Human Reasoning and Automated Deduction

KI 2012 Workshop Proceedings

Thomas Barkowsky, Marco Ragni, Frieder Stolzenburg (Eds.)

SFB/TR 8 Report No. 032-09/2012
Report Series of the Transregional Collaborative Research Center SFB/TR 8 Spatial Cognition
Universität Bremen / Universität Freiburg

Contact Address:

Dr. Thomas Barkowsky
SFB/TR 8
Universität Bremen
P.O.Box 330 440
28334 Bremen, Germany

Tel +49-421-218-64233
Fax +49-421-218-64239
barkowsky@sfbtr8.uni-bremen.de
www.sfbtr8.uni-bremen.de

Human Reasoning and Automated Deduction

KI 2012 Workshop Proceedings

Thomas Barkowsky, Marco Ragni, and Frieder Stolzenburg (Eds.)

Preface

Automated deduction aims at formalizing diverse aspects of reasoning and has many application areas from software verification to mathematical theorem proving. It is originally based on algorithmic methods derived from mathematical logics. In contrast, *human reasoning* cannot be completely described by classical logical systems. Sources of explanations are incomplete knowledge, incorrect beliefs, or inconsistencies. Still, humans have an impressive ability to derive acceptable conclusions. From the very beginning of AI research, there has been a strong emphasis on incorporating mechanisms of human rationality and cognition into reasoning systems.

This workshop continues a series of successful workshops initiated by the Special Interest Group *"Cognition"* in the GI. This fifth workshop, which is held in conjunction with KI 2012, aims at bringing together researchers from AI, Automated Deduction, Computational Logics, and Cognitive Science to foster a multi-disciplinary exchange and to discuss possibilities to overcome the historic separation. The call was open for different topics and we received a variety of papers: contributions which are strongly focused on non-monotonic logical approaches as a new and vibrant possibility to explain human reasoning and as a bridging function to inspire both fields, on using cognitive systems for moral reasoning, on manipulation tasks in cognitive robotics, and on human reasoning in abstract and social contexts. Taken together, we are surprised by the already existing inter- and transdisciplinary work, and we see that both fields are not as distinct as we initially thought. Our wish is that new inspirations and collaborations will emerge from this workshop.

The organizers of this workshop would like to thank the organizers of the KI 2012 conference, the Spatial Cognition Research Center SFB/TR 8 and the Special Priority Program "New Frameworks of Rationality" (SPP 1516) for their support. We also would like to thank the members of the Program Committee for their help in selecting and improving the submitted papers, and finally all participants of the workshop for their contributions.

Thomas Barkowsky
Marco Ragni
Frieder Stolzenburg

Contents

A Simple Model for the Wason Selection Task

Emmanuelle-Anna Dietz*, Steffen Hölldobler*, Marco Ragni**

International Center for Computational Logic, Technische Universität Dresden
D-01062 Dresden, Germany
Center for Cognitive Science, Friedrichstraße 50
D-79098 Freiburg, Germany

Abstract. The Wason selection task is probably the most famous and best investigated research paradigm in the psychology of reasoning. In the classical abstract version people are presented with cards and have to check a conditional statement. Numerous psychological studies have shown that most people do not solve this task in terms of classical logic correctly and tend to make similar reasoning errors. When the same reasoning problem is framed within a social setting, most people solve the task correctly. All major reasoning theories have tried to explain the logical errors and the differences between the abstract and the social framing. In this paper we present a new computational logic approach based on the three-valued Łukasiewicz logic. According to Kowalski's representation, we formalize the abstract and the social case, and show that when reasoning towards the corresponding representations, our computational approach adequately reflects the psychological results.

1 Introduction

In the last century the classical (propositional) logic calculus has played an important role as a normative concept for psychologists investigating human reasoning. Psychological research, however, showed that humans systematically deviate from the logically correct answers. Some attempts to formalize this behavior are already made in the field of Computational Logic such as in non-monotonic logic, common sense reasoning or three-valued logics, where incomplete information is expressible. Furthermore, the field of Artificial Neural Networks and Cognitive Science focus on challenging problems that aim to simulate and understand human reasoning. Their results are important for our purpose as they give detailed insight about reasoning processes relative to human behavior.

Computational approaches which aim at explaining human reasoning should be *cognitively adequate*. Usually, the concept of adequacy is measured by distinguishing between conceptual and inferential adequacy [1]. *Conceptual adequacy* deals with the representational part of the system. The aim is to have a representation of the given information such that it captures the structure of

* {dietz,sh}@iccl.tu-dresden.de
** ragni@cognition.uni-freiburg.de

how it appears in human knowledge. *Inferential adequacy* measures whether the computations behave similarly to human reasoning.

Accordingly, Stenning and van Lambalgen [2] argue that human reasoning should be modeled by, first, reasoning towards an appropriate representation and, second, by reasoning with respect to this representation. As appropriate representation for the *suppression task*, Stenning and van Lambalgen propose logic programs under completion semantics based on the three-valued logic used by Fitting [3]. Hölldobler and Kencana Ramli [4] have shown that this approach contains some mistakes but can be corrected by proposing the three-valued Łukasiewicz [5] logic.

Based on [6] which follows the approach of [4] and models the suppression task as logic programs together with their weak completion, we apply this formalization to the *Wason selection task* [7]. The following section explains the Wason selection task in detail. After that, we give the necessary definitions for the formalization of this task which is then presented in Section 4. The last section discusses implications.

2 Wason Selection Task

The Wason selection task was first published in [7], where subjects had to check a given conditional statement on some instances. If the problem was presented as a rather abstract description then almost all subjects made the same logical mistakes. Griggs and Cox [8] developed an isomorphic representation of the problem in a social context, and surprisingly almost all of the subjects solved this task logically correctly.

The Abstract Case Consider the conditional

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

and four cards on a table showing the letters D and F as well as the numbers 3 and 7. Furthermore, we know that each card has a letter on one side and a number on the other side. Which cards must be turned to prove that the conditional holds? Assume the conditional is represented in classical propositional logic by the implication

$$3 \leftarrow D, \tag{1}$$

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 the cards showing D and 7. However, as repeated experiments have shown consistently (see Table 1), subjects believe differently.

[1] We prefer to write implications in the form *conclusion if condition* as it is common in logic programming.

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

Table 1. The results of the abstract case of the Wason selection task.

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

Table 2. The results of the social case of the Wason selection task.

Whereas 89% of the subjects correctly determine that the card showing D must be turned (a number other than 3 on the other side would falsify the implication), 62% of the subjects incorrectly suggests to turn the card showing 3 (no relevant information can be found which would falsify the implication). Likewise, whereas only 25% of the subjects correctly believe that the card showing 7 need to be turned (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 (no relevant information can be found which would falsify the implication). In other words, the overall correctness of the answers for the abstract selection task if modeled by an implication in classical two-valued logic is pretty bad.

The Social Case Griggs and Cox [8] adapted Wason selection task to a social case. Consider the conditional

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

and again consider four cards, where on one side there is the person's age and on the other side of the card what the person is drinking: *drinking beer*, *drinking coke*, *22 years old* and *16 years old*. Which drinks and persons must be checked to prove that the conditional holds? If the conditional is represented by the implication

$$o \leftarrow b, \tag{2}$$

where o represents a person being older than 19 years and b represents the person drinking a beer, then in order to verify the implication one must turn the cards *drinking a beer* and *16 years of age*. Subjects usually solve the social version of the selection task correctly. Table 2 shows the results represented in [8] for the social case.

The Problem Is there a formalization which adequately models the answers provided by subjects on the abstract as well as on the social case of the task?

In the last chapter of [2], Stenning and van Lambalgen give a detailed overview of various explanations for the problem addressed. Wason [7] 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 for further reasoning. Evans [9] describes 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 not 3 is on the card and therefore not D has to be on the other side. Most people would give the correct conclusion when explicitly generating an impossible situation: If there is D on one side and there is not 3 on the other side, then false. Why do people not make these mistakes in the social case?

3 Preliminaries

We define the necessary notations we will use throughout this paper and restrict ourselves to propositional logic as this is sufficient to solve the selection task.

3.1 Licenses for Implications

As already mentioned in the introduction, Stenning and van Lambalgen distinguish between two steps when modeling human reasoning. We adopt the first step, in particular, the idea to represent conditionals by licenses for implications. 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

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

instead of (1) and

$$o \leftarrow b \wedge \neg ab_2 \tag{4}$$

instead of (2), where $\neg ab_1$ and $\neg ab_2$ are used to express that the corresponding rules hold unless there are some abnormalities.

3.2 Logic Programs

A *logic program* $\mathcal{P}$ is a finite set of expressions of the form $A \leftarrow B_1 \wedge \ldots \wedge B_n$, where $n \geq 1$, A is an atom, and each B_i, $1 \leq i \leq n$, is either a literal, $\top$, or $\bot$. A is called *head* and $B_1 \wedge \ldots \wedge B_n$ is called *body* of the clause. A clause of the form $A \leftarrow \top$ is called *positive fact*, whereas a clause of the form $A \leftarrow \bot$ is called *negative fact*.

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

1. All clauses with the same head $A \leftarrow body_1,\ A \leftarrow body_2,\ \ldots$ are replaced by $A \leftarrow body_1 \vee body_2 \vee \ldots$.
2. If an atom A is not the head of any clause in $\mathcal{P}$ then add $A \leftarrow \bot$.
3. All occurrences of $\leftarrow$ are replaced by $\leftrightarrow$.

The resulting set is called *completion* of $\mathcal{P}$ ($\mathsf{c}\,\mathcal{P}$). If step 2 is omitted, then the resulting set is called *weak completion* of $\mathcal{P}$ ($\mathsf{wc}\,\mathcal{P}$).

