relevant to one another with respect to* $\mathcal{P}$ iff

$$\begin{array}{l} (\{A \leftarrow \top \mid A \in \mathcal{C}\} \cup \{A \leftarrow \bot \mid \neg A \in \mathcal{C}\} \cup dep(\mathcal{C}, \mathcal{P})) \\ \cap\, (\{A \leftarrow \top \mid A \in \mathcal{D}\} \cup \{A \leftarrow \bot \mid \neg A \in \mathcal{D}\} \cup dep(\mathcal{D}, \mathcal{P})) \neq \emptyset. \end{array}$$

The notion of weak relevance is clarified by Example 9.5.

9.6.2. Strong Relevance

The idea behind strong relevance is to check whether the truth value of $\mathcal{D}$ changes if $\mathcal{D}$ does not depend on $\mathcal{C}$ anymore. Consider yet another definition of relevance:

$\mathcal{C}$ is *strongly relevant to* $\mathcal{D}$ *with respect to* $\mathcal{P}$ iff

$$\begin{array}{lll} \mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{D}) = \top & \text{and} & \mathsf{lm}\,\mathsf{wc}\,\mathcal{P}'(\mathcal{D}) \neq \top \\ & \text{where} & \mathcal{P}' = \mathcal{P} \setminus (\mathsf{def}(\mathcal{C}, \mathcal{P}) \cup dep(\mathcal{C}, \mathcal{P})). \end{array}$$

Different than the defition for weak relevance, the definition for strong relevance is not symmetrical. Example 9.6 clarifies this notion of relevance. The notion of strong relev-

Example 9.6. Consider again $\mathcal{P}_{12}$ (page 175): In order to verify whether b is strongly relevant to a with respect to $\mathcal{P}_{12}$, we first need to check that both are true in $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}$, which is indeed the case. After that, note that $\mathcal{P}_{13}$ is

$$\mathcal{P}_{12} \setminus \{b \leftarrow \top\} = \{a \leftarrow b, a \leftarrow \top\}$$

where

$$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{13}(a) = \top.$$

Accordingly, b is not strongly relevant to a with respect to $\mathcal{P}_{12}$. Consider $\mathcal{P}_{11}$ (page 174) again together with $cond(c, a)$ and

$$\mathcal{E}_4 = \{b \leftarrow \top\}.$$

As a and b are true in $\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P}_{11} \cup \mathcal{E}_4)$ and, additionally, a is not true under $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{14}$, where $\mathcal{P}_{14}$ is

$$(\mathcal{P}_{11} \cup \mathcal{E}_4) \setminus \{\mathsf{def}(c, \mathcal{P}) \cup dep(c, \mathcal{P})\} = \{a \leftarrow c\}.$$

c is strongly relevant to a.

ance captures best our intention, and therefore we will assume it in the following. Step 1 in MRFA is modified in two ways: First, by checking whether for the *true* conditionals, the condition is relevant to the consequence; and second by allowing abduction, in case the consequence is *unknown*:

1. a) If $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{D}) = \top$ and $\mathcal{C}$ is strongly relevant to $\mathcal{D}$ then $cond(\mathcal{C}, \mathcal{D})$ is *true*.

   b) If $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \top$ and $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{D}) = \bot$ then $cond(\mathcal{C}, \mathcal{D})$ is *false*.

   c) If $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{C}) = \top$ and $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}(\mathcal{D}) = \mathrm{U}$ then

      i. if $\mathcal{E} \subset (\mathcal{A}_\mathcal{P} \setminus (\{A \leftarrow \top \mid A \in \mathcal{D}\} \cup \{A \leftarrow \bot \mid \neg A \in \mathcal{D}\}))$ is a minimal explanation for $\mathcal{O} \subseteq \mathcal{D}$, $\mathsf{lm}\,\mathsf{wc}\,(\mathcal{P} \cup \mathcal{E})(\mathcal{D}) = \top$ and $\mathcal{C}$ is strongly relevant to $\mathcal{D}$ with respect to $\mathcal{P} \cup \mathcal{E}$ then $cond(\mathcal{C}, \mathcal{D})$ is *true* in the context of $\mathcal{E}$.

      ii. else $cond(\mathcal{C}, \mathcal{D})$ is *unknown*.

If none of the cases applies, because $\mathcal{C}$ fails to be relevant to $\mathcal{D}$, the condition of the conditional is not relevant to the consequence, i.e. the conditional is meaningless. Example 9.7 discusses an extension of the Kennedy example from Section 9.3.2

Example 9.7. $\mathcal{P}_{15}$ consists of the following clauses:

$$\begin{array}{rcl} k & \leftarrow & os \wedge \neg ab_1. \\ k & \leftarrow & ses \wedge \neg ab_2. \\ os & \leftarrow & \top. \\ ab_1 & \leftarrow & \bot. \\ ab_2 & \leftarrow & \bot. \\ k & \leftarrow & \top. \end{array}$$

Almost all clauses are the same as in $\mathcal{P}_5$ (page 165), except of the last one which additionally states that, independently of whether Oswald shot Kennedy, Kennedy is dead. Consider the conditional

If Oswald shot, then Kennedy is dead.

represented as $cond(os, k)$. $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{15} = \langle\{k, os\}, \{ab_1, ab_2\}\rangle$, where k and os are both *true*. os is not strongly relevant to k because both are *true* in $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{15}$ and k is still *true* in $\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}_{16}$:

$$\langle\{k\}, \{ab_1, ab_2\}\rangle$$

where $\mathcal{P}_{16}$ is

$$\mathcal{P}_{15} \setminus (\mathsf{def}(os, \mathcal{P}_{15}) \cup dep(os, \mathcal{P}_{15})) \quad = \quad \mathcal{P}_{15} \setminus \{os \leftarrow \top\}.$$

Nevertheless, os and k are weakly relevant to one another, because

$$\begin{array}{l} (\{os \leftarrow \top\} \cup dep(os, \mathcal{P})) \cap (\{k \leftarrow \top\} \cup dep(k, \mathcal{P})) \\ = \ (\{os \leftarrow \top\}) \cap (\{k \leftarrow \top\} \cup \{os \leftarrow \top\}) \\ = \ \{os \leftarrow \top\}. \end{array}$$

In this example, os is not essential for k, and not strongly relevant to it; but it is conceivable that os influences (the truth value of) k by revision or abduction, and so both are weakly relevant to one another.

9.6.3. Relevance Property in Logic Programming

Relevance is defined for the general case by Pinto and Pereira [2011] and adopted from Dix [1995]. The relevant part of a program $\mathcal{P}$ for an atom A, is defined as follows:

$$\mathsf{rel}(A, \mathcal{P}) \quad = \quad \mathsf{def}(A, \mathcal{P}) \cup \{A' \leftarrow \mathit{body} \in \mathcal{P} \mid A \text{ depends on } A'\}.$$

A semantics is said to be *relevant* or to *enjoy the relevance property* iff for every program $\mathcal{P}$ and for all $A \in \mathsf{atoms}(\mathcal{P})$ the following holds:

$$\begin{array}{lllll} \mathcal{P} \models & A & \text{iff} & \mathsf{rel}(A, \mathcal{P}) \models & A \quad \text{and} \\ \mathcal{P} \models & \neg A & \text{iff} & \mathsf{rel}(A, \mathcal{P}) \models & \neg A. \end{array}$$

To clarify this notion consider program $\mathcal{P}_{17}$ consisting of the following clauses:

$$\begin{array}{lll} a & \leftarrow & \neg b. \\ b & \leftarrow & \neg a. \end{array}$$

Under the Weak Completion Semantics and the Well-founded Semantics, the least model of the weak completion of $\mathcal{P}_{17}$ is $\langle \emptyset, \emptyset \rangle$ and under the Stable Model Semantics, we have two partial stable models, namely $\langle \{a\}, \{b\} \rangle$ and $\langle \{b\}, \{a\} \rangle$.

Assume that $\mathcal{P}_{18}$ is a program that consists only of clauses which are not relevant to a and b. Can we guarantee that in $\mathcal{P}_{17} \cup \mathcal{P}_{18}$, a and b will stay unknown under the Weak Completion Semantics and the Well-founded Semantics and the stable models will stay the same? If this is the case, then, these semantics enjoy the relevance property [Pinto and Pereira, 2011]. Assume that $\mathcal{P}_{18}$ consists of exactly one clause:

$$\begin{array}{lll} a & \leftarrow & c. \\ c & \leftarrow & \neg c. \end{array}$$

As now a is involved in an odd negative cycle, a stays unknown under the Partial Stable Model Semantics, and therefore, the only (partial) stable model of $\mathcal{P}_{17} \cup \mathcal{P}_{18}$ is $\langle \emptyset, \{b\} \rangle$. This is a counterexample, which shows that the Stable Model Semantics does not enjoy the relevance property. On the other hand, the truth values of a and b stay the same under the Well-founded Semantics and the Weak Completion Semantics: In both cases, the model $\mathcal{P}_{17} \cup \mathcal{P}_{18}$ is $\langle \emptyset, \emptyset \rangle$. It is straightforward to see from the definition of the $\Phi_{\mathcal{P}}$ operator that the Weak Completion Semantics enjoys the relevance property. In [Dix, 1995], it has been shown that the Well-founded Semantics enjoys the relevance property as well.

9.7. Conclusion

This chapter presents a novel approach to conditional evaluation: ARSC is an abstract reduction system that is flexible enough to model various evaluation steps for conditionals, possibly leading to different outcomes. We conjecture that humans reason with a third truth value and prefer abduction to revision, and formalize this hypothesis in MRFA.

We additionally assume that humans take relevance into account where we discuss several concepts of relevance and show why strong relevance fits best in our system. As discussed in [Skovgaard-Olsen, Singmann, and Klauer, 2016], it seems that usually psychological studies don't give a lot of attention to the concept of relevance, even though their own experimental results show that relevance affects the participants' results. On the contrary, as stated in [Skovgaard-Olsen, Singmann, and Klauer, 2016], the Mental Model Theory and probabilistic theories deny that relevance plays a role when humans reason with (indicative) conditionals. As mentioned by Skovgaard-Olsen, Singmann, and Klauer [2016], Spohn [2013] proposes to distinguish between positive relevance, irrelevance and negative relevance, in order to get a better picture of whether relevance plays a role in human reasoning. Here, we have not paid attention to this distinction, but these psychological observations should be taken into account for future investigations.

Finally, as there is not enough experimental data in the literature, in Section 9.4, we summarize central questions that need to be investigated through psychological experiments. The results will give us insights on whether our approach or a variation thereof can adequately model human reasoning.

10. Correspondence to a Related System

In this chapter, we show that MRFA, an evaluation system for conditionals presented in Chapter 9 is more general then another logic programming approach for evaluating conditionals that has been proposed by Schulz [2014]. We first reconsider the $\Phi_{\mathcal{P}}$ operator and establish some of its properties in Section 10.1. After that, in Section 10.2 we present Schulz' Approach. Finally, the main result of this Chapter, the formal correspondence of MRFA to Schulz's Approach, is presented in Section 10.3.[1]

10.1. Semantic Operator Revisited

Before looking into conditionals, we need to reconsider the $\Phi_{\mathcal{P}}$ operator of Section 2.3 and establish some of its properties. Given a program $\mathcal{P}$, the least fixed point of $\Phi_{\mathcal{P}}$ can be computed by iterating the operator starting with the empty interpretation: $\Phi_{\mathcal{P}}{\uparrow}0 = \langle\emptyset,\emptyset\rangle$, $\Phi_{\mathcal{P}}{\uparrow}(n+1) = \Phi_{\mathcal{P}}(\Phi_{\mathcal{P}}{\uparrow}n)$ for all $n \in \mathbb{N}$.

Proposition 10.1. *The following holds:*

1. *$\Phi_{\mathcal{P}}$ is monotonic, i.e. $I \subseteq J$ implies $\Phi_{\mathcal{P}}(I) \subseteq \Phi_{\mathcal{P}}(J)$.*
2. *For all $n \geq 0$ we find $\Phi_{\mathcal{P}}(\Phi_{\mathcal{P}}{\uparrow}n) \supseteq \Phi_{\mathcal{P}}{\uparrow}n$.*
3. *For all $n \geq 0$ we find $\Phi_{\mathcal{P}}{\uparrow}(n+1) = \Phi_{\mathcal{P}}{\uparrow}n \cup \langle J^{\top}, J^{\bot}\rangle$, where*

$$J^{\top} = \{A \mid \Phi_{\mathcal{P}}{\uparrow}n(A) = \mathrm{U}, A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ and } \Phi_{\mathcal{P}}{\uparrow}n(\mathit{body}) = \top\},$$
$$J^{\bot} = \{A \mid \Phi_{\mathcal{P}}{\uparrow}n(A) = \mathrm{U}, \mathsf{def}(A,\mathcal{P}) \neq \emptyset \text{ and for all } A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ we find that } \Phi_{\mathcal{P}}{\uparrow}n(\mathit{body}) = \bot\}.$$

Proof.

1. See Proposition 3.21 in [Kencana Ramli, 2009].
2. The proof is by induction on n: The case $n = 0$ holds because

$$\Phi_{\mathcal{P}}(\Phi_{\mathcal{P}}{\uparrow}0) = \Phi_{\mathcal{P}}(\langle\emptyset,\emptyset\rangle) \supseteq \langle\emptyset,\emptyset\rangle.$$

[1] The results of this chapter are published in [Dietz and Hölldobler, 2015].