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

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

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

$\leftarrow_L$	$\top$	U	$\bot$
$\top$	$\top$	$\top$	$\top$
U	U	$\top$	$\top$
$\bot$	$\bot$	U	$\top$

$\leftrightarrow_L$	$\top$	U	$\bot$
$\top$	$\top$	U	$\bot$
U	U	$\top$	U
$\bot$	$\bot$	U	$\top$

Table 3. The truth tables of the three-valued Łukasiewicz logic.

3.3 Three-Valued Logics

In Table 3 the truth tables of the three-valued Łukasiewicz logic [5] are depicted, where $\top$, $\bot$, and U denote *true*, *false*, and *unknown*, respectively. Based on these truth tables the notions of logical equivalence $\equiv_{3\mathrm{L}}$ and logical consequence $\models_L$ can be defined in the usual way. One should also note that the replacement theorem holds for the Łukasiewicz logic as well, i.e. a subformula of a formula can be replaced by an equivalent one without changing the semantics of the formula. We will represent three-valued interpretations by tuples of the form $\langle I^\top, I^\bot \rangle$, where $I^\top$ contains all atoms which are mapped to $\top$, $I^\bot$ contains all atoms which are mapped to $\bot$, $I^\top$ and $I^\bot$ are disjoint, and all atoms which occur neither in $I^\top$ nor in $I^\bot$ are mapped to U. A *model* for $\mathcal{P}$ is an interpretation I where each clause occurring in $\mathcal{P}$ is mapped to $\top$.

3.4 Computing Least Models

Stenning and van Lambalgen [2] devised an operator to compute the least fixed point for programs discussed herein:

Let I be an interpretation in $\Phi_\mathcal{P}(I) = \langle J^\top, J^\bot \rangle$, where

$$J^\top = \{A \mid \text{there exists } A \leftarrow body \in \mathcal{P} \text{ with } I(body) = \top\},$$
$$J^\bot = \{A \mid \text{there exists } A \leftarrow body \in \mathcal{P} \text{ and for all } A \leftarrow body \in \mathcal{P} \text{ we find } I(body) = \bot\}.$$

Hölldobler and Kencana Ramli [4] have shown that the least fixed point of $\Phi_\mathcal{P}$ is identical to the *least model of the weak completion* ($\mathsf{lm}_{Ł}\mathsf{wc}$) of $\mathcal{P}$ and can be computed by iterating $\Phi_\mathcal{P}$ starting with the empty interpretation $I = \langle \emptyset, \emptyset \rangle$.

As shown in [4], programs as well as their weak completions admit the model intersection property under the Łukasiewicz logic and, hence, each weakly completed program $\mathsf{wc}\,\mathcal{P}$ has a least model. Consider the social case where the person is drinking beer and nothing abnormal is known, that is $\mathcal{P}_{beer} = \{ab_2 \leftarrow \bot, b \leftarrow \top\}$. The least fixed point of $\Phi_{\mathcal{P}_{beer}}$ is computed starting with interpretation $I_0 = \langle \emptyset, \emptyset \rangle$:

$$I_1 = \Phi_{\mathcal{P}_{beer}}(I_0) = \langle \{b\}, \{ab_2\} \rangle = \Phi_{\mathcal{P}_{beer}}(I_1)$$

where $\langle \{b\}, \{ab_2\} \rangle$ is the least model of the weak completion of $\mathcal{P}_{beer}$ under Łukasiewicz logic.

3.5 Abduction

Following [10] we consider an *abductive framework* consisting of a program $\mathcal{P}$ as knowledge base, a set $\mathcal{A}$ of abducibles consisting of the (positive and negative) facts for each undefined predicate symbol in $\mathcal{P}$ and the logical consequence relation $\models_{\text{Ł}}^{\mathsf{lm\,wc}}$, where A is *undefined* in $\mathcal{P}$ if and only if $\mathcal{P}$ does not contain a clause of the form $A \leftarrow \mathit{body}$ and $\mathcal{P} \models_{\text{Ł}}^{\mathsf{lm\,wc}} F$ if and only if $\mathsf{lm}_{\text{Ł}}\mathsf{wc}\,\mathcal{P}(F) = \top$ for the formula F. As *observations* we consider literals.

Let $\langle \mathcal{P}, \mathcal{A}, \models_{\text{Ł}}^{\mathsf{lm\,wc}} \rangle$ be an abductive framework and $\mathcal{O}$ an observation. $\mathcal{O}$ is *explained* by $\mathcal{E}$ if and only if $\mathcal{E} \subseteq \mathcal{A}$, $\mathcal{P} \cup \mathcal{E}$ is satisfiable, and $\mathcal{P} \cup \mathcal{E} \models_{\text{Ł}}^{\mathsf{lm\,wc}} \mathcal{O}$. Usually, minimal explanations are preferred. In case there exist several minimal explanations, then two forms of reasoning can be distinguished. F follows *sceptically* from program $\mathcal{P}$ and observation $\mathcal{O}$ ($\mathcal{P}, \mathcal{O} \models_s F$) if and only if $\mathcal{O}$ can be explained and for all minimal explanations $\mathcal{E}$ we find $\mathcal{P} \cup \mathcal{E} \models_{\text{Ł}}^{\mathsf{lm\,wc}} \mathcal{O}$, whereas F follows *credulously* from $\mathcal{P}$ and $\mathcal{O}$ ($\mathcal{P}, \mathcal{O} \models_c F$) if and only if there exists a minimal explanation $\mathcal{E}$ such that $\mathcal{P} \cup \mathcal{E} \models_{\text{Ł}}^{\mathsf{lm\,wc}} \mathcal{O}$.

4 Modeling the Selection Task

According to Kowalski [11], people view the conditional in the abstract case as a *belief*. For instance, the subjects perceive the task to examine whether the rule is either true or false. On the other hand, in the social case, the subjects 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.

In [12] psychological experiments with several variations of the so called *abstract deontic selection task* are carried out. The authors show that the performance of the abstract case can significantly be improved when introducing a deontic notion. This seems to support Kowalski's interpretation, which we will adopt in the following and model our formalism accordingly.

The Social Task In this case most humans are quite familiar with the conditional as it is a standard law. They are also aware – it is common sense knowledge – that there are no exceptions or abnormalities and, hence, ab_2 is set to $\bot$.

Let us assume that conditional (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 in the case *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 (4), as well. Overall, in the cases *drinking beer* and *16 years old* the social constraint (4) is not entailed by the least model of the weak completion of the program. Hence, we need to check these cases out and, hopefully,

case	program $\mathcal{P}$	$\mathsf{lm}_{Ł}\,\mathsf{wc}\,\mathcal{P}$		
drinking beer	$\{ab_2 \leftarrow \bot, b \leftarrow \top\}$	$\langle\{b\},\{ab_2\}\rangle$	$\not\models_L$	(4)
drinking coke	$\{ab_2 \leftarrow \bot, b \leftarrow \bot\}$	$\langle\emptyset,\{b,ab_2\}\rangle$	$\models_L$	(4)
16 years old	$\{ab_2 \leftarrow \bot, o \leftarrow \bot\}$	$\langle\emptyset,\{o,ab_2\}\rangle$	$\not\models_L$	(4)
22 years old	$\{ab_2 \leftarrow \bot, o \leftarrow \top\}$	$\langle\{o\},\{ab_2\}\rangle$	$\models_L$	(4)

Table 4. The computational logic approach for the social case of the selection task.

observation $\mathcal{O}$	explanation $\mathcal{E}$	$\mathsf{lm}_{Ł}\,\mathsf{wc}\,(\mathcal{P}\cup\mathcal{E})$		
D	$\{D \leftarrow \top\}$	$\langle\{D,3\},\{ab_1\}\rangle$	$\rightsquigarrow$	turn
F	$\{F \leftarrow \top\}$	$\langle\{F\},\{ab_1\}\rangle$	$\rightsquigarrow$	no turn
3	$\{D \leftarrow \top\}$	$\langle\{D,3\},\{ab_1\}\rangle$	$\rightsquigarrow$	turn
7	$\{7 \leftarrow \top\}$	$\langle\{7\},\{ab_1\}\rangle$	$\rightsquigarrow$	no turn

Table 5. The computational logic approach for the abstract case of the selection task.

find that the beer drinker is older than 19 and that the 16 years old is not drinking beer.

The Abstract Task In this case the conditional is artificial. There is no common sense knowledge about such a case.

Let us assume that conditional (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

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

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 Łukasiewicz semantics. Unfortunately, this least model does not explain any symbol on any card. In order to explain an observed card, we need to consider abduction, where the set of abducibles is

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

Remember, the set of abducibles is defined as all undefined facts. Now consider the four different cases, where the explanations $\mathcal{E}$ are minimal and basic. In the cases where F or 7 were observed, the least model of the weak completion of $\mathcal{P}\cup\mathcal{E}$ simply confirms the observation; no further action is needed. In the case where D was observed, the least model maps also 3 to $\top$; however, this can only be confirmed if the card showing D is turned. Likewise, in the case where 3 is observed, D is also mapped to $\top$ in the least model, which can only be confirmed if the card is turned.

5 Conclusion

We have presented a computational logic approach for modeling human reasoning in the Wason selection task. It is based on a previously proposed approach that adequately models another psychological study, the suppression task. We extended our approach with an idea from Kowalski's task representation: in order to solve the social case correctly, the conditional is 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 sceptical reasoning within an abductive framework. Taken together, the use of a non-monotonic ternary logic seems to be more appropriate for human reasoning (especially in explaining human decisions) than classical logic approaches.