From the induction hypothesis $\Phi_{\mathcal{P}}(\Phi_{\mathcal{P}} \uparrow n) \supseteq \Phi_{\mathcal{P}} \uparrow n$ we conclude by the monotonicity of $\Phi_{\mathcal{P}}$ that $\Phi_{\mathcal{P}}(\Phi_{\mathcal{P}}(\Phi_{\mathcal{P}} \uparrow n)) \supseteq \Phi_{\mathcal{P}}(\Phi_{\mathcal{P}} \uparrow n)$. Then, we know that $\Phi_{\mathcal{P}}(\Phi_{\mathcal{P}} \uparrow (n+1)) \supseteq \Phi_{\mathcal{P}} \uparrow (n+1)$.

3.

$$\begin{aligned} \Phi_{\mathcal{P}} \uparrow (n+1) &= \Phi_{\mathcal{P}} \uparrow n \cup \Phi_{\mathcal{P}} \uparrow (n+1) && \text{By 2.} \\ &= \Phi_{\mathcal{P}} \uparrow n \cup (\Phi_{\mathcal{P}} \uparrow (n+1) \setminus \Phi_{\mathcal{P}} \uparrow n) \\ &= \Phi_{\mathcal{P}} \uparrow n \cup \langle J^{\top}, J^{\perp} \rangle && \text{By } \Phi_{\mathcal{P}} \uparrow n(A) = U \\ & && \text{in } J^{\top} \text{ and } J^{\perp}. \end{aligned}$$ □

10.2. Schulz's Approach

Schulz [2014] presents another computational logic approach, where the $\Phi_{\mathcal{P}}$ operator is modified such that it allows to evaluate conditionals. In this section, let $\mathcal{L}$ be a finite and consistent set of ground literals. Given a set $\mathcal{L}$, the interpretation $\langle L^{\top}, L^{\perp} \rangle$ is defined as $L^{\top} = \{A \mid A \in \mathcal{L}\}$ and $L^{\perp} = \{A \mid \neg A \in \mathcal{L}\}$, where A denotes a ground atom.

Let $I = \langle I^{\top}, I^{\perp} \rangle$ be an interpretation. Schulz defines

$$\tau_{\mathcal{P},\mathcal{L}}(\langle I^{\top}, I^{\perp} \rangle) = \langle I^{\top}, I^{\perp} \rangle \cup \langle J^{\top}, J^{\perp} \rangle,$$

where

$$\begin{aligned} J^{\top} &= \{A \mid \ I(A) = \mathrm{U},\ A \leftarrow \mathit{body} \in \mathsf{def}(A, \mathcal{P}) \text{ and } I(\mathit{body}) = \top\}, \\ J^{\perp} &= \{A \mid \ I(A) = \mathrm{U},\ \mathsf{def}(A, \mathcal{P}) \neq \emptyset \text{ and} \\ &\qquad \text{for all } A \leftarrow \mathit{body} \in \mathsf{def}(A, \mathcal{P}) \text{ we find that } I(\mathit{body}) = \perp\}. \end{aligned}$$

In contrast to the $\Phi_{\mathcal{P}}$ operator, which is iterated starting with the empty interpretation, the $\tau_{\mathcal{P}}$ operator is iterated as follows:

$$\tau_{\mathcal{P}} \uparrow 0 = \langle L^{\top}, L^{\perp} \rangle \qquad \text{and} \qquad \tau_{\mathcal{P}} \uparrow (n+1) = \tau_{\mathcal{P}}(\tau_{\mathcal{P}} \uparrow\ n).$$

As shown by Schulz, the $\tau_{\mathcal{P}}$ operator admits a least fixed point which shall be denoted by $\mathsf{lfp}\, \tau_{\mathcal{P},\mathcal{L}}$. $\mathsf{lfp}\, \tau_{\mathcal{P},\mathcal{L}}$ can be computed by iterating the operator starting with the interpretation: $\tau_{\mathcal{P}} \uparrow 0 = \langle L^{\top}, L^{\perp} \rangle$, $\tau_{\mathcal{P}} \uparrow (n+1) = \tau_{\mathcal{P}}(\tau_{\mathcal{P}} \uparrow n)$ for all $n \in \mathbb{N}$ Moreover, in [Schulz, 2014] reasoning is performed with respect to this fixed point, i.e. $\mathcal{P}, \mathcal{L} \models_s F$ iff $\mathsf{lfp}\, \tau_{\mathcal{P},\mathcal{L}}(F) = \top$. Note that by the first condition, $I(A) = U$, in both $J^{\top}$ and $J^{\perp}$, monotonicity is guaranteed for $\tau_{\mathcal{P},\mathcal{L}}$.

10.3. Correspondence

Before we show the correspondence between the approach by Schulz and our approach, let us first identify some general properties of the operators $\Phi_{\mathcal{P}}$ and $\tau_{\mathcal{P}}$.

Proposition 10.2. $\mathsf{lfp}\,\Phi_{\mathcal{P}}$ *and* $\mathsf{lfp}\,\tau_{\mathcal{P},\mathcal{L}}$ *exist.*

The existence of $\mathsf{lfp}\,\Phi_{\mathcal{P}}$ and $\mathsf{lfp}\,\tau_{\mathcal{P},\mathcal{L}}$ was established in [Hölldobler and Kencana Ramli, 2009b] and in [Schulz, 2014], respectively. Given that $\tau_{\mathcal{P},\mathcal{L}}$ is monotonic and $\tau_{\mathcal{P},\mathcal{L}} \uparrow 0$ starts with $\langle L^{\top}, L^{\bot}\rangle$, the following proposition follows immediately.

Proposition 10.3. *For all* $L \in \mathcal{L}$ *we find* $\mathcal{P}, \mathcal{L} \models_s L$.

Proposition 10.3 for $\tau_{\mathcal{P},\mathcal{L}}$ corresponds to Proposition 9.1(3) for $\Phi_{rev(\mathcal{P},\mathcal{L})}$.

Theorem 10.4. $\mathsf{lfp}\,\Phi_{rev(\mathcal{P},\mathcal{L})} = \mathsf{lfp}\,\tau_{\mathcal{P},\mathcal{L}}$.

We show Theorem 10.4 by showing intermediate steps first.

Lemma 10.5. *For all* $n \in \mathbb{N}$*, we find*

$$\Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow n \subseteq \tau_{\mathcal{P},\mathcal{L}}\uparrow n \subseteq \Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow (n+1).$$

Proof.
To simplify the presentation, we will omit the indices of the operators $\tau_{\mathcal{P}}$ and $\Phi_{rev(\mathcal{P},\mathcal{L})}$ in this proof. The proof is by induction on n. In case $n = 0$ we find

$$\Phi\uparrow 0 = \langle \emptyset, \emptyset\rangle \;\subseteq\; \langle L^{\top}, L^{\bot}\rangle = \tau\uparrow 0 \;\subseteq\; \langle I^{\top}, I^{\bot}\rangle = \Phi\uparrow 1,$$

where

$$\begin{array}{lcll} I^{\top} & = & \{A \mid A \leftarrow \top \in rev(\mathcal{P},\mathcal{L})\} & \supseteq\; L^{\top}, \\ I^{\bot} & = & \{A \mid \mathsf{def}(A, rev(\mathcal{P},\mathcal{L})) = \{A \leftarrow \bot\}\} & \supseteq\; L^{\bot}. \end{array}$$

As induction hypothesis, we assume that the result holds for n, i.e.

$$\Phi\uparrow n \;\subseteq\; \tau\uparrow n \;\subseteq\; \Phi\uparrow (n+1). \tag{10.1}$$

In the induction step, we need to show that the result holds for $n+1$. We start by showing that

$$\Phi\uparrow (n+1) \;\subseteq\; \tau\uparrow (n+1). \tag{10.2}$$

By Proposition 10.1(3.) and the definition of $\tau_{\mathcal{P}}$, (10.2) is equivalent to

$$\Phi\uparrow n \cup \langle I^{\top}, I^{\bot}\rangle \;\subseteq\; \tau\uparrow n \cup \langle J^{\top}, J^{\bot}\rangle,$$

where

$$\begin{array}{lll} I^{\top} & = \{A \mid & \Phi\uparrow n(A) = \mathrm{U} \text{ and } A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ and } \Phi\uparrow n(\mathit{body}) = \top\}, \\ I^{\bot} & = \{A \mid & \Phi\uparrow n(A) = \mathrm{U} \text{ and } \mathsf{def}(A,\mathcal{P}) \neq \emptyset \text{ and} \\ & & \text{for all } A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ we find that } \Phi\uparrow n(\mathit{body}) = \bot\}, \\ J^{\top} & = \{A \mid & \tau\uparrow n(A) = \mathrm{U} \text{ and } A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ and } \tau\uparrow n(\mathit{body}) = \top\}, \\ J^{\bot} & = \{A \mid & \tau\uparrow n(A) = \mathrm{U} \text{ and } \mathsf{def}(A,\mathcal{P}) \neq \emptyset \text{ and} \\ & & \text{for all } A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ we find that } \tau\uparrow n(\mathit{body}) = \bot\}. \end{array}$$

From the induction hypothesis (10.1), we conclude that

$$\Phi{\uparrow}n \;\subseteq\; \tau{\uparrow}n \cup \langle J^{\top}, J^{\bot}\rangle. \tag{10.3}$$

Now suppose that $A \in I^{\top}$. Then, $\Phi{\uparrow}n(A) = \mathrm{U}$ and we distinguish between two cases:

1. If $\tau{\uparrow}n(A) = \mathrm{U}$, then because $A \in I^{\top}$ and by the induction hypothesis (10.1), $\Phi{\uparrow}n \;\subseteq\; \tau{\uparrow}n$, and, hence, $A \in J^{\top}$.
2. If $\tau{\uparrow}n(A) \neq \mathrm{U}$, then A must already been assigned to either *true* or *false* under $\tau{\uparrow}n$. As $A \in I^{\top}$ and by (10.1), $\tau{\uparrow}n \subseteq \Phi{\uparrow}(n+1)$, and, hence, $\tau{\uparrow}n(A) = \top$.

Likewise, we find for $A \in I^{\bot}$ that either $A \in J^{\bot}$ or $\tau{\uparrow}n(A) = \bot$. Therefore,

$$\langle I^{\top}, I^{\bot}\rangle \;\subseteq\; \tau{\uparrow}n \cup \langle J^{\top}, J^{\bot}\rangle \tag{10.4}$$

and (10.2) follows immediately from (10.3) and (10.4).

We turn to the proof of

$$\tau{\uparrow}(n+1) \;\subseteq\; \Phi{\uparrow}(n+2). \tag{10.5}$$

By the definition for $\tau_{\mathcal{P}}$ and Proposition 10.1(3.), this corresponds to

$$\tau{\uparrow}n \cup \langle J^{\top}, J^{\bot}\rangle \;\subseteq\; \Phi{\uparrow}(n+1) \cup \langle I^{\top}, I^{\bot}\rangle,$$

where

$$\begin{aligned}
J^{\top} &= \{A \mid \tau{\uparrow}n(A) = \mathrm{U} \text{ and } A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ and } \tau{\uparrow}n(\mathit{body}) = \top\},\\
J^{\bot} &= \{A \mid \tau{\uparrow}n(A) = \mathrm{U} \text{ and } \mathsf{def}(A,\mathcal{P}) \neq \emptyset \text{ and}\\
&\qquad \text{for all } A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ we find that } \tau{\uparrow}n(\mathit{body}) = \bot\},\\
I^{\top} &= \{A \mid \Phi{\uparrow}(n+1)(A) = \mathrm{U} \text{ and}\\
&\qquad A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ and } \Phi{\uparrow}(n+1)(\mathit{body}) = \top\},\\
I^{\bot} &= \{A \mid \Phi{\uparrow}(n+1)(A) = \mathrm{U} \text{ and } \mathsf{def}(A,\mathcal{P}) \neq \emptyset \text{ and}\\
&\qquad \text{for all } A \leftarrow \mathit{body} \in \mathsf{def}(A,\mathcal{P}) \text{ we find that } \Phi{\uparrow}(n+1)(\mathit{body}) = \bot\}.
\end{aligned}$$

By the induction hypothesis (10.1), we find

$$\tau{\uparrow}n \;\subseteq\; \Phi{\uparrow}(n+1) \cup \langle I^{\top}, I^{\bot}\rangle. \tag{10.6}$$

Now suppose that $A \in J^{\top}$. Then, $\tau{\uparrow}n(A) = \mathrm{U}$ and we distinguish between two cases:

1. If $\Phi{\uparrow}(n+1)(A) = \mathrm{U}$, then $A \in I^{\top}$, because of the induction hypothesis (10.1).
2. If $\Phi{\uparrow}(n+1)(A) \neq \mathrm{U}$, then A is assigned to either *true* or *false* under $\Phi{\uparrow}(n+1)$. By (10.2), $\Phi{\uparrow}(n+1)(A) = \top$.

Likewise, we find for $A \in J^{\bot}$ that either $A \in I^{\bot}$ or $\Phi{\uparrow}(n+1)(A) = \bot$. Therefore,

$$\langle J^{\top}, J^{\bot}\rangle \;\subseteq\; \Phi{\uparrow}(n+1) \cup \langle I^{\top}, I^{\bot}\rangle \tag{10.7}$$

and (10.5) follows immediately from (10.6) and (10.7). □

We can now show the correspondence of the two operators.

Theorem 10.4. $\mathsf{lfp}\,\Phi_{rev(\mathcal{P},\mathcal{L})} = \mathsf{lfp}\,\tau_{\mathcal{P},\mathcal{L}}$.

Proof.
$\mathsf{lfp}\,\Phi_{rev(\mathcal{P},\mathcal{L})}$ is computed by iterating $\Phi_{rev(\mathcal{P},\mathcal{L})}$ starting with the empty interpretation, $\Phi_{\mathcal{P}}\uparrow 0 = \langle\emptyset,\emptyset\rangle$. According to Proposition 10.2, $\Phi_{rev(\mathcal{P},\mathcal{L})}$ has a fixed point, i.e. $\Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow n = \Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow(n+1)$ for some $n \in \mathbb{N}$. According to Lemma 10.5, $\Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow n = \tau_{\mathcal{P},\mathcal{L}}\uparrow n = \Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow(n+1)$. But then, as $\Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow n$ is the least fixed point of $\Phi_{rev(\mathcal{P},\mathcal{L})}$, it also holds that $\Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow(n+1) = \tau_{\mathcal{P},\mathcal{L}}\uparrow(n+1) = \Phi_{rev(\mathcal{P},\mathcal{L})}\uparrow(n+2)$. Accordingly, $\tau_{\mathcal{P},\mathcal{L}}\uparrow n = \tau_{\mathcal{P},\mathcal{L}}\uparrow(n+1)$, thus $\tau_{\mathcal{P},\mathcal{L}}\uparrow n$ is the least fixed point of $\tau_{\mathcal{P},\mathcal{L}}$. □

Glass of Wine Let us consider an example discussed by Schulz [2014]:[2]

> *If she drops the glass of wine (drop), then the glass of wine breaks (broken). She drops the glass of wine.*

This scenario can be represented by the following three clauses in program $\mathcal{P}_3$:

$$\begin{array}{rcl} \mathit{broken} & \leftarrow & \mathit{drop} \wedge \neg \mathit{ab}. \\ \mathit{ab} & \leftarrow & \bot. \\ \mathit{drop} & \leftarrow & \top. \end{array}$$

Now, consider $\mathcal{L} = \{\neg\mathit{broken}\}$. Then, $\mathit{rev}(\mathcal{P}_3,\mathcal{L})$ consists of

$$\begin{array}{rcl} \mathit{ab} & \leftarrow & \bot. \\ \mathit{drop} & \leftarrow & \top. \\ \mathit{broken} & \leftarrow & \bot. \end{array}$$

$L^{\top} = \emptyset,\ L^{\bot} = \{\mathit{broken}\}$, and the two fixed points are computed as follows:

	$\tau_{\mathcal{P}_3,\mathcal{L}}$	$\Phi_{rev(\mathcal{P}_3,\mathcal{L})}$,
$\uparrow 0$	$\langle\emptyset,\{\mathit{broken}\}\rangle$	$\langle\emptyset,\emptyset\rangle$
$\uparrow 1$	$\langle\{\mathit{drop}\},\{\mathit{ab},\mathit{broken}\}\rangle$	$\langle\{\mathit{drop}\},\{\mathit{ab},\mathit{broken}\}\rangle$

As expected, the least fixed points of $\tau_{\mathcal{P}_3,\mathcal{L}}$ and $\Phi_{rev(\mathcal{P}_3,\mathcal{L})}$ are identical.

[2] The following two sentences are adapted versions of the sentences in [Schulz, 2014], which originally were follows: *If you drop glass, it breaks. She dropped that wine glass.*

10.4. Conclusion

In this chapter, we formally show the correspondence to Schulz' approach and observe that we can handle more human reasoning tasks. Coming back to the examples discussed in Section 9.3.2 we observe that they can be modeled by Schulz' approach only if the appropriate initial set $\mathcal{L}$ is given. Schulz does not provide any means to obtain these sets. One should note that these sets are not simply the unknown conditions of the given conditionals. We compute the additional assignments by MRFA as explained in Section 9.5. In fact, we are unaware of any computational logic approach which can handle as many human reasoning episodes as our approach based on the Weak Completion Semantics. Yet, there are still many open and interesting questions, some of which will be mentioned in the sequel.

Part IV.

Conclusions

11. Open Questions and Outlook

Overall, the goal of developing a methodology that allows us to formalize episodes of human reasoning is far from being exhaustively explored. In this chapter we will discuss a few open questions.

11.1. Weak Completion Semantics Revisted

As discussed in Section 5.3, there are still open questions about the Weak Completion Semantics. First of all, Łukasiewicz semantics was chosen because it solved a technical bug in [Stenning and van Lambalgen, 2008] and nice properties such as the model intersection property. However, the same results would be yielded with the S-semantics. Is there any reason to prefer one three-valued semantics over the other? Why should we restrict ourselves to three-valued semantics? Might other more-than-three-valued logics not be suitable as well?

Further, can we really assume that people compute their models according to the $\Phi_{\mathcal{P}}$ operator? How does their reasoning differ in case they start with some background information? Can we simulate this aspect by starting the iteration of the $\Phi_{\mathcal{P}}$ operator with a non-empty interpretation?

We have introduced integrity constraints, however, we have not yet investigated them in the context of human reasoning. Luís Moniz Pereira[1] remarked that we might also think of testing the $\mathcal{IC}$ at each step of iteration. As the $\Phi_{\mathcal{P}}$ operator is monotonic, as soon as the body of the $\mathcal{IC}$ is true, there is no least model of the program that satisfies $\mathcal{IC}$.

Under the Weak Completion Semantics, positive information is preferred over unknown information and unknown information is preferred over negative information. This preference might not always be consistent with human reasoning. We could allow the expression of integrity constraints to $\bot \leftarrow q$. Any model of a program containing such an integrity constraint must map q to $\bot$. However, how can we search for a model that satisfies this integrity constraint? Will we have to define a new semantic operator or is there a way of testing the integrity constraint at each step of the iteration as discussed above?

[1] personal communication, February 10, 2016

11.2. Abduction

Abduction seems to be a powerful tool when modeling human reasoning. During formalization of all tasks, we assumed that explanations should be minimal and consequences should follow skeptically. Almost all tasks required skeptical abduction and the task formalized in Chapter 8 additionally requires explanations to be minimal. However, how likely is it that humans compute all minimal explanations first and then consider only the consequences that follow skeptically? It seems more convenient that some explanations are more likely to be considered than others, not depending on their minimality but depending on some other parameter, such as the context or the background information. Similarly, humans might apply skeptical abduction but instead reason based on whether a consequence is likely to follow from all possible explanations.

Contextual Reasoning

Chapter 4 takes the assumption that context plays a role while searching for explanations as starting point and shows that the Weak Completion Semantics cannot model the famous Tweety example adequately. As has already been observed by Reiter [1980], exception cases should be treated differently than usual cases: In case there is no reason to assume exception cases to be true, they should be false. We partially agree with this view, and further think that exception cases are actually ignored if there is no evidence for them to be true. We overcome these limitations by first introducing contextual programs, which allow us to syntactically specify contextual knowledge in the logic programs. Second, we formalize our intention within a contextual abductive reasoning approach and show how the previous limitations can be solved. It seems that there is a link to Reiter's [1980] default logic, however we have not shown a formal correspondence.

An open question, which we need to address to the cognitive scientists, is, whether the above assumptions, the way that humans are influenced by their background knowledge and whether they deal differently with usual cases than with exception cases, can be tested psychologically, and if so, whether the results of the experiments support these assumptions.

Complexity of Human Reasoning Tasks

The least model of the weak completion can be computed by the $\Phi_{\mathcal{P}}$ operator in polynomial time as has been shown in [Hölldobler, Philipp, and Wernhard, 2011], which is an advantageous property compared to other logic programming approaches, such as the Stable Model Semantics. Skeptical abduction on the other hand, has less desirable properties: Deciding whether a formula follows skeptically from an abductive framework is DP-complete, a complexity that is outside of NP [Hölldobler, Philipp, and

Wernhard, 2011]. Furthermore, deciding whether a contextual explanation is minimal lies in PSPACE [Dietz Saldanha, Hölldobler, and Philipp, 2017c].

These results are good indications to believe that humans are unlikely to reason in the same way as we apply skeptical abduction, in particular, they might not filter out all non-minimal explanations. Possibly, they generate only a few explanations and only consider them and their consequences partially. How they generate these few explanations might depend on their relevance in the context. Whether and how this mechanism can work out in detail, still needs to be investigated.

Neural Network Realization

As already mentioned in Section 5.3, Hölldobler and Kencana Ramli [2009a], showed that the computation of the least fixed point of the $\Phi_{\mathcal{P}}$ operator can be realized within a connectionist network, with the core-method [Bader, Hitzler, Hölldobler, and Witzel, 2007]. Furthermore, Dietz Saldanha, Hölldobler, Kencana Ramli, and Palacios Medinacelli [2017a] have shown a connectionist realization of skeptical abduction under the Weak Completion Semantics within the core-method. However, this approach is not restricted to minimal explanations. In [Palacios Medinacelli, 2016], a formal specification is provided that produces all possible explanations in a specific order such that minimal explanations can be detected and all non-minimal possible explanations can be discarded.

Summing up the above discussion, this specification does not seem to be the way humans search for explanations. As we already stated, humans might consider explanations which are more likely based on other parameters than the minimality characterization. One such other parameter is identified and proposed in [Dietz Saldanha, Hölldobler, and Rocha, 2017d], where conditionals are either obligations or factual conditionals and the condition can be either necessary or sufficient for the consequence to be true. Depending on the characterization of the conditional and its condition, the set of abducibles differs and accordingly, different explanations are generated.

Quantified Statements and the Search for Alternative Models

The approach in [Costa, Dietz Saldanha, and Hölldobler, 2017b] extends the approach presented in Chapter 7, and shows that, by taking two additional principles in account for the representation of quantified statements, the results improve by an overall match of 89%. One of the principles assumes that participants search for alternative models when no valid conclusion can be derived. This is modeled with the help of skeptical abduction. Taking this approach as starting point, we can now reach more than the initially limited maximum of 93.6%. How much more can we improve the results now? A possible way to approach this question is to study the individual syllogistic premises. Why do some syllogistic premises predict the answers of the participants so badly? Are

there other assumptions humans do when reasoning with quantified statements, that we have not found out yet?

Integrating Probabilties

Yet, approaches exclusively based on logic might not be sufficient, but instead, an integration together with probability could be helpful for modeling human reasoning, as has been proposed by Johnson-Laird, Khemlani, and Goodwin [2015]. Reconsidering the evaluation system for conditionals in Chapter 9, instead of assuming MRFA, one could think about a possible integration with probabilities, where a probability is attached to each reduction.

Evaluation Benchmark

Commonsense reasoning, a branch of Artificial Intelligence, is concerned with, among others, the representation and the reasoning about so-called commonsense knowledge, i.e. knowledge that everyone is expected to know about. Evaluation systems are necessary to determine the performance of proposed commonsense reasoning approaches. Some evaluation systems have been presented in [Roemmele, Bejan, and Gordon, 2011, Maslan, Roemmele, and Gordon, 2015, Levesque, Davis, and Morgenstern, 2012]. This ties in with McCarthy's [1959, 1998] idea, to have a set of challenge problems, which an adequate commonsense reasoning system should be able to solve. In order to measure the adequacy of this system we need to be able to evaluate how this system performs on a whole benchmark of problems. Finally, this would allow us to compare this systems' results with other ones. Observing the emerging attention for these approaches shows us that a new awareness is currently being established about what computational systems should be capable of doing, if we intend to make them cognitively adequate.

11.3. Psychological Experiments

As we stated in the introduction, a system that aims at being cognitively adequate, has to be evaluated with respect to the way humans reason. In turn, when we want to evaluate our approach, we depend on the data cognitive scientists provide us with.

Need for Experimental Data

A question immediately arising from the third part of the thesis is to verify whether humans reason according to MRFA, i.e. do they prefer abduction to revision? Or do they prefer some other derivation, not identified yet? Do they possibly reason differently with different types of conditionals as has been investigated in [Dietz Saldanha, Hölldobler,

and Rocha, 2017d]? Does the context or the person's background knowledge influence the evaluation of the conditional as we claim in Chapter 4? Yet another aspect to consider, is that in our system, the outcome of how the conditional is evaluated, depends mainly on the order in which the conditions of the conditional are considered. A possible psychological experiment could investigate whether the order of the conditions in a conditional also matters for human reasoners. An indication for this assumption is the spatial reasoning approach that we presented in Chapter 6, where the investigated spatial reasoning task delivers evidence that the order of the premises influences the model construction. These questions and the ones presented in Section 9.4 have to be answered if we want to learn more about how humans reason with conditionals. The necessary psychological experiments can only be implemented together with experts from the area of Cognitive Science.