References

1. Strube, G.: Wörterbuch der Kognitionswissenschaft. Klett-Cotta (1996)
2. Stenning, K., Lambalgen, M.: Human reasoning and cognitive science. Bradford Books. MIT Press (2008)
3. Fitting, M.: A kripke-kleene semantics for logic programs. The Journal of Logic Programming **2** (1985)
4. Hölldobler, S., Ramli, C.K.: Logic programs under three-valued lukasiewicz semantics. In Hill, P.M., Warren, D.S., eds.: ICLP. Volume 5649 of Lecture Notes in Computer Science., Springer (2009)
5. Łukasiewicz, J.: O logice trójwartościowej. Ruch Filozoficzny **5** (1920) 169–171 English translation: On Three-Valued Logic. In: *Jan Łukasiewicz Selected Works*. (L. Borkowski, ed.), North Holland, 87-88, 1990.
6. Dietz, E.A., Hölldobler, S., Ragni, M.: A Computational Approach to the Suppression Task. to appear in Proceedings of the 34th Cognitive Science Conference (2012)
7. Wason, P.: Reasoning about a rule. Quarterly Journal of Experimental Psychology **20** (1968) 273–281
8. Griggs, R., Cox, J.: The elusive thematic materials effect in the wason selection task. British Journal of Psychology **73** (1982) 407–420
9. Evans, J.: Interpretation and matching bias in a reasoning task. British Journal of Psychology **24** (1972) 193–199
10. Kakas, A.C., Kowalski, R.A., Toni, F.: Abductive logic programming. Journal of Logic and Computation **2** (1993) 719–770
11. Kowalski, R.: Computational Logic and Human Thinking: How to be Artificially Intelligent. 1 edn. Cambridge University Press (2011)
12. Beller, S., Bender, A.: Competent deontic reasoning: The abstract deontic selection task revisited. In Miyake, N., Peebles, D., Cooper, R.P., eds.: Proceedings of the 34th Annual Conference of the Cognitive Science Society, Austin, TX: Cognitive Science Society (2012) 114 – 119

MoralLISA – An Extension of the Analogy Model LISA for Moral Decision Making

Peter Großmann and Michael Siebers and Ute Schmid*

Faculty Information Systems and Applied Computer Science, University of Bamberg

Abstract. A well known empirical finding in moral decision making is that moral judgements often are guided by so called sacred values and that such values can be specific to a certain cultural background. We present preliminary work where we explore an extension of the analogy model LISA to model moral decision making based on sacred values. The model is applied to stories used in a psychological experiment.

1 Introduction

In decision making, typically it is pesupposed that an agent selects an action which satisfies his/her goals best. Moral decision making takes into account criteria such as kindness and fairness. As in general decision making, psychological theories of moral decision making are often modelled in an utilitarian, rational framework – that is, it is assumed that a decision is the outcome of a reasoning process (Waldmann, Nagel, & Wiegmann, 2012). In contrast, there are deontological approaches to explain moral decision making which highlight the emotional and intuitive aspect of decision making (Paxton & Greene, 2010).

There exist some computational models of moral decision making: ACORDA (Pereira & Saptawijaya, 2011) is a an example of a computational approach within the utilitarian framework. The system is realized in PROLOG using prospective logic. Wallach, Franklin, and Allen (2010) use the artificial general intelligence system LIDA to take into account rational as well as emotional aspects of moral decision making.

A special aspect of moral decisions is that they can be strongly dominated by so called sacred values (Waldmann et al., 2012; Tetlock, 2003), that is, values which people find impossible to violate. Such values as "do not kill a human being" are often researched in the context of moral dilematta such as the trolley dilemma (Hauser, Cushman, Young, Jin, & Mikhail, 2007). In this scenario, a trolley is threatening to kill five people.

* *Moral*LISA was realized by Peter Großmann in his master thesis, supervised by Ute Schmid. Corresponding author: Ute Schmid, ute.schmid@uni-bamberg.de.

If this can be prevented by redirecting the trolley to another track where one person is standing and gets killed, subjects will decide to change the track and save the five people. However, if the only solution is to push one person in front of the trolley, subjects will avoid this act – although in both alternatives one person gets killed.

There is some evidence that sacred values are specific to culture (Tetlock, 2003). Dehghani, Gentner, Forbus, Ekhtiari, and Sachdeva (2009) argue that typical stories which are known to everybody in a specific cultural setting have a strong impact on moral decision making when sacred values are involved. The authors propose that such stories serve as base analogies and that moral decisions are felled based on analogical mapping and transfer from such cultural narratives to new situations. Furthermore, Dehghani (2009) proposed a cognitive model – MoralDM – for analogy based moral decision making which is based on the structure-mapping engine (SME, Falkenhainer, Forbus, & Gentner, 1989).

However, because the analogy process of SME is based on syntactical structural similarity, sacred values need to be dealt with in a different module. Nevertheless, in our opinion, analogical reasoning is a convincing approach to explain how sacred values guide the mental process of moral decision making which is currently not taken into account in computational models of machine ethics (Pereira & Saptawijaya, 2011; Wallach et al., 2010). Therefore, we investigated an alternative cognitive model for analogical reasoning, namely the hybrid system LISA (Hummel & Holyoak, 1997, 2003). In LISA, nodes of the base and target structure are mapped by an activation spreading mechanism. The structual information is complemented by semantic units which are shared by base and target. We believe that LISA offers a more natural architecture to deal with sacred values in analogy-based moral decision making.

In the following, we first describe the analogy model Lisa. Afterwards we introduce our extension *Moral*Lisa. Then we present the moral decision problems investigated by Dehghani et al. (2009) and we show how *Moral*Lisa can be applied to these problems.

2 The Analogy System LISA

In standard reasoning, a chain of rules is applied to derive a conclusion. In contrast, in analogical reasoning a new (target) problem is mapped with a known base problem and information from the base is transfered to the target. That is, inference is realized by transfer. Analogical reasoning is based on structured representations – typically terms or directed

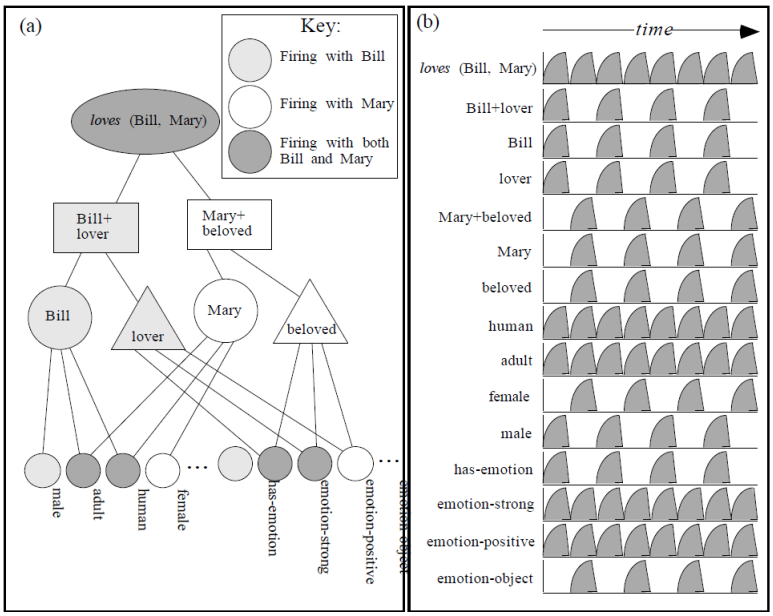


Fig. 1. (a) Illustration of the representation of *loves (Bill, Mary)* in LISA. (b) A time-series illustration of the representation of *loves (Bill, Mary)*. Each graph corresponds to one unit (a). The abscissa of the graph represents time and the ordinate represents the corresponding units activation.(Figure 3 from Hummel & Holyoak, 2003)

acyclic graphs. Nodes between base and target are mapped in a structure perserving way. If a node in the base is connected to a substructure which has non correspondence in the target, this substructure is carried over.

There are different cognitive models of analogical reasoning: The structure mapping engine (SME, Falkenhainer et al., 1989) is one of the most well-known systems. An alternative approach is LISA which is a hybrid system combining structural representations with semantic features.[1]

Representations in LISA are based on propositions and consist of four kinds of entities (see Figure 1a): (1) A proposition such as *loves(Bill, Mary)* is composed of (2) bindings between roles and role-fillers ("sub-propositions"). For example, *Bill* is the filler of the role *lover*. These bindings are constructed over (3) objects (such as *Bill*) and relations (such as *lover*). Finally, (4) there are sematic units which represent features of objects and relations (such as *male* for *Bill* and *emotion-strong* for *lover*).

[1] A Python implementation of LISA by John Hummel is available at http://internal.psychology.illinois.edu/ jehummel/models.php.

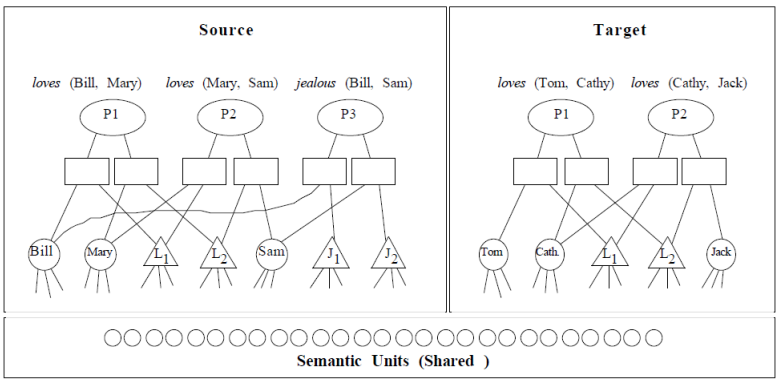


Fig. 2. LISAs representations of two analogs. The source states that Bill loves Mary (P1), Mary loves Sam (P2), and Bill is jealous of Sam (P3). The target states that Tom loves Cathy (P1) and Cathy loves Jack (P2). (Figure 5 from Hummel & Holyoak, 2003)

Entities between adjoining levels share bidirectional excitatory connections.