Problem with Aggregated Data

By only considering aggregated values of the psychological experiments, we might not see important information about the reasoning process of the participants. Ragni, Dietz, Kola, and Hölldobler [2016] reconsider a wide amount of psychological results of the Wason selection task and show that some assumptions originally made based on the aggregated data can be refuted when looking at the individual participants patterns: (1) only very few participants chose the biconditional pattern *turn all cards*, (2) not even half of the participants in the social task chose the classical logical patterns *modus ponens* and *modus tollens*, (3) the three most favored patterns in both tasks are the same and (4) the matching pattern *modus ponens* and *modus tollens*, which was always assumed to be the most favored pattern in the abstract case, appears only to be chosen by 23% of the participants. These results emphasize that we should not only look at aggregated data of psychological results, but consider the individual patterns of the participants. Further, this serves as indication that *the* human reasoner does not exist, but instead we might better search for groups of human reasoners.

These findings show how much we depend on the psychologists, who can decide on the amount of information they want to provide us with. Most of the reported results don't give us insight about the patterns the participants opted for, but instead only about the aggregated data.

12. Summary

During the last decades many psychologists and cognitive scientists have shown that humans systematically deviate from classical logical answers. Some of these psychological experiments such as Wason's selection task, Byrne's suppression task and Evans, Barston, and Pollard's syllogistic reasoning, have been discussed in this thesis. Instead of simply arguing that human reasoning cannot be adequately modeled by any logic in general, we just put into question *classical* logic. We argued that even though classical logic is not adequate to model episodes of human reasoning, there might be other non-classical logic approaches that could be appropriate. Our goal was to formalize episodes of human reasoning with respect to conditionals within a non-monotonic approach. Yet, the findings of this thesis should not be reduced to just a formalization of these reasoning tasks.

The goal of the first part of this thesis was to allow an easy access to the Weak Completion Semantics and to clarify where to categorize this semantics in relation to other already existing approaches. In particular we showed the formal correspondence of the Weak Completion Semantics and the Well-founded Semantics. Additionally, we proposed an extension, contextual reasoning, which allows us to syntactically determine, which explanations should be preferred over others, depending on the context.

The second part of the thesis was about modeling well-known human reasoning tasks within the Weak Completion Semantics. One aspect was to investigate whether the Weak Completion Semantics is adequate for modeling human reasoning tasks. Another aspect was to explore how we could apply Logic Programming techniques, such as abduction and integrity constraints in the modeling process. We presented the formalizations of Byrne's suppression task, Wason's selection task, a spatial reasoning task and a syllogistic reasoning task and the belief-bias effect. The results of Chapter 7, which is about modeling quantified statements and predicting participants' conclusions, stand out here, because for the first time we were actually able to evaluate the performance of the Weak Completion Semantics by comparing the obtained results to the results of other approaches. We predicted 64 syllogisms with only one logic programming representation for each of the four possible quantified statements and showed that the Weak Completion Semantics performs better than any of the other twelve cognitive theories.

The goal of the third part emerged from the results of the second part: After having shown that the Weak Completion Semantics seems to be adequate for modeling a broad range of human reasoning tasks, it was just natural to construct a general system for the evaluation of conditionals. We proposed an abstract reduction system and formulated

a hypothesis: We assume that humans have a preferred derivation when reasoning with conditionals, namely MRFA, i.e. they prefer abduction over revision. We illustrated this derivation with the help of examples in the literature. Finally, we investigated the formal correspondence of MRFA with another conditional evaluation approach proposed by Schulz, and showed that MRFA is more general.

Summing up, we presented a possible path to bridging the gap between Cognitive Science and Computational Logic. The results of this thesis cannot be taken as the ultimate solution. On the contrary, the findings of this thesis deliver a starting point and guideline for many new open questions within both areas, Cognitive Science and Computational Logic.

Appendices

A. Overview of Several Two- and Three-valued Semantics

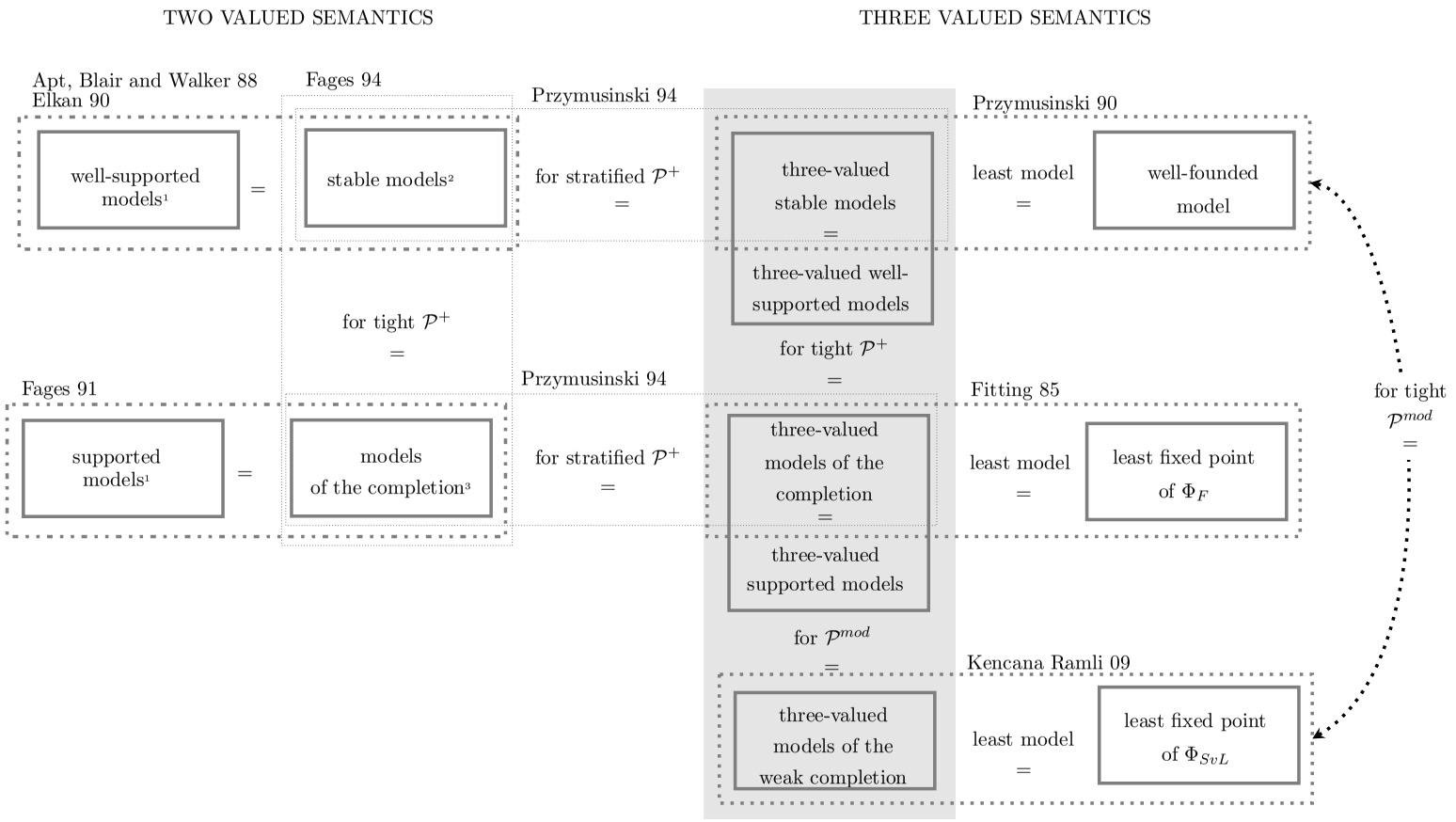


Figure A.1.

Overview of several two- and three-valued semantics. We show the correspondences in the gray box. $\mathcal{P}^{\mathsf{mod}}$ is defined as $\mathcal{P}^+ \cup \bigcup_{A \in \mathsf{undef}(\mathcal{P})} \{A \leftarrow \neg A',\ A' \leftarrow \neg A\}$ where $\mathcal{P}^+$ is $\mathcal{P}$ without assumptions.

[1] Supported and well-supported models are discussed in Section 3.2.3.

[2] Stable models are discussed in Section 3.2.1.

[3] Models of the completion are discussed in Section 2.1.

B. Level Mapping Characterization for the Weak Completion Semantics and the Well-Founded Semantics

We compare weak completion and well-founded semantics by their level mapping characterizations. For this purpose we need to define a *three-valued level mapping* for $\mathcal{P}$ which is a level mapping that may be undefined for some atoms. An *I-three-valued level mapping* ℓ_I for $\mathcal{P}$ is a three-valued level mapping for an interpretation where the domain of ℓ_I is $dom(\ell_I) = I^\top \cup I^\bot$ and ℓ_I is a function $\ell_I : I^\top \cup I^\bot \to \mathbb{N}$. All atoms which are unknown under I are not mapped to a number by ℓ_I.

Hitzler and Wendt [2005] characterize well-founded semantics for normal logic programs as follows: Let $\mathcal{P}$ be a normal program, $I = \langle I^\top, I^\bot \rangle$ be a model of $\mathcal{P}$ and ℓ_I be an I-three-valued level mapping of $\mathcal{P}$. *$\mathcal{P}$ is said to satisfy (WF) w.r.t. I and ℓ_I* if for every $A \in dom(\ell_I)$ one of the following conditions is satisfied:

(WFi) $A \in I^\top$ and there exists a clause $A \leftarrow body$ in $\mathcal{P}$ such that it holds for all literals L in $body$: $L \in I^\top$ and $\ell_I(A) > \ell_I(L)$.

(WFii) $A \in I^\bot$ and for all clauses $A \leftarrow body$ in $\mathcal{P}$, one of the following conditions holds:

- **(WFiia)** there exists a literal L in $\mathsf{pos}(body)$ such that $L \in I^\bot$ and $\ell_I(A) \geq \ell_I(L)$,
- **(WFiib)** there exists a literal L in $\mathsf{neg}(body)$ such that $L \in I^\top$ and $\ell_I(A) > \ell_I(L)$.

If $A \in dom(\ell_I)$ satisfies (WFi), then we say that A *satisfies* (WFi) *with respect to I and ℓ_I*, and similarly if $A \in dom(\ell_I)$ satisfies (WFii).

Theorem B.1. *Let $\mathcal{P}$ be a normal program with the well-founded model M. Then M is the greatest model among all models I for which there exists an I-three-valued level mapping ℓ_I for $\mathcal{P}$ such that $\mathcal{P}$ satisfies (WF) w.r.t. I and ℓ_I.*

Intuitively, a level mapping that satisfies (WF) w.r.t. to all $A \in dom(\ell_I)$ is acyclic w.r.t. $\langle I^\top, \emptyset \rangle$ and stratified w.r.t. $\langle \emptyset, I^\bot \rangle$.

Kencana Ramli [2009] gives the following level mapping characterization for the least model of the weak completion semantics:

Let $\mathcal{P}$ be a logic program, $I = \langle I^\top, I^\bot \rangle$ be a model of $\mathcal{P}$ and ℓ_I be an I-three-valued level mapping for $\mathcal{P}$. *$\mathcal{P}$ is said to satisfy (L) w.r.t. I and ℓ_I* if for every $A \in dom(\ell_I)$ one of the following conditions is satisfied:

(WCi) $A \in I^\top$ and there exists a clause $A \leftarrow \mathit{body}$ in $\mathcal{P}$ such that it holds for all literals L in body: $L \in I^\top$ and $\ell_I(A) > \ell_I(L)$.

(WCii) $A \in I^\bot$ and there exists a clause $A \leftarrow \mathit{body}$ in $\mathcal{P}$ and for all such clauses, one of the following conditions holds:

(WCiia) there exists a literal L in $\mathsf{pos}(\mathit{body})$ such that $L \in I^\bot$ and $\ell_I(A) > \ell_I(L)$,

(WCiib) there exists a literal L in $\mathsf{neg}(\mathit{body})$ such that $L \in I^\top$ and $\ell_I(A) > \ell_I(L)$.

If $A \in \mathit{dom}(\ell_I)$ satisfies (WCi), then we say that A satisfies (WCi) w.r.t. I and ℓ_I, and similarly if $A \in \mathit{dom}(\ell_I)$ satisfies (WCii).

Theorem B.2. *Let $\mathcal{P}$ be a logic program with M, the least model of the weak completion. Then M is the greatest model among all models I for which there exists an I-three-valued level mapping ℓ_I of $\mathcal{P}$ such that $\mathcal{P}$ satisfies (WC) w.r.t. I and ℓ_I.*

Intuitively, the level mapping that satisfies (WC) w.r.t. to all $A \in \mathit{dom}(\ell_I)$ is acyclic w.r.t. $\langle I^\top, \emptyset\rangle$ and w.r.t. $\langle \emptyset, I^\bot\rangle$.

Both characterizations differ on two conditions: First, consider the conditions (WFii) and (WCii):

(WFii) $A \in I^\bot$ and for all clauses $A \leftarrow \mathit{body}$ in $\mathcal{P}$, one of the following conditions holds: [...]

(WCii) $A \in I^\bot$ and there exists a clause $A \leftarrow \mathit{body}$ in $\mathcal{P}$ and for all such clauses, one of the following conditions holds: [...]