LISA allows higher-order relations to represent more complex statements. For example, *knows (Mary, loves (Bill, Mary))* can be represented as top-level proposition with the role filler *Mary* and the role *Knower* and the proposition *loves(Bill, Mary)* as further role filler and role *know*. That is, the proposition *loves(Bill, Mary)* is object for the higher-order expression and simultaneously a proposition which is further decomposed as given in Figure 1a.

Nodes in a representation are synchronous are asynchronous activated over discrete time steps (see Figure 1b): When a filler is bound to a role, the connected elements fire in synchrony with one another. For example, *male* fires with *Bill*, *adult* fires with *Bill* and *Mary*.

For a base ("driver") and target ("recipient") analog mapping connections are established between nodes of corresponding type (that is roles can only be mapped to roles, objects to objects). If nodes in the base and target representation are activated simultaneously, the weight of the mapping connection is incremented, otherwise decremented (see Figure 2). The process starts with random or specifically set values for the driver analog and is run over a fixed number of time steps. When processing is finished, the connections with heigh positive values constitute the mapping hypothesis.

The weights are updated by a simple Hebbian rule $\Delta h_{ij} = a_i \times a_j$ with a_i as activation of a unit i in the driver (base) and a_j as activation of unit j in the recipient (target).

Finally, transfer from base to target is realised by "self-supervised" learning: If a subset of nodes from the driver has no correspondence to nodes in the target, all recipient nodes which are mapped to other driver nodes are inhibited and no other nodes will be excited. This is used as a cue to create new entities in the target. For example, in Figure 2, LISA infers that Sally will be jealous of Cathy.

3 MoralLISA

An obvious way to apply LISA to moral decision making including sacred values is the following: (1) Representing a well-known culture specific narrative as a propositional base representation, (2) representing a new moral decision making problem as propositional target representation, and (3) representing the sacred values as semantic features. Because the sacred values should strongly influence the decision, we planned simply to initialize them with high activation values. In principle, LISA allows that specific units are initialized with specific activation values at starting time. However, the available implementation of LISA resets the activations the standard values during mapping.

A similar effect can be obtained by increasing the weights of mapping relations when sacred values are involved. Therefore, we modified the formula which LISA uses to calculate the increase of the weight of a mapping relation (see above) to

$$\Delta h_{ij} = a_i \times a_j + s \times \gamma$$

where a_i and a_j are the current activations of a unit i in the base and a unit j in the target, s is the number of shared sacred values and $\gamma \in [0, \ldots, 1]$ modifies the influence of the sacred values. For our test setttings we set $\gamma = 0.1$.

Besides the inclusion of sacred values, we extended LISA to identify synonymous expressions for the semantic units. This allows a more natural transfer of stories represented in natural language to the LISA representation.

4 To Lose or Not to Lose a Wrestling Match

Dehghani et al. (2009) used the following story which is very well known in the Iranian culture as a base problem:

> Pourya Vali was the most famous wrestler of his time. The morning before wrestling with a young athlete from another province, he goes to a mosque and sees the mother of the young athlete praying and saying "God, my son is going to wrestle with Pourya Vali. Please watch over him and help him win the match so he can use the prize money to buy a house". Pourya Vali thinks to himself that the young wrestler needs the money more than he does, and also winning the match will break the heart of the old mother. He has two choices, he can either win the match and keep his status as the best wrestler in the world or he could lose the match and make the old mother happy. Even though he was known not to ever lose a match, he loses that one on purpose.

In an experiment with students in Iran and students in North-America he presupposed that the Iranian students are all acquainted with this story while the American students are not. He presented to each subject one of four target problems:

Surface Change: wrestling → ping-pong, house → marriage,
Structure Change: house → expensive clothes,
Surface and Structure Change: wrestling → ping-pong, house → expensive clothes,
Sacred Value Change: not Pourya Vali, but his opponent is the most famous wrestler.

When presented with the story which has a change on the surface, significantly more Iranian than North-American students answered that Pourya Vali should lose on purpose. The authors explain these findings by analogy making between the well-known base-story in Iran and the target for the Iranian students. The North-American students either have their own base story embedding the values of "fair play" and "the best may win" or use a rule-based reasoning approach to reach their decision.

5 Reasoning with Sacred Values in MoralLISA

We presented the base problem and the four different target problems to *Moral*LISA. The base problem consists of three relations: *praying(Mother)*, *fighting(Vali, Son)* and *losing(Vali)*. The third relation is modelled as effect of the other relations. Instead of a graphical representation as above we give a representation in the LISA syntax as illustration in Figure 3. The target stories only contain two relations. The third relation should be inferred by *Moral*LISA.

In correspondence to the empirical findings, *Moral*LISA only inferred that Vali should deliberately lose the match for the target with the surface

```
Analog ValiBase
  Defpreds
    Praying 1 positive;
    Fighting 2 harmful;
    Losing 1 altruistic;
  end;
  Defobjs
    Vali wrestler famous;
    Son wrestler notfamous;
    Mother sympathetic;
  end;
  DefProps
    P1 Praying ( Mother );
    P2 Fighting (Vali Son);
    P3 Losing (Vali);
  end;
  DefGroups
    G1 Props: P1 P2;
       Semantics: cause;
    G2 Props: P3;
       Semantics: effect;
    G3 Groups: G1 G2;
       Semantics: cause-relation
  end;
done;
```

Fig. 3. Representation of the Base Story

change. Furthermore, *Moral*LISA also could infer the third relation when the semantic features in the base were synonyms such as *famous/notable*, *harmful/adverse*, *sympathetic/likeable*.

6 Conclusions

We presented a simple extension of the analogy system LISA to model moral decision making based on sacred values. The hybrid LISA architecture combines the structure sensitivity of symbolic approaches with the flexibility of connectionist approaches. It allows to include the impact of sacred values in a natural way. Guidance by sacred values, however, is only one aspect of moral decision making. A more comprehensive cognitive model needs to take into account additional reasoning strategies such as rule-based and heuristic approaches (Waldmann et al., 2012; Paxton & Greene, 2010).

References

Dehghani, M. (2009). *A cognitive model of recognition-based moral decision making.* Unpublished doctoral dissertation, Northwestern University, Department of Electrical Engineering and Computer Science, Evanston, IL.

Dehghani, M., Gentner, D., Forbus, K., Ekhtiari, H., & Sachdeva, S. (2009). Analogy and moral decision making. In B. Kokinov, K. Holyoak, & D. Gentner (Eds.), *Proceedings of the second international conference on analogy.* Sofia, Bulgaria: NBU Press.

Falkenhainer, B., Forbus, K., & Gentner, D. (1989). The structure mapping engine: Algorithm and example. *Artificial Intelligence*, *41*, 1-63.

Hauser, M. D., Cushman, F. A., Young, L., Jin, R., & Mikhail, J. M. (2007). A dissociation between moral judgment and justifi cation. *Mind and Language*, *22*, 121.

Hummel, J., & Holyoak, K. (1997). Distributed representation of structure: A theory of analogical access and mapping. *Psychological Review*, *104*(3), 427-466.

Hummel, J. E., & Holyoak, K. J. (2003). Relational reasoning in a neurally-plausible cognitive architecture: An overview of the LISA project. *Cognitive Studies: Bulletin of the Japanese Cognitive Science Society*, *10*, 58-75.

Paxton, J. M., & Greene, J. D. (2010). Moral reasoning: Hints and allegations. *Topics in Cognitive Science*, *2*(3), 511-527.

Pereira, L. M., & Saptawijaya, A. (2011). Modelling morality with prospective logic. In M. Anderson & S. L. Anderson (Eds.), *Machine ethics* (p. 398-421). Cambridge University Press.

Tetlock, P. E. (2003). Thinking the unthinkable: Sacred values and taboo cognitions. *Trends in Cognitive Sciences*, *7*, 320324.

Waldmann, M. R., Nagel, J., & Wiegmann, A. (2012). Moral judgment. In K. J. Holyoak & R. G. Morrison (Eds.), *The Oxford Handbook of Thinking and Reasoning* (p. 364-389). Oxford University Press.

Wallach, W., Franklin, S., & Allen, C. (2010). A conceptual and computational model of moral decision making in human and artificial agents. *Topics in Cognitive Science*, *2*(3), 454485.

Pouring and Mixing Liquids — Understanding the Physical Effects of Everyday Robot Manipulation Actions

Reinhard Klapfer[1], Lars Kunze[1], and Michael Beetz[2]

[1] Intelligent Autonomous Systems, Technische Universität München, Germany
kunzel@cs.tum.edu
[2] Artificial Intelligence, University of Bremen, Germany
beetz@tzi.de

Abstract. Humans have an excellent ability to reason about the consequences of their own actions which often prevents them from ending up in undesired situations. If robot assistants are to accomplish everyday manipulation tasks like humans, they should be equipped with a similar capability for making temporal projections. In this paper, we investigate how robots can infer the effects of their own actions, in particular when dealing with liquids. The proposed system allows robots to determine the appropriate action parameters by envisioning the outcome of their own actions based on detailed physics-based simulations.

1 Introduction

Modern robotics attempts to go beyond simple pick-and-place scenarios and equip robots with the capabilities for executing more complex tasks like, for example, making pancakes. While preparing pancakes, ingredients for the dough have to be mixed, stirred and poured into a pan. Accomplishing such everyday manipulation tasks successfully requires robots to predict the consequences of their parameterized actions and to reason about them. In this work we place our emphasis on robots performing everyday manipulation tasks which involve the handling of liquids. For example, a robot about to pour a pancake mix onto a pancake maker has to decide where to hold the container, at what angle and for how long without spilling something.