By condition (WFii) all undefined atoms are in $I^\bot$ in the well-founded model whereas under weak completion semantics, they stay unknown. Furthermore, they differ in conditions (WFiia) and (WCiia):

(WFiia) there exists a literal L in $\mathsf{pos}(\mathit{body})$ such that $L \in I^\bot$ and $\ell(A) \geq \ell(L)$,

(WCiia) there exists a literal L in $\mathsf{pos}(\mathit{body})$ such that $L \in I^\bot$ and $\ell(A) > \ell(L)$,

In a well-founded model, all atoms which are part of a positive cycle are in $I^\bot$, whereas under weak completion these atoms stay unknown. Considering Theorem 3.9 again, we made one restriction and two adaptations:

1. We restrict the correspondence to tight logic programs because of the difference between condition (WFiia) and condition (WCiia).
2. Under well-founded semantics we consider $\mathcal{P}^+$ instead of $\mathcal{P}$ because well-founded semantics is not defined for programs with negative facts.
3. For all atoms $A \in \mathsf{undef}(\mathcal{P})$ we introduce an auxiliary atom A' and add the following two clauses $A \leftarrow \neg A'$ and $A' \leftarrow \neg A$, so condition (WFii) does not apply for undefined atoms anymore and A stays unknown.

C. Ground Program of Example 4

The ground program for Example 4, $g\mathcal{P}_4$, consists of the following clauses:

$$
\begin{array}{rcll}
l(p,h,1) & \leftarrow & \top. & \text{1. Add premises} \\
l(d,h,2) & \leftarrow & \top. & \\
\\
l(p,h,1) & \leftarrow & \bot. & \text{2. Closed-world assumption left relation} \\
l(p,d,1) & \leftarrow & \bot. & \\
l(h,p,1) & \leftarrow & \bot. & \\
l(h,d,1) & \leftarrow & \bot. & \\
l(d,p,1) & \leftarrow & \bot. & \\
l(d,h,1) & \leftarrow & \bot. & \\
\\
ol(p,1) & \leftarrow & \bot. & \text{3. Closed-world assumption occupied phase 1} \\
ol(d,1) & \leftarrow & \bot. & \\
ol(h,1) & \leftarrow & \bot. & \\
or(p,1) & \leftarrow & \bot. & \\
or(d,1) & \leftarrow & \bot. & \\
or(h,1) & \leftarrow & \bot. & \\
\\
ln(p,h,1) & \leftarrow & l(p,h,1) \wedge \neg ol(h,1) \wedge \neg or(p,1). & \text{4. Neighbor left relation} \\
ln(p,h,2) & \leftarrow & l(p,h,2) \wedge \neg ol(h,2) \wedge \neg or(p,2). & \\
ln(p,d,1) & \leftarrow & l(p,d,1) \wedge \neg ol(d,1) \wedge \neg or(p,1). & \\
ln(p,d,2) & \leftarrow & l(p,d,2) \wedge \neg ol(d,2) \wedge \neg or(p,2). & \\
ln(h,p,1) & \leftarrow & l(h,p,1) \wedge \neg ol(p,1) \wedge \neg or(h,1). & \\
ln(h,p,2) & \leftarrow & l(h,p,2) \wedge \neg ol(p,2) \wedge \neg or(h,2). & \\
ln(h,d,1) & \leftarrow & l(h,d,1) \wedge \neg ol(d,1) \wedge \neg or(h,1). & \\
ln(h,d,2) & \leftarrow & l(h,d,2) \wedge \neg ol(d,2) \wedge \neg or(h,2). & \\
ln(d,p,1) & \leftarrow & l(d,p,1) \wedge \neg ol(p,1) \wedge \neg or(d,1). & \\
ln(d,p,2) & \leftarrow & l(d,p,2) \wedge \neg ol(p,2) \wedge \neg or(d,2). & \\
ln(d,h,1) & \leftarrow & l(d,h,1) \wedge \neg ol(h,1) \wedge \neg or(d,1). & \\
ln(d,h,2) & \leftarrow & l(d,h,2) \wedge \neg ol(h,2) \wedge \neg or(d,2). & \\
\\
ln(p,h,2) & \leftarrow & ln(p,h,1). & \text{5. Neighbor left relation stays always} \\
ln(p,d,2) & \leftarrow & ln(p,d,1). & \\
ln(h,p,2) & \leftarrow & ln(h,p,1). & \\
ln(h,d,2) & \leftarrow & ln(h,d,1). & \\
ln(d,h,2) & \leftarrow & ln(d,h,1). & \\
ln(d,p,2) & \leftarrow & ln(d,p,1). &
\end{array}
$$

$ol(p,2)$	$\leftarrow$	$ln(h,p,1).$	6. Occupied left/ occupied right relation
$ol(p,2)$	$\leftarrow$	$ln(d,p,1).$	
$ol(h,2)$	$\leftarrow$	$ln(p,h,1).$	
$ol(h,2)$	$\leftarrow$	$ln(d,h,1).$	
$ol(d,2)$	$\leftarrow$	$ln(p,d,1).$	
$ol(d,2)$	$\leftarrow$	$ln(h,d,1).$	
$or(p,2)$	$\leftarrow$	$ln(p,h,1).$	
$or(p,2)$	$\leftarrow$	$ln(p,d,1).$	
$or(h,2)$	$\leftarrow$	$ln(h,p,1).$	
$or(h,2)$	$\leftarrow$	$ln(h,d,1).$	
$or(d,2)$	$\leftarrow$	$ln(d,p,1).$	
$or(d,2)$	$\leftarrow$	$ln(d,h,1).$	
$l(p,h,2)$	$\leftarrow$	$l(p,d,2) \wedge ln(h,d,1).$	7. Left relation and neighbor
$l(p,d,2)$	$\leftarrow$	$l(p,h,2) \wedge ln(d,h,1).$	
$l(d,h,2)$	$\leftarrow$	$l(d,p,2) \wedge ln(h,p,1).$	
$l(d,p,2)$	$\leftarrow$	$l(d,h,2) \wedge ln(p,h,1).$	
$l(h,d,2)$	$\leftarrow$	$l(h,p,2) \wedge ln(d,p,1).$	
$l(h,p,2)$	$\leftarrow$	$l(h,d,2) \wedge ln(p,d,1).$	
$l(p,h,2)$	$\leftarrow$	$l(d,h,2) \wedge ln(d,p,1).$	8. Left relation and neighbor
$l(p,d,2)$	$\leftarrow$	$l(d,d,2) \wedge ln(h,p,1).$	
$l(d,h,2)$	$\leftarrow$	$l(p,h,2) \wedge ln(p,d,1).$	
$l(d,p,2)$	$\leftarrow$	$l(h,p,2) \wedge ln(h,d,1).$	
$l(h,p,2)$	$\leftarrow$	$l(d,p,2) \wedge ln(d,h,1).$	
$l(h,d,2)$	$\leftarrow$	$l(p,d,2) \wedge ln(p,h,1).$	
$left(X,Y)$	$\leftarrow$	$ln(X,Y,2).$	9. Conclusions about left relations
$left(X,Z)$	$\leftarrow$	$left(X,Y) \wedge leftOf(Y,Z).$	10. Left relation is transitive
$right(X,Y)$	$\leftarrow$	$left(Y \wedge X).$	11. Right relation

D. Predictions of 64 Syllogistic Premises under the Weak Completion Semantics compared to Participants' Responses

Syllogism	Premises	Weak Completion Semantics									Participants									Match in %
		Aac	Eac	Iac	Oac	Aca	Eca	Ica	Oca	NVC	Aac	Eac	Iac	Oac	Aca	Eca	Ica	Oca	NVC	
AA1	Aab, Abc	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	89
AA2	Aba, Acb	0	0	1	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	78
AA3	Aab, Acb	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	89
AA4	Aba, Abc	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	78
AI1	Aab, Ibc	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	78
AI2	Aba, Icb	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	100
AI3	Aab, Icb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	89
AI4	Aba, Ibc	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	78
AE1	Aab, Ebc	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	78
AE2	Aba, Ecb	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	89
AE3	Aab, Ecb	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	89
AE4	Aba, Ebc	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	89
AO1	Aab, Obc	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	78
AO2	Aba, Ocb	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	1	78
AO3	Aab, Ocb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	89
AO4	Aba, Obc	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	78
IA1	Iab, Abc	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	89
IA2	Iba, Acb	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	67
IA3	Iab, Acb	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	89
IA4	Iba, Abc	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	78
II1	Iab, Ibc	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	89
II2	Iba, Icb	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	1	78
II3	Iab, Icb	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	89
II4	Iba, Ibc	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	89
IE1	Iab, Ebc	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	67
IE2	Iba, Ecb	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	89
IE3	Iab, Ecb	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	1	78
IE4	Iba, Ebc	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	67
IO1	Iab, Obc	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	89
IO2	Iba, Ocb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	100
IO3	Iab, Ocb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	100
IO4	Iba, Obc	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	89
EA1	Eab, Abc	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	100
EA2	Eba, Acb	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	67
EA3	Eab, Acb	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	89
EA4	Eba, Abc	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	89
EI1	Eab, Ibc	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	89
EI2	Eba, Icb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	89
EI3	Eab, Icb	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	1	78
EI4	Eba, Ibc	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	67
EE1	Eab, Ebc	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	89
EE2	Eba, Ecb	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	1	78
EE3	Eab, Ecb	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	89
EE4	Eba, Ebc	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	89
EO1	Eab, Obc	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	1	78
EO2	Eba, Ocb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	100
EO3	Eab, Ocb	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	89
EO4	Eba, Obc	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	100
OA1	Oab, Abc	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	1	78
OA2	Oba, Acb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	78
OA3	Oab, Acb	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	1	78
OA4	Oba, Abc	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	78
OI1	Oab, Ibc	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	0	1	78
OI2	Oba, Icb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	89
OI3	Oab, Icb	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	89
OI4	Oba, Ibc	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	100
OE1	Oab, Ebc	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	1	78
OE2	Oba, Ecb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	89
OE3	Oab, Ecb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	100
OE4	Oba, Ebc	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	100
OO1	Oab, Obc	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	89
OO2	Oba, Ocb	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	89
OO3	Oab, Ocb	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	89
OO4	Oba, Obc	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	89
Overall																				85

Table D.1.: Predictions under the Weak Completion Semantics (columns 3 to 11) and the participants' responses (columns 12 to 20). The matching percentage is in the last column.

E. Proof for $\mathbf{S}_{dog}$ and $\mathbf{S}_{vit}$ in Natural Deduction

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\lfloor((\forall X)(p(X)\rightarrow\neg v(X))\wedge(\exists Y)(h(Y)\wedge v(Y)))\rfloor^{1}}{(\exists Y)(h(Y)\wedge v(Y))}(\wedge E) \quad \dfrac{\dfrac{\lfloor(h(Z)\wedge v(Z))\rfloor^{3}}{h(Z)}(\wedge E)}{(h(Z)\wedge\neg p(Z))}}{(\exists Z)(h(Z)\wedge\neg p(Z))}(\exists I)}{(((\forall X)(p(X)\rightarrow\neg v(X))\wedge(\exists Y)(h(Y)\wedge v(Y)))\rightarrow((\exists Z)(h(Z)\wedge\neg p(Z))))}(\rightarrow I)^{1}}{(\exists Z)(h(Z)\wedge\neg p(Z))}(\exists E)^{3} \quad \dfrac{\dfrac{\dfrac{\lfloor(h(Z)\wedge v(Z))\rfloor^{3}}{v(Z)}(\wedge E) \quad \dfrac{\lfloor p(Z)\rfloor^{2}}{(p(Z)\rightarrow\neg v(Z))}(\rightarrow E)}{\neg v(Z)}(\neg E)}{\dfrac{\square}{\neg p(Z)}(\neg I)^{2}}}{(h(Z)\wedge\neg p(Z))}(\wedge I) \quad \dfrac{\dfrac{\lfloor((\forall X)(p(X)\rightarrow\neg v(X))\wedge(\exists Y)(h(Y)\wedge v(Y)))\rfloor^{1}}{(\forall X)(p(X)\rightarrow\neg v(X))}(\wedge E)}{(p(Z)\rightarrow\neg v(Z))}(\forall E)$$

Figure E.1.: A proof for $((((\forall X)(p(X)\rightarrow\neg v(X))\wedge(\exists Y)(h(Y)\wedge v(Y))))\rightarrow((\exists Z)(h(Z)\wedge\neg p(Z))))$ in natural deduction according to the notation and the definitions in [Hölldobler, 2009].[a]

[a] There are two types of rules: the introduction rules (I) and the elimination rules (E). The assumptions are surrounded by the brackets $\lfloor$ and $\rfloor$, and annotated with an index. The rules which are annotated with an index refer to the assumptions that triggered these rules. For a complete description of the calculus, see [Hölldobler, 2009].

List of Symbols

$\mathcal{P}$	logic program
$g\mathcal{P}$	ground logic program
$\mathcal{P}^+$	logic program without negative facts
$\mathcal{A}\mathsf{t}$	non-empty finite set of atoms
$\mathsf{atoms}(\mathcal{P})$	set of all atoms occurring in $g\mathcal{P}$
$\mathsf{defined}(\mathcal{P})$	set of all atoms defined in $g\mathcal{P}$
$\mathsf{undef}(\mathcal{P})$	set of all atoms undefined in $g\mathcal{P}$
$\mathsf{def}(\mathcal{L}, \mathcal{P})$	definition of the set of literals $\mathcal{L}$ in $g\mathcal{P}$
$\mathcal{C}$	fixed set of constants
$\mathsf{constants}(\mathcal{P})$	set of all constants occurring in $\mathcal{P}$
$\mathsf{c}\,\mathcal{P}$	completion of $g\mathcal{P}$
$\mathsf{wc}\,\mathcal{P}$	weak completion of $g\mathcal{P}$
$\top$	truth value *true*
$\bot$	truth value *false*
U	truth value *unknown*
$I = \langle I^\top, I^\bot \rangle$	three-valued interpretation
$I^\top$	interpretation consisting of the atoms that are mapped to $\top$
$I^\bot$	interpretation consisting of the atoms that are mapped to $\bot$
$\preceq_t$	truth-ordering
$\preceq_k$	knowledge-ordering
$\mathcal{M}$	model
$\leftarrow_{\mathrm{K}}$	implication under SvL-semantics
$\leftarrow_{\mathrm{L}}$	implication under Łukasiewicz Semantics
$\leftarrow_{\mathrm{S}}$	implication under S-semantics
$\mathsf{lm}_2\mathcal{P}$	least two-valued model of $\mathcal{P}$
$\mathsf{lm}\,\mathsf{wc}\,\mathcal{P}$	least model of the weak completion of $\mathcal{P}$
lfp	least fixed point
$\models^s_{wcs}$	skeptical consequence relation under weak completion semantics
$\models^c_{wcs}$	credulous consequence relation under weak completion semantics
$\models_{wcs}$	consequence relation under weak completion semantics
$\mathsf{T}_\mathcal{P}$	two-valued semantic operator
$\Phi_\mathcal{P}$	semantic operator introduced by Stenning and van Lambalgen
$\Phi_{\mathsf{F},\mathcal{P}}$	semantic operator introduced by Fitting
$\mathrm{U} \leftarrow \mathit{body}$	integrity constraint
$\mathcal{IC}$	set of integrity constraints

$\mathcal{A}$	set of abducibles
$\mathcal{O}$	observation, non-empty set of literals
$\mathcal{E}$	explanation, set of facts and assumptions
ctxt	truth functional context operator

List of Tables

List of Figures

Index

Bibliography

E. W. Adams. Subjunctive and indicative conditionals. *Foundations of Language*, 6(1): 89–94, 1970. (Cited on page 155.)

J. Adler and L. Rips. *Reasoning: Studies of Human Inference and Its Foundations.* Cambridge University Press, 2008. (Cited on page 137.)

A. R. Anderson and N. D. Belnap. *Entailment: The Logic of Relevance and Necessity, Vol. I.* Princeton University Press, NJ, 1975. (Cited on page 173.)

A. R. Anderson, N. D. Belnap, and J. M. Dunn. *Entailment: The Logic of Relevance and Necessity, Vol. II.* Princeton University Press, NJ, 1992. (Cited on page 173.)

K. R. Apt and M. H. van Emden. Contributions to the theory of logic programming. *Journal of the ACM*, 29(3):841–862, 1982. (Cited on page 33.)

K. R. Apt, H. A. Blair, and A. Walker. Towards a theory of declarative knowledge. In J. Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 89–148. Morgan Kaufmann, San Francisco, CA, 1988. (Cited on page 49.)

F. Baader and T. Nipkow. *Term Rewriting and All That.* Cambridge University Press, United Kingdom, 1998. (Cited on pages 160 and 164.)

S. Bader, P. Hitzler, S. Hölldobler, and A. Witzel. The core method: Connectionist model generation for first-order logic programs. In B. Hammer and P. Hitzler, editors, *Perspectives of Neural-Symbolic Integration*, volume 77 of *Studies in Computational Intelligence*, pages 205–232. Springer Berlin Heidelberg, 2007. (Cited on pages 98 and 191.)

S. Banach. Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fundamenta Mathematicae*, 3(1):133–181, 1922. (Cited on page 70.)

C. Baral and M. Hunsaker. Using the probabilistic logic programming language p-log for causal and counterfactual reasoning and non-naive conditioning. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence, (IJCAI 2007)*, pages 243–249, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc. (Cited on page 156.)

C. Baral, M. Gelfond, and J. N. Rushton. Probabilistic reasoning with answer sets. *Theory and Practice of Logic Programming*, 9(1):57–144, 2009. (Cited on page 156.)

S. Beller and A. Bender. Competent deontic reasoning: The abstract deontic selection task revisited. In N. Miyake, D. Peebles, and R. P. Cooper, editors, *Proceedings of the 34th Annual Conference of the Cognitive Science Society (CogSci 2012)*, pages 114–119. Austin, TX: Cognitive Science Society, 2012. (Cited on page 93.)

T. J. M. Bench-Capon. Representing counterfactual conditionals. In A. G. Cohn, editor, *Proceedings of the Seventh Conference of the Society for the Study of Artificial Intelligence and Simulation of Behaviour*, pages 51–60. Pitman and Kaufmann, Brighton, England, 1989. (Cited on page 156.)

A. Beringer and S. Hölldobler. On the adequateness of the connection method. In *In Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 9–14, 1993. (Cited on page 15.)

W. Bibel. Perspectives on automated deduction. In R. Boyer, editor, *Automated Reasoning: Essays in Honor of Woody Bledsoe*, pages 77–104. Kluwer Academic, Utrecht, 1991. (Cited on page 15.)

R. M. Byrne. Suppressing valid inferences with conditionals. *Cognition*, 31:61–83, 1989. (Cited on pages 11, 16, 32, 85, 87, 89, 97, and 195.)

R. M. Byrne. *The Rational Imagination: How People Create Alternatives to Reality.* MIT Press, Cambridge, MA, 2007. (Cited on page 172.)

R. M. Byrne. Counterfactual thought. *Annual Review of Psychology*, 67(1):135–157, 2016. (Cited on page 156.)

R. M. Byrne and Johnson-Laird. Spatial reasoning. *Journal of Memory and Language*, 28(5):564–575, 1989. (Cited on page 102.)

L. J. Chapman and J. P. Chapman. Atmosphere effect re-examined. *Journal of Experimental Psychology*, 58(3):220–6, 1959. (Cited on page 136.)

K. L. Clark. Negation as failure. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, volume 1, pages 293–322. Plenum Press, New York, NY, 1978. (Cited on pages 28, 43, 64, and 66.)

A. Costa, E.-A. Dietz, S. Hölldobler, and M. Ragni. Syllogistic reasoning under the weak completion semantics. In U. Furbach and C. Schon, editors, *Proceedings of the Workshop on Bridging the Gap between Human and Automated Reasoning on the 25th International Joint Conference on Artificial Intelligence (IJCAI 2016)*, CEUR Workshop Proceedings. CEUR-WS.org, 2016. (Cited on pages 18 and 117.)

A. Costa, E.-A. Dietz, S. Hölldobler, and M. Ragni. A computational logic approach to human syllogistic reasoning. 2017a. submitted. (Cited on page 117.)

A. Costa, E.-A. Dietz Saldanha, and S. Hölldobler. Monadic reasoning using weak completion semantics. In S. Hölldobler, A. Malikov, and C. Wernhard, editors, *Proceedings of the Young Scientist's Second International Workshop on Trends in Information Processing (YSIP 2)*, CEUR Workshop Proceedings. CEUR-WS.org, 2017b. (Cited on page 191.)

B. Davey and H. Priestley. *Introduction to Lattices and Order*. Cambridge mathematical text books. Cambridge University Press, 2002. (Cited on page 68.)

T. Dawson, E. Gilovich and D. T. Regan. Motivated reasoning and performance on the wason selection task. *Personality and Social Psychology Bulletin*, 28(10):1379–1387, 2002. (Cited on page 99.)

E.-A. Dietz. A computational logic approach to syllogisms in human reasoning. In U. Furbach and C. Schon, editors, *Proceedings of the Workshop on Bridging the Gap between Human and Automated Reasoning on the 25th International Conference on Automated Deduction (CADE 25)*, CEUR Workshop Proceedings, pages 17–31. CEUR-WS.org, 2015. (Cited on page 133.)

E.-A. Dietz. A computational logic approach to the belief bias in human syllogistic reasoning. In *10th International and Interdisciplinary Conference on Modeling and Using Context*, volume 10257 of *Lecture Notes in Computer Science*. Springer, 2017. (Cited on pages 19 and 133.)

E.-A. Dietz and S. Hölldobler. Modeling the suppression task under three-valued Łukasiewicz and well-founded semantics. In P. Egré and R. Ripley, editors, *Proceedings ESSLLI 2012 workshop on trivalent logics and their applications*, pages 27–36, 2012. (Cited on page 43.)

E.-A. Dietz and S. Hölldobler. A new computational logic approach to reason with conditionals. In F. Calimeri, G. Ianni, and M. Truszczynski, editors, *Logic Programming and Nonmonotonic Reasoning, 13th International Conference, (LPNMR 2015)*, volume 9345 of *Lecture Notes in Artificial Intelligence*, pages 265–278. Springer, 2015. (Cited on pages 20 and 181.)

E.-A. Dietz, S. Hölldobler, and M. Ragni. A computational logic approach to the suppression task. In N. Miyake, D. Peebles, and R. P. Cooper, editors, *Proceedings of the 34th Annual Conference of the Cognitive Science Society, (CogSci 2012)*, pages 1500–1505. Austin, TX: Cognitive Science Society, 2012a. (Cited on pages 16, 18, and 85.)

E.-A. Dietz, S. Hölldobler, and M. Ragni. A simple model for the Wason selection task. In T. Barkowsky, M. Ragni, and F. Stolzenburg, editors, *Human Reasoning and Automated Deduction: KI 2012 Workshop Proceedings*, volume SFB/TR 8 Report 032-09/2012 of *Report Series of the Transregional Collaborative Research Center SFB/TR 8 Spatial Cognition*, pages 41–48, Bremen, 2012b. Universität Bremen / Universität Freiburg. (Cited on page 85.)

E.-A. Dietz, S. Hölldobler, and M. Ragni. A computational logic approach to the abstract and the social case of the selection task. In *Proceedings of the 11th International Symposium on Logical Formalizations of Commonsense Reasoning, (Commonsense 2013)*, Aeya Nappa, Cyprus, 2013. (Cited on pages 17, 18, 43, and 85.)

E.-A. Dietz, S. Hölldobler, and C. Wernhard. Modeling the suppression task under weak completion and well-founded semantics. *Journal of Applied Non-Classical Logics*, 24 (1-2):61–85, 2014. (Cited on pages 16, 18, 43, and 85.)

E.-A. Dietz, S. Hölldobler, and R. Höps. A computational logic approach to human spatial reasoning. In *IEEE Symposium Series on Computational Intelligence, (SSCI 2015)*, pages 1627–1634. IEEE, 2015a. (Cited on pages 18 and 101.)

E.-A. Dietz, S. Hölldobler, and L. M. Pereira. On conditionals. In G. Gottlob, G. Sutcliffe, and A. Voronkov, editors, *Global Conference on Artificial Intelligence*, Epic Series in Computing. EasyChair, 2015b. (Cited on pages 19 and 155.)

E.-A. Dietz, S. Hölldobler, and L. M. Pereira. On indicative conditionals. In S. Hölldobler and Y. Liang, editors, *Proceedings of the First International Workshop on Semantic Technologies*, volume 1339 of *CEUR Workshop Proceedings*, pages 19–30. CEUR-WS.org, 2015c. (Cited on page 155.)

E.-A. Dietz, S. Hölldobler, and M. Ragni. A syllogistic reasoning theory and three examples. In S. Hölldobler and K. Chinnasarn, editors, *Proceedings of the Second International Workshop on Semantic Technologies*, volume 1494 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015d. (Cited on pages 18 and 117.)

E.-A. Dietz Saldanha, S. Hölldobler, C. D. P. Kencana Ramli, and L. Palacios Medinacelli. A core method for the weak completion semantics with skeptical abduction. 2017a. submitted. (Cited on pages 98 and 191.)

E.-A. Dietz Saldanha, S. Hölldobler, and L. M. Pereira. Contextual reasoning: Usually birds can abductively fly. In *Logic Programming and Nonmonotonic Reasoning - 14th International Conference, (LPNMR 2017)*, 2017b. (Cited on pages 17 and 63.)

E.-A. Dietz Saldanha, S. Hölldobler, and T. Philipp. The complexity of contextual abduction in human reasoning tasks. In S. Hölldobler, A. Malikov, and C. Wernhard, editors, *Proceedings of the Young Scientist's Second International Workshop on Trends in Information Processing (YSIP 2)*, CEUR Workshop Proceedings. CEUR-WS.org, 2017c. (Cited on page 191.)

E.-A. Dietz Saldanha, S. Hölldobler, and I. L. Rocha. Obligation versus factual conditionals under the weak completion semantics. In S. Hölldobler, A. Malikov, and C. Wernhard, editors, *Proceedings of the Young Scientist's Second International Workshop on Trends in Information Processing (YSIP 2)*, CEUR Workshop Proceedings. CEUR-WS.org, 2017d. (Cited on pages 191 and 192.)

K. Dieussaert, W. Schaeken, W. Schroyens, and G. D'Ydewalle. Strategies during complex conditional inferences. *Thinking and Reasoning*, 6(2):125–161, 2000. (Cited on page 12.)

J. Dix. A classification theory of semantics of normal logic programs: I. strong properties, ii. weak properties. *Fundamenta informaticae*, 22(3):227 – 255, 257–288, 1995. (Cited on page 178.)

D. Edgington. On conditionals. *Mind*, 104(414):235–329, 1995. (Cited on page 156.)

D. Edgington. *Causation First: Why Causation is Prior to Counterfactuals*. Consciousness and self-consciousness. Oxford University Press, 2011. (Cited on page 156.)