Humans are able to reason about these physical processes and to estimate the right *parameterization* intuitively based on both their experiences and common sense. Understanding everyday physical phenomena, that is representing and reasoning about them, is an endeavor in the field of Artificial Intelligence which dates at least back to the work of Hayes [5]. More recently, there has been work on physical reasoning problems like "Cracking an egg" ([10]) which is listed on the common sense problem page[3]. In [1], Davis presents a formal solution to the problem of pouring liquids and in his work on the representation of matter [2], he investigated the advantages and disadvantages of various representations

[3] http://www.commonsensereasoning.org/problem_page.html

including those for liquids. In [3], he says that it is tempting to use simulations for spatial and physical reasoning. But he argues that simulations are not suitable for the interpretation of natural language texts because many entities in texts are underspecified. However, in the context of robotics entities in the environment can often be sufficiently recognized and represented using internal models.

In this work, we build on the concept of simulation-based temporal projections as proposed in [8, 7]. Everyday robot manipulation tasks are simulated with varying parameterizations, world states of the simulation are monitored and logged, and the resulting logs are translated into a first-order representations called timelines. These timelines are then used to answer logical queries on the resulting data structures in order to investigate an understanding of the robot of the executed task. The main contribution of this work is the design and implementation of data structures and algorithms for representing and reasoning about liquids within this framework.

2 Simulation-Based Temporal Projection Framework

This section gives a brief overview of the simulation-based temporal projection framework introduced in [8, 7] and describes the contributions of this work.

The overall framework is depicted in Figure 1. It is based on state-of-the-art technologies such as ROS[4], the Gazebo simulator[5] and the point cloud library PCL[6]. Within the simulation (a) a robot can freely navigate and interact with objects. The behavior of the robot is specified in a robot control program (b). In the simulator we represent, e.g., a pancake mix as particles using the data structures of Gazebo. Given that we are in particularly interested in analyzing the behavior of liquids we group the simulated particles by an Euclidean clustering technique. Having obtained information of clusters makes it possible to reason about the fusion or division of volumes or chunks of liquids. The clustering is realized as node located at an intermediate layer (c). As Gazebo uses ODE which is only capable of dealing with rigid bodies a simulation of liquids in the simulator is only an approximation. Therefore we use the information about the clusters to initialize a more accurate simulation of liquids by considering physical aspects such as diffusion and convection (d). The robot's actions, its interactions with the objects, the state of the liquid, the clusters and the state of the environment (world) are crucial information for the reasoning framework. In order to process this knowledge in a subsequent processing step, controllers and monitors are used to access the data. A monitor (e) is a set of programs that listen (subscribe) to a topic published by various components. For example, the information about the world state is retrieved by accessing a topic published by a controller attached to the description of the world and logged by a monitor (f,g). These logs are then translated into interval-based first-order representation, called timelines, which can be processed by the logical programming language Prolog (h) and are then used to identify failures or success scenarios of the robot simulation.

[4] `http://www.ros.org`
[5] `http://www.gazebosim.org`
[6] `http://www.pointclouds.org`

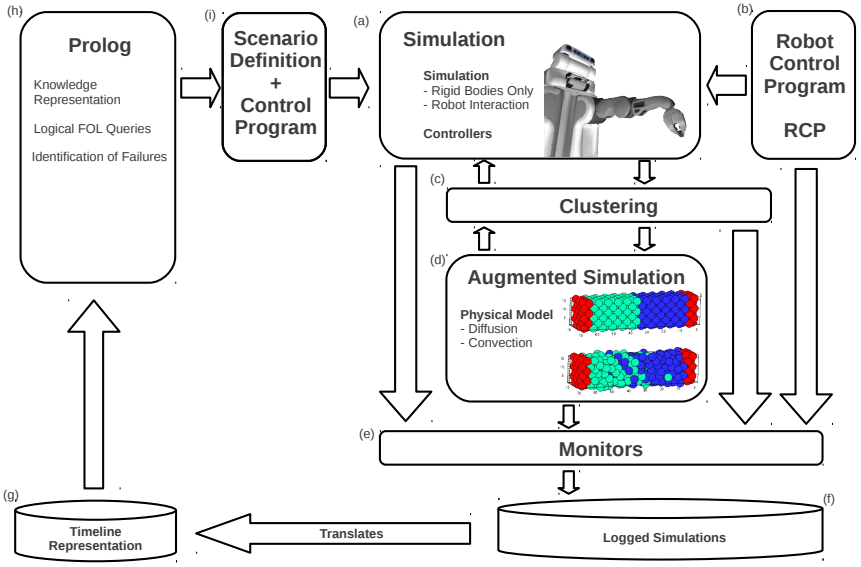


Fig. 1. The simulation-based temporal projection framework.

3 Simulation of Liquids

Simulating liquids is of great interest in physics and chemistry. As some processes occur very fast, events might not be observable in all its details in reality. The purpose of simulating liquids in our work is to observe the impact of the robot's action with respect to the liquid's behavior, which is of great importance when, e.g., mixing liquids. Different approaches have been incorporated when trying to simulate liquids depending on the required level of accuracy needed. In this work we propose two complementary approaches for simulating liquids, (1) as a graph-based model similar to [6] and (2) as a Monte-Carlo simulation for modeling diffusion and convection [4]. Both do not simulate a liquid in all its exactness but provide enough information for making logical inference about qualitative phenomena.

3.1 Representing the Pancake Mix as Graph-based Model

The model for the pancake mix (or liquid) was adapted from the work of Johnston in [6]. Originally, it was designed to simulate a wide range of physical phenomena including diverse domains such as physical solids or liquids as hypergraphs where each vertex and edge is annotated with a frame that is bound to a clock and linked to update rules that respond to discrete-time variants of Newton's laws of mechanics. Our pancake mix model can be in two states: first, the mix is liquid, and second, the mix becomes a deformable pancake after cooking. In the Gazebo simulator, we use a set of particles in the form of a sphere with an

associated diameter, mass and visual definition. The benefit of this model is that it is realized as graph with no connection between the vertices. This means that the individual particles could move freely to some extent. This was useful for performing the pouring task. Due to the fact of the particles not being connected with joints, the simulated liquid can be poured over the pancake maker where it dispenses due to its round shape. A controller was attached to the spheres that applies small forces to the particles in order to simulate the viscosity of the pancake mix. Currently, we do not consider heat as the trigger of transforming the liquid to a solid pancake but simply assume the event to occur after constant time. We identified all particles on the pancake maker and created the pancake based on a graph traversal algorithm starting at the cluster center.

Clustering of Particles The analysis of the behavior and contact information of liquids in an everyday manipulation task is of particular interest, as one might use this dynamic change of a liquid to reason about possible causes of failures. For example if a liquid is classified as one single cluster we can assume that this liquid represents one volume. Now let us assume the state of the system changes and from that one cluster we obtain two, meaning an event has occurred that has caused the volume to be split. We have designed the behavior of the clustering strategy (Euclidean Clustering) as illustrated in Figure 2.

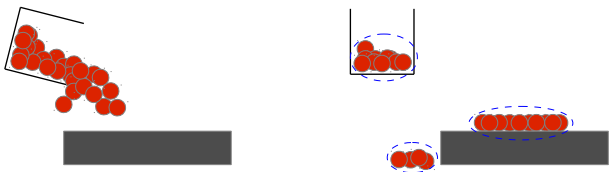


Fig. 2. Basic idea of the clustering approach: during simulation we identify clusters of particles. For example, after pouring, one cluster resides still in the mug, a second is on the pancake maker and a third is spilled onto the table. We are able to extract information including contacts, position, extension, and size of the individual clusters.

3.2 A More Sophisticated Model

Deformable bodies are seen as a big challenge in simulation and usually require a lot of computational power. The physical simulation approach [4] uses a Monte-Carlo process to simulate diffusion of liquids. Molecular movement is either provoked from heat or from a difference in potential. The rate of change depends on the diffusion coefficient and its respective change. This is a well known concept in physics described by equation 1 and denoted as the *macroscopic diffusion* equation or *Fick's second law* of diffusion. This differential equation takes into consideration a change of concentration over time.

$$\frac{\partial C}{\partial t} = D \cdot \frac{\partial^2 C}{\partial^2 t} \tag{1}$$

It can be shown [4] that *Random Walk* gives one particular solution for the above partial differential equation. Motivated by this idea we applied Algorithm 1 proposed by Frenkel et al. to simulate this physical effect. The algorithm follows the Metropolis scheme and uses a probability function to decide if a particle is going to be displaced or not. The Leonnard-Jones Potential Function (Equation 2) was

Algorithm 1 Metropolis Scheme.

1) Select a particle r at random and calculate its energy potential $U(r^N)$
2) Give the particle a random displacement, $r' = r + \Delta$
3) Calculate the new energy potential $U(r'^N)$
3) Accept the move from state r^N to r'^N with probability

$$\text{acc}(\ r^N \mapsto r'^N) = min\left(1, \exp\left(-\beta\left[U(r'^N) - U(r^N)\right]\right)\right)$$

used to model the interaction among the particles in the liquid, i.e., to model the particles' behavior according to the concentration of particles in their neighborhood. The parameters σ and ϵ are used to shape the function and r is the distance to neighboring particles.

$$V(r) = 4\epsilon\left[\left(\frac{\sigma}{r}^{12}\right) - \left(\frac{\sigma}{r}^{6}\right)\right] \tag{2}$$

Stirring a material is another type of mass transfer called convection. Convection is the movement of mass due to forced fluid movement. Convective mass transfer is a faster mass transfer than diffusion and happens when stirring is involved. The faster the fluid moves, the more mass transfer and therefore the less time it takes to mix the ingredients together. We simulated this physical property by simply introducing an impulse in stirring direction to the particles in the point cloud that are in reach of the cooking spoon. In this way, we could achieve with this simple model the behavior of molecular motion due to forced fluid movement.