C. Elkan. A rational reconstruction of nonmonotonic truth maintenance systems (research note). *Artificial Intelligence*, 43(2):219–234, 1990. (Cited on page 51.)

E. Erdem and V. Lifschitz. Tight logic programs. *Theory and Practice of Logic Programming*, 3(4):499–518, 2003. (Cited on page 49.)

J. S. Evans. Interpretation and matching bias in a reasoning task. *The Quarterly Journal of Experimental Psychology*, 24(2):193–199, 1972. (Cited on page 93.)

J. S. Evans. *Bias in human reasoning - causes and consequences*. Essays in cognitive psychology. Lawrence Erlbaum, 1989. (Cited on page 137.)

J. S. Evans. Thinking and believing. *Mental models in reasoning*, 2000. (Cited on page 137.)

J. S. Evans. Biases in deductive reasoning. In R. Pohl, editor, *Cognitive Illusions: A Handbook on Fallacies and Biases in Thinking, Judgement and Memory*. Psychology Press, 2012. (Cited on pages 13, 14, and 134.)

J. S. Evans and D. Over. *If*. Oxford cognitive science series. Oxford University Press, 2004. (Cited on pages 156 and 157.)

J. S. Evans, J. L. Barston, and P. Pollard. On the conflict between logic and belief in syllogistic reasoning. *Memory & Cognition*, 11(3):295–306, 1983. (Cited on pages 133, 134, 137, 140, 143, 145, 150, and 195.)

J. S. Evans, S. Handley, and C. Harper. Necessity, possibility and belief: A study of syllogistic reasoning. *Quarterly Journal of Experimental Psychology*, 54(3):935–958, 2001. (Cited on page 135.)

F. Fages. A new fixpoint semantics for general logic programs compared with the well-founded and the stable model semantics. *New Generation Computing*, 9(3/4):425–444, 1991. (Cited on page 51.)

F. Fages. Consistency of Clark's completion and existence of stable models. *Journal of Methods of Logic in Computer Science*, 1(1):51–60, 1994. (Cited on pages 49, 53, and 56.)

M. Fitting. A Kripke-Kleene semantics for logic programs. *Journal of Logic Programming*, 2(4):295–312, 1985. (Cited on pages 15, 32, 36, and 98.)

M. Fitting. Metric methods three examples and a theorem. *Journal of Logic Programming*, 21(3):113–127, 1994. (Cited on page 68.)

A. Garnham and J. Oakhill. *Thinking and Reasoning*. Wiley, 1994. (Cited on pages 136 and 137.)

M. Gebser, R. Kaminski, B. Kaufmann, and T. Schaub. *Clingo* = ASP + control: Preliminary report. In M. Leuschel and T. Schrijvers, editors, *Technical Communications of the Thirtieth International Conference on Logic Programming (ICLP 2014)*, volume arXiv:1405.3694v1, 2014. Theory and Practice of Logic Programming, Online Supplement. (Cited on page 115.)

M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In R. Kowalski and K. A. Bowen, editors, *Proceedings of the International Logic Programming Conference and Symposium, (ICLP/SLP 1988)*, pages 1070–1080, Cambridge, MA, 1988. MIT Press. (Cited on pages 44, 45, and 64.)

M. L. Ginsberg. Counterfactuals. *Artificial Intelligence*, 30(1):35–79, 1986. (Cited on page 156.)

G. P. Goodwin and P. Johnson-Laird. Reasoning about relations. *Psychological review*, 112(2):468, 2005. (Cited on page 113.)

S. Gottwald. *A Treatise on Many-Valued Logics*, volume 9 of *Studies in Logic and Computation*. Research Studies Press, Baldock, 2001. (Cited on page 32.)

H. P. Grice. Logic and conversation. In P. Cole and J. L. Morgan, editors, *Syntax and Semantics: Vol. 3: Speech Acts*, pages 41–58. Academic Press, New York, 1975. Reprinted as ch.2 of Grice 1989, pages 22–40. (Cited on page 121.)

R. A. Griggs and J. R. Cox. The elusive thematic-materials effect in wason's selection task. *British Journal of Psychology*, 73(3):407–420, 1982. (Cited on pages 85, 90, 92, and 94.)

C. S. Herrmann and F. W. Ohl. Creating brain-like intelligence. chapter Cognitive Adequacy in Brain-Like Intelligence, pages 314–327. Springer-Verlag, Berlin, Heidelberg, 2009. (Cited on page 12.)

C. S. Herrmann and F. Reine. Considering adequacy in neural network learning. In *Proceedings of International Conference on Neural Networks (ICNN 1996)*, volume 1, pages 270–275. IEEE, 1996. (Cited on page 15.)

P. Hitzler and M. Wendt. A uniform approach to logic programming semantics. *Theory and Practice of Logic Programming*, 5(1-2):93–121, 2005. (Cited on pages 49 and 201.)

C. Hoerl, T. McCormack, and S. R. Beck. *Introduction: Understanding Counterfactuals and Causation.* Consciousness and self-consciousness. Oxford University Press, 2011. (Cited on page 156.)

S. Hölldobler. *Logik und Logikprogrammierung 1: Grundlagen.* Kolleg Synchron. Synchron, 2009. (Cited on pages 25, 135, and 208.)

S. Hölldobler and C. D. P. Kencana Ramli. Logic programs under three-valued Łukasiewicz semantics. In P. M. Hill and D. S. Warren, editors, *25th International Conference on Logic Programming, (ICLP 2009)*, volume 5649 of *Lecture Notes in Computer Science*, pages 464–478, Heidelberg, 2009a. Springer. (Cited on pages 16, 35, 56, 72, 85, 87, 88, 98, and 191.)

S. Hölldobler and C. D. P. Kencana Ramli. Logics and networks for human reasoning. In C. Alippi, M. M. Polycarpou, C. G. Panayiotou, and G. Ellinas, editors, *International Conference on Artificial Neural Networks, (ICANN 2009), Part II*, volume 5769 of *Lecture Notes in Computer Science*, pages 85–94, Heidelberg, 2009b. Springer. (Cited on pages 16, 28, 32, 35, 56, 85, 87, 88, 98, 160, and 183.)

S. Hölldobler and C. D. P. Kencana Ramli. Contraction properties of a semantic operator for human reasoning. In L. Li and K. K. Yen, editors, *Proceedings of the Fifth International Conference on Information*, pages 228–231, 2009c. (Cited on page 68.)

S. Hölldobler and M. Thielscher. On the Adequateness of AI-Systems. In I. Plander, editor, *Proceedings of the International Conference on Artificial Intelligence and Information-Control Systems of Robots (AIICSR 1994)*, pages 41–46, Smolenice Castle, Slovakia, 1994. World Scientific Publishing Co. Singapore. (Invited talk). (Cited on page 15.)

S. Hölldobler, T. Philipp, and C. Wernhard. An abductive model for human reasoning. In *Logical Formalizations of Commonsense Reasoning (Commonsense 2011)*, AAAI Spring Symposium Series Technical Reports, pages 135–138, Cambridge, MA, 2011. AAAI Press. (Cited on pages 85, 88, 96, and 190.)

R. Höps. Menschliches räumliches Schließen und Ansätze aus der Computational Logic. Bachelor's thesis, Institute for Artificial Intelligence, Department of Computer Science, Technische Universität Dresden, Dresden, 2014. (Cited on pages 101 and 113.)

P. N. Johnson-Laird. *Mental models: towards a cognitive science of language, inference, and consciousness.* Harvard University Press, Cambridge, MA, 1983. (Cited on pages 102, 103, 118, 119, 121, and 136.)

P. N. Johnson-Laird and R. M. Byrne. Deduction. *Lawrence Erlbaum Associates, Inc*, 1991. (Cited on page 136.)

P. N. Johnson-Laird and R. M. J. Byrne. Conditionals: A theory of meaning, pragmatics, and inference. *Psychological Review*, 109(4):646–678, 2002. (Cited on page 157.)

P. N. Johnson-Laird, V. Girotto, and P. Legrenzi. Reasoning from inconsistency to consistency. *Psychological Review*, 111(3):640–661, 2004. (Cited on page 97.)

P. N. Johnson-Laird, S. S. Khemlani, and G. P. Goodwin. Logic, probability, and human reasoning. *Trends in Cognitive Reasoning*, 19(4):201–214, 2015. (Cited on pages 171 and 192.)

A. Kakas, R. Kowalski, and F. Toni. Abductive logic programming. *Journal of Logic and Computation*, 2(6):719–770, 1993. (Cited on page 39.)

C. D. P. Kencana Ramli. Logic programs and three-valued consequence operators. Master's thesis, Institute for Artificial Intelligence, Department of Computer Science, Technische Universität Dresden, Dresden, 2009. (Cited on pages 35, 68, 70, 71, 72, 123, 181, and 201.)

S. Khemlani and P. N. Johnson-Laird. Theories of the syllogism: A meta-analysis. *Psychological Bulletin*, 138(3):427–457, 2012. (Cited on pages 117, 118, 119, 121, 122, 125, 127, 132, and 134.)

S. C. Kleene. *Introduction to Metamathematics*. North-Holland, Amsterdam, 1952. (Cited on pages 15 and 30.)

J. W. Klop, M. Bezem, and R. C. D. Vrijer, editors. *Term Rewriting Systems*. Cambridge University Press, New York, NY, USA, 2001. (Cited on pages 160 and 164.)

M. Knauff. The cognitive adequacy of allen's interval calculus for qualitative spatial representation and reasoning. *Spatial Cognition and Computation*, 1(3):261–290, 1999. (Cited on page 12.)

M. Knauff, R. Rauh, and C. Schlieder. Preferred mental models in qualitative spatial reasoning: A cognitive assessment of Allen's calculus. In *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society, (CogSci 1995)*, pages 200–205, Hillsdale, NJ, 1995. Lawrence Erlbaum. (Cited on page 14.)

M. Knauff, R. Rauh, and J. Renz. A cognitive assessment of topological spatial relations: Results from an empirical investigation. In *Proceedings of the third International Conference on Spatial Information Theory, (COSIT 1997)*, volume 1329 of *Lecture Notes in Computer Science*, pages 193–206, Heidelberg, 1997. Springer. (Cited on page 14.)

R. Kowalski. *Computational Logic and Human Thinking: How to be Artificially Intelligent*. Cambridge University Press, Cambridge, 2011. (Cited on pages 13, 14, 90, 93, and 94.)

H. J. Levesque, E. Davis, and L. Morgenstern. The winograd schema challenge. In G. Brewka, T. Eiter, and S. A. McIlraith, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, (KR 2012), Rome, Italy, June 10-14, 2012*. AAAI Press, 2012. (Cited on page 192.)

D. Lewis. *Counterfactuals*. Blackwell Publishers, Oxford, 1973. (Cited on pages 155 and 156.)

D. Lewis. Probabilities of conditionals and conditional probabilities. *Philosophical Review*, 85(3):297–315, 1976. (Cited on page 156.)

D. Lewis. *On the Plurality of Worlds*. Blackwell Publishers, Oxford, 1986. (Cited on page 156.)

J. W. Lloyd. *Foundations of Logic Programming*. Springer-Verlag New York, Inc., New York, NY, USA, 1984. (Cited on page 25.)

J. Łukasiewicz. O logice trójwartościowej. *Ruch Filozoficzny*, 5:169–171, 1920. English translation: On three-valued logic. In: Łukasiewicz J. and Borkowski L. (ed.). (1990). *Selected Works*, Amsterdam: North Holland, pages 87–88. (Cited on pages 16, 30, and 98.)

K. Manktelow. *Reasoning and Thinking*. Psychology press, 2000. (Cited on page 104.)

E. D. Mares. *Relevant Logic : A Philosophical Interpretation*. Cambridge University Press, 2004. (Cited on page 173.)

N. Maslan, M. Roemmele, and A. S. Gordon. One Hundred Challenge Problems for Logical Formalizations of Commonsense Psychology. In *Proceedings of the Twelfth International Symposium on Logical Formalizations of Commonsense Reasoning (Commonsense 2015)*, Stanford, CA, 2015. (Cited on page 192.)

J. McCarthy. Programs with common sense. In *Proceedings of the Teddington Conference on the Mechanization of Thought Processes*, pages 75–91, London, 1959. Her Majesty's Stationary Office. (Cited on page 192.)

J. McCarthy. Epistemological problems of artificial intelligence. In *Proceedings of the 5th international joint conference on Artificial intelligence (IJCAI 1977)*, pages 1038–1044, San Francisco, CA, USA, 1977. Morgan Kaufmann Publishers Inc. (Cited on page 15.)

J. McCarthy. Elaboration tolerance. In *4th International Symposium on Logical Formalizations of Commonsense Reasoning*, 1998. (Cited on page 192.)

S. E. Newstead, S. J. Handley, and E. Buck. Falsifying mental models: Testing the predictions of theories of syllogistic reasoning. *Memory & Cognition*, 27(2):344–354, 1999. (Cited on page 150.)

M. Oaksford, N. Chater, and J. Larkin. Probabilities and polarity biases in conditional inference. *Journal of Experimental Psychology:Learning, Memory, and Cognition*, 26 (4):883–899, 2000. (Cited on page 157.)

K. Oberauer. Reasoning with conditionals: A test of formal models of four theories. *Cognitive Psychology*, 53(3):238–283, 2006. (Cited on page 157.)

T. C. Ormerod, K. I. Manktelow, and G. V. Jones. Reasoning with three types of conditional: Biases and mental models. *The Quarterly Journal of Experimental Psychology Section A*, 46(4):653–677, 1993. (Cited on page 97.)

L. Palacios Medinacelli. Skeptical abduction: A neural-symbolic approach. Master's thesis, Institute for Artificial Intelligence, Department of Computer Science, Technische Universität Dresden, Dresden, 2016. (Cited on page 191.)

J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, USA, 2000. (Cited on pages 156, 157, 158, and 167.)

J. Pearl. The algorithmization of counterfactuals. *Annals of Mathematics and Artificial Intelligence*, 61(1):29–39, 2011. (Cited on pages 156 and 157.)

L. M. Pereira and J. N. Aparício. Relevant counterfactuals. In J. P. Martins and E. M. Morgado, editors, *EPIA*, volume 390 of *Lecture Notes in Computer Science*, pages 107–118. Springer, 1989. (Cited on page 156.)

L. M. Pereira and A. M. Pinto. Inspecting side-effects of abduction in logic programs. In M. Balduccini and T. C. Son, editors, *Logic Programming, Knowledge Representation, and Nonmonotonic Reasoning: Essays in honour of Michael Gelfond*, volume 6565 of *LNAI*, pages 148–163. Springer, 2011. (Cited on pages 65, 74, and 81.)

L. M. Pereira and A. Saptawijaya. Abduction and beyond in logic programming with application to morality. *IfColog Journal of Logics and their Applications, Special issue on "Frontiers of Abduction"*, 3(1):37–71, 2016a. (Cited on pages 156 and 173.)

L. M. Pereira and A. Saptawijaya. *Programming Machine Ethics*, volume 26. Springer SAPERE series, Berlin, 2016b. (Cited on page 173.)

L. M. Pereira, J. N. Aparício, and J. J. Alferes. The extended stable models of contradiction removal semantics. In P. Barahona, L. M. Pereira, and A. Porto, editors, *5th Portuguese AI International Conference (EPIA1991)*, volume 541 of *Lecture Notes in Computer Science*, pages 105–119. Springer, Heidelberg, 1991a. (Cited on page 54.)

L. M. Pereira, J. N. Aparício, and J. J. Alferes. Hypothetical reasoning with well founded semantics. In B. Mayoh, editor, *Proceedings of the Scandinavian Conference on Artificial Intelligence (SCAI 1991)*, pages 289–300. IOS Press, Amsterdam, 1991b. (Cited on page 38.)

L. M. Pereira, E.-A. Dietz, and S. Hölldobler. An abductive reasoning approach to the belief bias effect. In C. Baral, G. D. Giacomo, and T. Eiter, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, (KR 2014), Vienna, Austria, July 20-24, 2014*, pages 653–656. AAAI Press, 2014a. (Cited on pages 17, 25, and 63.)

L. M. Pereira, E.-A. Dietz, and S. Hölldobler. Contextual abductive reasoning with side-effects. *Theory and Practice of Logic Programming*, 14(4-5):633–648, 2014b. (Cited on pages 17, 25, and 63.)

T. Philipp. Human reasoning and abduction. Bachelor's thesis, Institute for Artificial Intelligence, Department of Computer Science, Technische Universität Dresden, Dresden, 2010. (Cited on pages 159 and 162.)

A. M. Pinto and L. M. Pereira. Every normal logic program has a 2-valued minimal hypotheses semantics. In L. Antunes, H. Sofia, A. Pinto, R. Prada, and P. Trigo, editors, *Proceedings of 15th Portuguese International Conference on Artificial Intelligence (EPIA 2011)*, Epic Series in Computing, pages 283–297. EasyChair, 2011. (Cited on page 178.)

T. A. Polk and A. Newell. Deduction as verbal reasoning. *Psychological Review*, 102(3): 533–566, 1995. (Cited on pages 118, 119, and 150.)

T. C. Przymusinski. On the declarative semantics of deductive databases and logic programs. In J. Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 193–216. Morgan Kaufmann, San Francisco, CA, 1988. (Cited on page 49.)

T. C. Przymusinski. Every logic program has a natural stratification and an iterated least fixed point model. In A. Silberschatz, editor, *Proceedings of the Eighth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, (PODS 1989)*, pages 11–21, New York, NY, 1989. ACM. (Cited on pages 33 and 34.)

T. C. Przymusinski. Well-founded semantics coincides with three-valued stable semantics. *Fundamenta Informaticae*, 13(4):445–463, 1990. (Cited on pages 44, 45, 53, and 58.)

T. C. Przymusinski. Well founded and stationary models of logic programs. *Annals of Mathematics and Artificial Intelligence*, 12(3-4):141–187, 1994. (Cited on pages 43 and 44.)

M. Ragni and M. Knauff. A theory and a computational model of spatial reasoning with preferred mental models. 120(3):561 – 588, 2013. (Cited on pages 101, 102, 103, and 104.)

M. Ragni, M. Knauff, and B. Nebel. A computational model for spatial reasoning with mental models. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society (CogSci 2005)*, pages 1797–1802, 2005. (Cited on page 104.)

M. Ragni, E.-A. Dietz, I. Kola, and S. Hölldobler. Two-valued logic is not sufficient to model human reasoning, but three-valued logic is: A formal analysis. In C. Schon and U. Furbach, editors, *Proceedings of the Workshop on Bridging the Gap between Human and Automated Reasoning co-located with 25th International Joint Conference on Artificial Intelligence (IJCAI 2016), New York, USA*, volume 1651 of *CEUR Workshop Proceedings*, pages 61–73. CEUR-WS.org, 2016. (Cited on page 193.)

F. Ramsey. *The foundations of mathematics and other logical essays.* Harcourt, Brace and Company, 1931. (Cited on page 156.)

R. Reiter. A logic for default reasoning. *Artifical Intelligence*, 13(1-2):81–132, 1980. (Cited on pages 63, 64, and 190.)

J. Renz, R. Rauh, and M. Knauff. Towards cognitive adequacy of topological spatial relations. In *Spatial Cognition II, Integrating Abstract Theories, Empirical Studies, Formal Methods, and Practical Applications*, pages 184–197, London, UK, 2000. Springer-Verlag. (Cited on page 12.)

N. Rescher. *Many-valued logic.* McGraw-Hill, New York, NY, 1969. (Cited on page 33.)

N. Rescher. *What If?: Thought Experimentation In Philosophy.* Transaction Publishers, 2005. (Cited on pages 156 and 157.)

N. Rescher. *Conditionals.* MIT Press, Cambridge, MA, 2007. (Cited on pages 156 and 157.)

L. J. Rips. *The psychology of proof: Deductive reasoning in human thinking.* The MIT Press, Cambridge, MA, 1994. (Cited on pages 118, 119, and 121.)

M. Roemmele, C. A. Bejan, and A. S. Gordon. Choice of Plausible Alternatives: An Evaluation of Commonsense Causal Reasoning. In *AAAI Spring Symposium on Logical Formalizations of Commonsense Reasoning (Commonsense 2011)*, Stanford University, 2011. (Cited on page 192.)

T. Routen and T. J. M. Bench-Capon. Hierarchical formalizations. *International Journal of Man-Machine Studies*, 35(1):69–93, 1991. (Cited on page 156.)

C. Ruiz and J. Minker. Computing stable and partial stable models of extended disjunctive logic programs. In J. Dix, L. M. Pereira, and T. C. Przymusinski, editors, *Non-Monotonic Extensions of Logic Programming*, volume 927 of *Lecture Notes in Computer Science*, pages 205–229. Springer, Heidelberg, 1995. (Cited on page 30.)

K. Satoh and N. Iwayama. Computing abduction using the TMS. In K. Furukawa, editor, *Proceedings of the Eigth International Conference on Logic Programming, (ICLP 1991)*, pages 505–518, Cambridge, MA, 1991. MIT Press. (Cited on page 54.)

K. Schulz. Minimal models vs. logic programming: the case of counterfactual conditionals. *Journal of Applied Non-Classical Logics*, 24(1-2):153–168, 2014. (Cited on pages 181, 182, 183, and 185.)

L. Shastri and V. Ajjanagadde. From simple associations to systematic reasoning: a connectionist representation of rules, variables and dynamic bindings using temporal synchrony. *Behavioral and Brain Sciences*, 16:417–494, 1993. (Cited on page 15.)

N. Skovgaard-Olsen, H. Singmann, and K. C. Klauer. The relevance effect and conditionals. *Cognition*, 150:26 – 36, 2016. (Cited on page 179.)

S. Sloman. *Causal Models : How People Think about the World and Its Alternatives*. Oxford University Press, USA, 2005. (Cited on page 156.)

W. Spohn. A rankingtheoretic approach to conditionals. *Cognitive Science*, 37(6):1074–1106, 2013. (Cited on page 179.)

R. C. Stalnaker. A theory of conditionals. In N. Rescher, editor, *Studies in Logical Theory*, pages 98–112. Blackwell Publishers, Oxford, 1968. (Cited on page 156.)

R. C. Stalnaker and R. H. Thomason. A semantic analysis of conditional logic. *Theoria*, 36(1):23–42, 1970. (Cited on page 156.)

K. Stenning and M. van Lambalgen. Semantic interpretation as computation in nonmonotonic logic: The real meaning of the suppression task. *Cognitive Science*, 6(29): 916–960, 2005. (Cited on page 15.)

K. Stenning and M. van Lambalgen. *Human Reasoning and Cognitive Science*. A Bradford Book. MIT Press, Cambridge, MA, 2008. (Cited on pages 15, 32, 35, 36, 86, 87, 88, 96, 99, 120, 121, 137, and 189.)

G. Strube. The role of cognitive science in knowledge engineering. In F. Schmalhofer, G. Strube, and T. Wetter, editors, *Contemporary Knowledge Engineering and Cognition, First Joint Workshop*, volume 622 of *Lecture Notes in Computer Science*, pages 159–174. Springer, Heidelberg, 1992. (Cited on page 14.)

G. Strube. *Wörterbuch der Kognitionswissenschaft*. Klett-Cotta, Stuttgart, 1996. (Cited on page 14.)

A. Tarski. *Pacific Journal of Mathematics*, 5(2):285–309, 1955. (Cited on page 68.)

M. H. Van Emden and R. Kowalski. The semantics of predicate logic as a programming language. *ACM*, 23(4):733–742, 1976. (Cited on page 45.)

A. Van Gelder. The alternating fixpoint of logic programs with negation. In A. Silberschatz, editor, *Proceedings of the Eighth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, (PODS 1989)*, pages 1–10, New York, NY, 1989. ACM. (Cited on page 47.)

A. Van Gelder, K. A. Ross, and J. S. Schlipf. The well-founded semantics for general logic programs. *Journal of the ACM*, 38(3):619–649, 1991. (Cited on pages 44, 46, 47, and 64.)

J. Vennekens, M. Denecker, and M. Bruynooghe. CP-logic: A language of causal probabilistic events and its relation to logic programming. *CoRR*, abs/0904.1672, 2009. (Cited on page 156.)

J. Vennekens, M. Bruynooghe, and M. Denecker. Embracing events in causal modelling: Interventions and counterfactuals in CP-logic. In T. Janhunen and I. Niemelä, editors, *Logics in Artificial Intelligence - 12th European Conference, (JELIA 2010), Helsinki, Finland, September 13-15, 2010. Proceedings*, volume 6341 of *Lecture Notes in Computer Science*, pages 313–325. Springer, 2010. (Cited on page 156.)

N. Verschueren, W. Schaeken, and G. d'Ydewalle. A dual-process specification of causal conditional reasoning. *Thinking & Reasoning*, 11(3):239–278, 2005. (Cited on page 157.)

P. C. Wason. Reasoning about a rule. *Quarterly Journal of Experimental Psychology*, 20(3):273–281, 1968. (Cited on pages 85, 90, 91, 92, 95, and 195.)

C. Wernhard. Forward human reasoning modeled by logic programming modeled by classical logic with circumscription and projection. Technical Report Knowledge Representation and Reasoning 11-07, Technische Universität Dresden, Dresden, 2011. (Cited on page 16.)

C. Wernhard. Towards a declarative approach to model human reasoning with nonmonotonic logics. In T. Barkowsky, M. Ragni, and F. Stolzenburg, editors, *Human Reasoning and Automated Deduction: KI 2012 Workshop Proceedings*, volume SFB/TR 8 Report 032-09/2012 of *Report Series of the Transregional Collaborative Research Center SFB/TR 8 Spatial Cognition*, pages 41–48, Bremen, 2012. Universität Bremen / Universität Freiburg. (Cited on pages 16 and 60.)

M. Wilkins. The effect of changed material on the ability to do formal syllogistic reasoning. *Archives of Psychology*, 16(102):1–83, 1928. (Cited on page 136.)

J. Woodward. Psychological studies of causal and counterfactual reasoning. In C. Hoerl, T. McCormack, and S. R. Beck, editors, *Understanding counterfactuals, understanding causation : issues in philosophy and psychology*, Consciousness and self-consciousness. Oxford University Press, 2011. (Cited on page 156.)

R. S. Woodworth and S. B. Sells. An atmosphere effect in formal syllogistic reasoning. *Journal of Experimental Psychology*, 18(4):451–60, 1935. (Cited on page 136.)