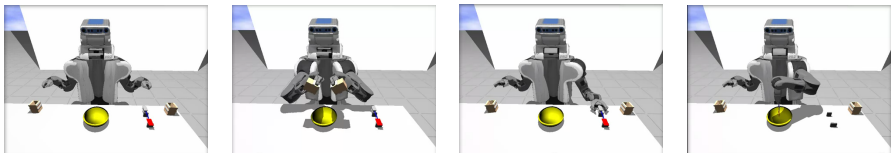
Measuring the Homogeneity Particular interest is the homogeneity of the liquid when stirred was involved in the conducted experiments. It was decided to use the local density of the particles represented as point cloud as a measure of divergence, while using the assumption that the inverse of this is a measure of homogeneity. This distance measure [9] is known as the Jensen-Shannon divergence and used widely in information theory. The Jensen-Shannon divergence is defined as:

$$JS(P,Q) = \tfrac{1}{2}S\left(P, \tfrac{P+Q}{2}\right) + \tfrac{1}{2}S\left(Q, \tfrac{P+Q}{2}\right) \tag{3}$$

where $S(P,Q)$ is the Kullback divergence shown in equation 4, and P and Q two probability distributions defined over a discrete random variable x.

$$S(P,Q) = \textstyle\sum_x P(x)\log\left(\frac{P(x)}{Q(x)}\right) \tag{4}$$

We propose the division of the point cloud in a three-dimensional grid. Each cell of the grid represents a discrete probability distribution x defined on the mixed probabilities of the two classes P and Q, computed as the relative frequency.



(a) Initial Position (b) Pouring (c) Grasping Spoon (d) Stirring

Fig. 3. PR2 pours two ingredients in a bowl and stirs them.

4 Preliminary Experimental Results

In this section we are going to highlight the results of the conducted experiments, namely mixing ingredients together in a bowl while measuring the level of homogeneity[7], and second, pouring the mix onto a pancake maker and reasoning about whether some liquid was spilled[8].

4.1 Mixing Liquids — Analysis of Homogeneity

For mixing the liquids, we used the previously described model (Section 3.2) to simulate the physical effects when stirring with different trajectories. Figure 3 shows the PR2 performing the task.

We selected the coefficients to represent two viscous liquids. Figure 4 shows the course of homogeneity without stirring and as we expect: the ingredients do not mix very well. We repeated the same experiment by assuming (1) an elliptic trajectory, (2) an up-and-down movement and (3) a spiral trajectory. The result of the experiment confirms our hypothesis: Stirring does indeed increase the homogeneity of mixed liquids. The result furthermore showed that with an elliptic trajectory the best result could be achieved. Given the knowledge of homogeneous and inhomogeneous regions in the liquid a robot could adapt its parameterization to increase performance.

4.2 Pouring Liquids — Reasoning about Clusters

In this experiment we address the scenario of pouring some pancake mix located in a container onto a pancake maker: the robot grasps a mug containing pancake mix from the table, lifts it and pours the content on a pancake maker (Figure 5). In this experiment we used the resulting timelines to analyze the qualitative outcome of the executed action. The parameterization of the task included the gripper position, the pouring angle and the pouring time. The task was considered

Table 1. Contact information of clusters.

Position	Angle	Time	Mug	Pan	Table	PR2	S pilled
		1.0	1	93,105	-	-	-
	2.0944	1.5	119,1	1,79	-	-	-
		2.0	125	74	1	-	Yes
		1.0	18,1	2,1,1,177	-	-	-
0.1	2.44346	1.5	24	174,1,1	-	-	-
		2.0	19	179,1	-	-	-
		1.0	6	193	1	-	Yes
	2.61799	1.5	11,1	186,1	1	-	Yes
		2.0	19	181	-	-	-
		1.0	133,1,	10	53,1	1	Yes
	2.0944	1.5	113,1,1	15	64,1,1,1,1	-	Yes
		2.0	1,1,127	11,57	1	1	Yes
		1.0	25	50,1	124	-	Yes
0.2	2.44346	1.5	24,2	23,1,1	1,1,1,146	-	Yes
		2.0	1,26	28,1	142,1,1	-	Yes
		1.0	11	-	189	-	Yes
	2.61799	1.5	11	42	146.1	-	Yes
		2.0	1,11	19,1	167,1	-	Yes

[7] http://www.youtube.com/watch?feature=player_embedded&v=ccHXmkKT8CE#!

[8] http://www.youtube.com/watch?feature=player_embedded&v=tzQk7S5PRaY

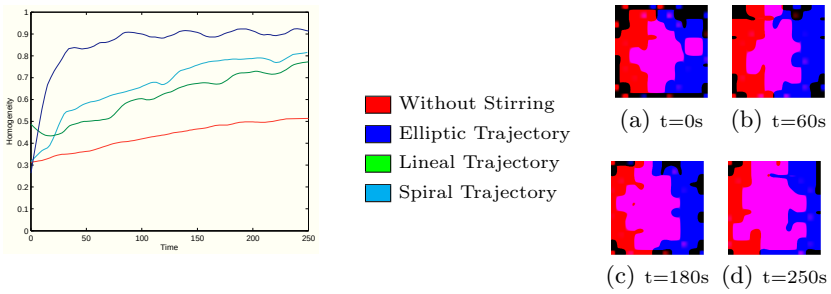
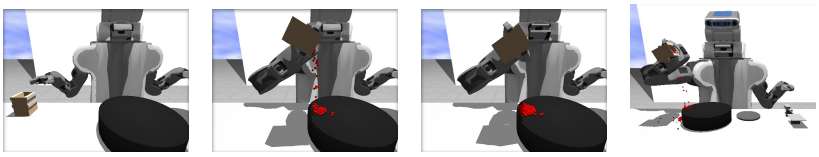


Fig. 4. Left: Homogeneity over time of different stirring trajectories. Right: The color coded images show the spatial distribution of homogeneity of two liquids (without stirring). Black stands for uncovered regions, red and blue for inhomogeneous liquids of corresponding classes and purple homogeneous regions.



(a) Grasping mug (b) Pouring (c) Success (d) Failure

Fig. 5. PR2 pours mix onto pancake maker.

to be successful if no pancake mix has been spilled, i.e. the liquid resides on the pancake maker or in the container and not on other objects such the kitchen table after the pouring action ends. We used the resulting clusters and their corresponding contact information to examine the outcome. Some results are summarized in Table 4.2 where the numbers for *Mug*, *Pan*, *Table*, and *PR2* denote the number of particles of clusters in contact with the respective models. The following Prolog expression shows exemplarily how information about clusters can be retrieved from timelines *TL*

```
?- holds_tt(occurs(pouring(Params)),Interval,TL), [_,End] = Interval,
     partOf(X,pancake_mix), holds(on(X,table),Time,TL),
     after(Time,End),
     simulator_value(size(X,Size),Time,TL),
     simulator_value(mean(X,Mean),Time,TL),
     simulator_value(var(X,Var),Time,TL).
```

where X denotes a cluster of pancake mix in contact with the table after the pouring action has been carried out.

5 Conclusions

The present work can be considered as interdisciplinary research of two fields: Robotics and Artificial Intelligence. With our approach we enable robots to reason about the consequences of their own actions. We equip them with the capability of making appropriate decisions about their parameterizations throughout

their activity using well-established methods of AI and detailed physical simulations. We have developed a system that simulates robot manipulation tasks involving liquids, monitors all relevant states and actions, and translates this information into first-order representations, called timelines. Then, we use the logic programming language Prolog within the previously developed framework [7] to answer queries based on the timeline data structures of the temporal projections. The main contribution of this work is the extension of the framework with respect to liquids. We have developed one representation within the simulator Gazebo and another as additional software layer used for simulating physical characteristics such as diffusion and convection.

We believe that the concept of simulation-based temporal-projections can be used to equip robotic agents with the ability to understand physical consequences of their actions which allows them to adjust their behavior.

Acknowledgments This work has been supported by the EU FP7 Project RoboHow (grant number 288533) and the cluster of excellence Cognition for Technical Systems (Excellence Initiative of the German Research Foundation (DFG)).

References

1. E. Davis. Pouring liquids: A study in commonsense physical reasoning. *Artif. Intell.*, 172(12-13):1540–1578, Aug. 2008.
2. E. Davis. Ontologies and representations of matter. In M. Fox and D. Poole, editors, *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*. AAAI Press, 2010.
3. E. Davis. Qualitative spatial reasoning in interpreting text and narrative. *Spatial Cognition and Computation*, 2012. Forthcoming.
4. D. Frenkel and B. Smit. *Understanding Molecular Simulation, Second Edition: From Algorithms to Applications (Computational Science)*. Academic Press, 2 edition, Nov. 2001.
5. P. Hayes. The naive physics manifesto. In D. Michie, editor, *Expert Systems in the Micro Electronic Age*, pages 242–270. Edinburgh University Press, 1979.
6. B. Johnston and M.-A. Williams. Comirit: Commonsense reasoning by integrating simulation and logic. In *Proceedings of the 2008 conference on Artificial General Intelligence 2008: Proceedings of the First AGI Conference*, pages 200–211, Amsterdam, The Netherlands, The Netherlands, 2008. IOS Press.
7. L. Kunze, M. E. Dolha, and M. Beetz. Logic Programming with Simulation-based Temporal Projection for Everyday Robot Object Manipulation. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, San Francisco, CA, USA, September, 25–30 2011.
8. L. Kunze, M. E. Dolha, E. Guzman, and M. Beetz. Simulation-based temporal projection of everyday robot object manipulation. In Yolum, Tumer, Stone, and Sonenberg, editors, *Proc. of the 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*, Taipei, Taiwan, May, 2–6 2011. IFAAMAS.
9. A. P. Majtey, P. W. Lamberti, and D. P. Prato. Jensen-shannon divergence as a measure of distinguishability between mixed quantum states. *Phys. Rev. A*, 72:052310, Nov 2005.
10. L. Morgenstern. Mid-Sized Axiomatizations of Commonsense Problems: A Case Study in Egg Cracking. *Studia Logica*, 67(3):333–384, 2001.

A structural base for conditional reasoning

Gabriele Kern-Isberner and Christian Eichhorn

Department of Computer Science
Technische Universität Dortmund, Germany
gabriele.kern-isberner@cs.uni-dortmund.de
christian.eichhorn@tu-dortmund.de

Abstract. There are several approaches implementing reasoning based on conditional knowledge bases, one of the most popular being System Z [1]. We look at ranking functions [2] in general, conditional structures and c-representations [3] in order to examine the reasoning strength of the different approaches by learning which of the known calculi of nonmonotonic reasoning (System P and R) and *Direct Inference* are applicable to these inference relations. Furthermore we use the recently proposed *Enforcement*-postulate [4] to show dependencies between these approaches.

1 Introduction

Default reasoning is often based on uncertain rules of the form "if A then usually B" representing semantically meaningful relationships between A and B that may serve as guidelines for rational decision making. Such rules are denoted as *conditionals* and formally written as $(B|A)$. Conditionals are different from material implications $A \Rightarrow B$ in that they can not be interpreted truth functionally but need richer epistemic structures to be evaluated. Ordinal conditional functions, or *ranking functions* [2] provide a most convenient way for evaluating conditionals. Here, a conditional $(B|A)$ is accepted if the rank of its verification $A \wedge B$ is more plausible than the rank of its falsification $A \wedge \neg B$. However, it is often not clear where the numerical ranks come from, and people might be reluctant to accept conditionals just due to a comparison of numbers. In this paper, we show how the acceptance of conditionals can be based on structural arguments that emerge from elaborating systematically the three-valued nature of conditionals. More precisely, we assume a knowledge base of conditionals to be explicitly given, and investigate inferences that can be drawn from this knowledge base in a rational way, like in the well-known penguin example: Let $\Delta = \{(f|b), (\overline{f}\,|p), (b|p)\}$ be the set of conditionals $(f|b) \simeq$ "birds usually fly", $(\overline{f}\,|b) \simeq$ "penguins usually do not fly" and $(b|p) \simeq$ "penguins are usually birds". Commonsense deliberations tell us that from these conditionals we should be able to infer that birds fly if they are not penguins, but penguin-birds not. The intricacy of this example lies in the nonmonotonic inheritance from a superclass to a subclass: albeit being birds, penguins do not inherit the flight capacity of birds.

We briefly recall the conditional structures approach [3] which allows us to define a preference relation between possible worlds and henceforth a nonmonotonic inference relation between formulas. We prove results on the quality of this inference relation but also illustrate its limits. Fortunately, conditional structures can be linked to ranking functions via c-representations, and together with the novel *Enforcement* postulate adapted from belief revision, we are able to show that rank based inferences may respect structural (i.e., non-numerical) information.

2 Preliminaries

Let $\Sigma = \{V_1, \ldots, V_n\}$ be a set of propositional atoms. A *literal* is a positive or negative atom. The set of formulas $\mathfrak{L}$ over Σ, with the connectives $\wedge$ (*and*), $\vee$ (*or*) and $\neg$ (*not*) shall be defined in the usual way. Let $A, B \in \mathfrak{L}$, we will in the following omit the connective $\wedge$ and write AB instead of $A \wedge B$ as well as indicate negation by overlining, i.e. $\overline{A}$ means $\neg A$; the symbol "$\Rightarrow$" is used as material implication, i.e., $A \Rightarrow B$ is equivalent to $\overline{A} \vee B$. *Interpretations*, or *possible worlds*, are also defined in the usual way; the set of all possible worlds is denoted by Ω. We often use the equivalence between worlds and *complete conjunctions*, i.e. conjunctions of literals where every variable $V_i \in \Sigma$ appears exactly once. A model ω of a propositional formula $A \in \mathfrak{L}$ is a possible world that satisfies A, written as $\omega \models A$. The set of all models of A is denoted by $Mod(A)$. For formulas $A, B \in \mathfrak{L}$, A *entails* B, written as $A \models B$, iff $\mathrm{Mod}(A) \subseteq \mathrm{Mod}(B)$, i.e. iff for all $\omega \in \Omega$, $\omega \models A$ implies $\omega \models B$. For sets of formulas $\mathcal{A} \subseteq \mathfrak{L}$ we have $Mod(\mathcal{A}) = \bigcap_{A \in \mathcal{A}} Mod(A)$. A *conditional* $(B|A)$ encodes a defeasible rule "if A then *usually* B" with the trivalent evaluation $[\![(B|A)]\!]_\omega = true$ if and only if $\omega \models AB$ (verification), $[\![(B|A)]\!]_\omega = false$ if and only if $\omega \models A\overline{B}$ (falsification) and $[\![(B|A)]\!]_\omega = undefined$ if and only if $\omega \models \overline{A}$ (non-applicability). The language of all conditionals over $\mathfrak{L}$ is denoted by $(\mathfrak{L} \mid \mathfrak{L})$. Let $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$ be a finite set of conditionals. A conditional $(B|A)$ is *tolerated* by Δ if and only if there is a world $\omega \in \Omega$ such that $\omega \models AB$ and $\omega \models A_i \Rightarrow B_i$ for every $1 \leq i \leq n$. Δ is *consistent* if and only if for every nonempty subset $\Delta' \subseteq \Delta$ there is a conditional $(B|A) \in \Delta'$ that is tolerated by Δ'. We will call such a consistent Δ a *knowledge base* and it shall represent the knowledge an agent uses as a base for reasoning. In this paper, we will only consider Δ that are consistent.

3 Properties of qualitative conditional reasoning

We consider inference relations $\mid\!\sim$ between sets of formulas $\mathcal{A} \subseteq \mathfrak{L}$ and single formulas A. $\mathcal{A} \mid\!\sim A$ means that A can be inferred defeasibly fom $\mathcal{A}$. Contrary to $\models$, $\mid\!\sim$ will usually be nonmonotonic, i.e. we may have $\mathcal{A} \mid\!\sim A$ but $\mathcal{A} \cup \{B\} \not\mid\!\sim A$. The various possible inference relations can be judged by certain *quality criteria* that have been designed for describing rational human reasoning. From this set of criteria, *calculi* are subsets of the quality criteria canon used to classify inference relations. The usual calculi are System O, C, P and R, where System C and O are included in System P, thus we present the most established systems, Systems P and R.

Definition 1 (System P). *[5]*
System P consists of the following conditions

- *Reflexivity* (REF)*:* *From A defeasibly infer A, resp. $A \mid\!\sim A$,*
- *Right Weakening* (RW)*:* *$A \mid\!\sim B$ and $B \models C$ imply $A \mid\!\sim C$,*
- *Left Logical Eqivalence* (LLE)*:* *$A \mid\!\sim C$ and $A \equiv B$ imply $B \mid\!\sim C$,*
- (Cut)*:* *$A \mid\!\sim B$ and $AB \mid\!\sim C$ imply $A \mid\!\sim C$,*
- *Cautious Monotony* (CM)*:* *$A \mid\!\sim B$ and $A \mid\!\sim C$ imply $AB \mid\!\sim C$,*
- (Or)*:* *$A \mid\!\sim C$ and $B \mid\!\sim C$ imply $(A \vee B) \mid\!\sim C$.*

As well as being an important quality criterion for nonmonotonic reasoning systems, empirical studies show that human reasoning makes use of the conditions of System P (c.f. [6]) which renders the inspection of System P especially worthwhile.
The calculi are syntactical and should be based on semantics to be evaluated. A very general one is preferential satisfaction that uses the notion of preferential models which we will introduce with the next two definitions.

Definition 2 ((Classical) Preferential model). *[7]*
Let $M = \{m_1, m_2, \ldots\}$ be an arbitrary set of states, which could be, but is not

limited to, a set of interpretations of a logical language. Let $\vdash$ *be an arbitrary relation* $\vdash \subset M \times \mathfrak{L}$ *called* satisfaction relation *and* $\prec$ *an arbitrary relation* $\prec \subseteq M \times M$ *called* preference relation. *If* $m_1 \prec m_2$ *then* m_1 *is* preferred *to* m_2. *The triple* $\mathcal{M} = \langle M, \vdash, \prec \rangle$ *is called a* preferential model. *A preferential model is called* classical *if* $\prec$ *is transitive, and for all* $m \in M$ *it holds that* $m \vdash \overline{A}$ *iff* $m \nvdash A$ *and* $m \vdash A \vee B$ *iff* $m \vdash A$ *or* $m \vdash B$.

Definition 3 (Preferential satisfaction). *[7]*
Let $\mathcal{A} \subseteq \mathfrak{L}$, $\mathcal{M} = \langle M, \vdash, \prec \rangle$ *be a preferential model and* $m \in M$ *be a state. We say that* m *satisfies* $\mathcal{A}$ *(*$m \vdash \mathcal{A}$*) iff* $m \vdash A$ *for every* $A \in \mathcal{A}$, *and* m preferentially *satisfies* $\mathcal{A}$ *(written* $m \vdash_\prec \mathcal{A}$*) iff* $m \vdash \mathcal{A}$ *and there is no* $m' \in M$ *such that* $m' \vdash \mathcal{A}$ *and* $m' \prec m$. *We define* $[\mathcal{A}] = \{m \in M \,|\, m \vdash \mathcal{A}\}$ *and say* m *is* $\prec$*-minimal in* $[\mathcal{A}]$ *iff* $m \vdash_\prec \mathcal{A}$.

Preferential satisfaction is based on a notion of *minimality*. Since $\prec$ is defined to be an arbitrary relation, it is possible for $[A]$ not to have a minimal element (e.g. because $[A]$ is infinite or contains circles $m_1 \prec m_2 \prec \cdots \prec m_1$). The following definition ensures that an associated minimal element exists.

Definition 4 (Stoppered preferential models). *[7]*
We call a preferential model $\mathcal{M} = \langle M, \vdash, \prec \rangle$ stoppered *if and only if for every set* $\mathcal{A} \subseteq \mathfrak{L}$ *and every* $m \in M$ *if* $m \in [\mathcal{A}]$ *then there is a* $\prec$*-minimal element* m' *in* $[\mathcal{A}]$ *such that either* $m' = m$ *or* $m' \prec m$.

Having defined preferential models, one can now define an entailment relation on preferential models that facilitates reasoning.

Definition 5 (Preferential entailment). *[7]*
Let $\langle M, \vdash, \prec \rangle$ *be a preferential model,* $m, m' \in M$ *and* $A, B \in \mathfrak{L}$. *We define* B *to be* preferentially entailed *by* A *(written* $A \mathrel{|\!\sim} B$*) in the following way:*

$$A \mathrel{|\!\sim} B \qquad \textit{iff} \qquad \forall\, m \in M \;:\; \quad m \vdash_\prec A \quad \textit{implies} \quad m \vdash B \tag{1}$$

Preferential entailment complies with various properties, [7] has shown that if the underlying preferential model is stoppered, these relations fulfil the properties of System P which we will stress in the following proposition.

Proposition 1. *[7] All preferential entailment operations that are generated by a classical stoppered preferential model comply with System P.*

For classical stoppered preferential models, equation (1) is equivalent to

$$A \mathrel{|\!\sim} B \quad \text{iff} \quad \forall\, m' :\; m' \vdash A\overline{B} \qquad \exists\, m :\; m \vdash AB \quad \text{with} \quad m \prec m'.$$

The second calculus we announced to inspect was System R which is basically System P with the additional (non-Horn) property *Rational Monotony*.

Definition 6 (System R). *[8]*
System R is composed of (REF), (Cut), (CM), (RW), (LLE), (Or) *and*
- *Rational Monotony* (RM) $\qquad A \mathrel{|\!\sim} B$ *and* $A \not\mathrel{|\!\sim} \overline{C}$ *implies* $AC \mathrel{|\!\sim} B$.

As given in section 2, a conditional $(B|A)$ stands for the defeasible rule "A usually entails B" and therefore suggests for each $(B|A) \in \Delta$ that $A \mathrel{|\!\sim_\Delta} B$ holds if $\mathrel{|\!\sim_\Delta}$ is based on Δ. This is claimed by the next property.

Definition 7 (Direct Inference (DI)**).** *[9]*
Let $\Delta \subseteq (\mathfrak{L}|\mathfrak{L})$, *let* $\mathrel{|\!\sim_\Delta}$ *be an inference relation based on* Δ. $\mathrel{|\!\sim_\Delta}$ *complies with* (DI) *iff for every* $(B|A) \in \Delta$ *it holds that* $A \mathrel{|\!\sim_\Delta} B$.

4 Ranking Functions (OCF)

An *ordinal conditional function* (OCF, [2]), also called *ranking function*, is a function $\kappa : \Omega \to \mathbb{N}_0^\infty$ with $\kappa^{-1}(0) \neq \varnothing$ which maps each world $\omega \in \Omega$ to a degree of implausibility $\kappa(\omega)$; ranks of formulas $A \in \mathfrak{L}$ are calculated as $\kappa(A) = \min\{\kappa(\omega) \mid \omega \models A\}$. For conditionals $(B|A)$ we have ranks of $\kappa(B|A) = \kappa(AB) - \kappa(A)$ and $\kappa \models (B|A)$ iff. $\kappa(AB) < \kappa(A\overline{B})$, i.e. iff. AB is more plausible than $A\overline{B}$. In this case, we call κ a (ranking) model of $(B|A)$. A ranking function induces a preference relation $\leq_\kappa$ on worlds such that $\omega \leq_\kappa \omega'$ iff $\kappa(\omega) \leq \kappa(\omega')$. We write $\omega <_\kappa \omega'$ iff $\omega \leq_\kappa \omega'$ and $\omega' \not\leq_\kappa \omega$.
OCF-reasoning uses the $<$-relation on natural numbers and the classical inference relation $\models$ which implies immediately that $\langle \Omega, \models, <_\kappa \rangle$ is a classical stoppered preferential model. The inference relation $\mid\!\sim_\kappa$ turns out to be

$$A \mid\!\sim_\kappa B \qquad \text{iff} \qquad \kappa(AB) < \kappa(A\overline{B}) \qquad \text{iff} \qquad \kappa \models (B|A).$$

For a conditional knowledge base $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$ a ranking function κ is Δ-*admissible* iff $\kappa \models (B_i|A_i)$ for every $1 \leq i \leq n$. We write κ_Δ to illustrate that κ is Δ-admissible. Note that for $\mid\!\sim_\kappa$, (DI) is equivalent to Δ-admissability of κ.
Proposition 1 immediately yields the following statement:

Corollary 1. *$\mid\!\sim_\kappa$ complies with System P.*

System R is no consequence of proposition 1. [1] has shown that every κ_Δ complies with (RM), by which the next proposition arises:

Proposition 2. *$\mid\!\sim_\kappa$ complies with System R.*

5 Reasoning with conditional structures

Our intention is to focus on reasoning mechanisms based on the information contained in the knowledge base. By OCF, we have high qualitative reasoning, but the ranks of the worlds need to use the knowledge base as information source. In the following we will examine an approach using the structural information induced by the conditionals in a knowledge base.
Given a conditional knowledge base $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$ we assign a pair of abstract symbols $\mathbf{a}_i^+$ and $\mathbf{a}_i^-$ to each $(B_i|A_i) \in \Delta$ to illustrate the effect of conditionals on worlds. With these, we define the free abelian group $\mathfrak{F}_\Delta = \langle \mathbf{a}_1^+, \mathbf{a}_1^-, \ldots, \mathbf{a}_n^+, \mathbf{a}_n^- \rangle$ on Δ with generators $\mathbf{a}_1^+, \mathbf{a}_1^-, \ldots, \mathbf{a}_n^+, \mathbf{a}_n^-$ consisting of all products $(\mathbf{a}_1^+)^{\alpha_1}(\mathbf{a}_1^-)^{\beta_1} \cdot \ldots \cdot (\mathbf{a}_n^+)^{\alpha_n}(\mathbf{a}_n^-)^{\beta_n}$ with $\alpha_i, \beta_i \in \mathbb{Z}$ for all $1 \leq i \leq n$ [3]. We will keep in mind that in abelian groups commutativity holds, e.g. $\mathbf{a}_1^+\mathbf{a}_2^- = \mathbf{a}_2^-\mathbf{a}_1^+$. To connect a world and the effect of a conditional to this world we define the function $\sigma_i : \Omega \to \mathfrak{F}_\Delta$ by $\sigma_i(\omega) := \mathbf{a}_i^+$ iff. $\omega \models AB$, $\sigma_i(\omega) := \mathbf{a}_i^-$ iff. $\omega \models A\overline{B}$ and $\sigma_i(\omega) := 1$ iff. $\omega \models \overline{A}$ for each $1 \leq i \leq n$. So, $\mathbf{a}_i^+$ ($\mathbf{a}_i^-$) indicates that ω verifies (falsifies) $(B_i|A_i)$, and the neutral group element 1 corresponds to non-applicability of the conditional.

Definition 8 (Conditional structure). *[3]*
Let $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$, $\mathfrak{F}_\Delta = \langle \mathbf{a}_1^+, \mathbf{a}_1^-, \ldots, \mathbf{a}_n^+, \mathbf{a}_n^- \rangle$ and σ_i be as defined above. The conditional structure σ_Δ of a world regarding Δ is the function $\sigma_\Delta : \Omega \to \mathfrak{F}_\Delta$ defined as

$$\sigma_\Delta(\omega) = \prod_{i=1}^{n} \sigma_i(\omega) = \prod_{\substack{i=1 \\ \omega \models A_i B_i}}^{n} \mathbf{a}_i^+ \cdot \prod_{\substack{i=1 \\ \omega \models A_i \overline{B}_i}}^{n} \mathbf{a}_i^-$$

For every world ω the conditional structure $\sigma_\Delta(\omega)$ indicates formally which conditionals in Δ are verified, or falsified by, or not applicable to this world. Note that the group structure allows an elegant way of encoding this.

Example 1. We use the introductory example $\Delta = \{(f|b), (\overline{f}\,|p), (b|p)\}$. The conditional structures with respect to Δ are shown in the following table.

ω	$\sigma_\Delta(\omega)$	ω	$\sigma_\Delta(\omega)$	ω	$\sigma_\Delta(\omega)$	ω	$\sigma_\Delta(\omega)$
pbf	$\mathbf{a}_1^+\mathbf{a}_2^-\mathbf{a}_3^+$	$pb\overline{f}$	$\mathbf{a}_1^-\mathbf{a}_2^+\mathbf{a}_3^+$	$\overline{p}bf$	$\mathbf{a}_1^+$	$\overline{p}b\overline{f}$	$\mathbf{a}_1^-$
$p\overline{b}f$	$\mathbf{a}_2^-\mathbf{a}_3^-$	$p\overline{b}\,\overline{f}$	$\mathbf{a}_2^+\mathbf{a}_3^-$	$\overline{p}\overline{b}\,f$	1	$\overline{p}\overline{b}\,\overline{f}$	1

With σ_Δ we define a preference relation on worlds based on structural information by σ-preferring a world ω to a world ω' iff ω falsifies less conditionals than ω' and ω' falsifies at least the conditionals falsified by ω.

Definition 9 ($\prec_\sigma$-preference). *[4]*
A world ω shall be σ-preferred to a world ω', in terms $\omega \prec_\sigma \omega'$, if and only if for every $1 \leq i \leq n$, $\sigma_i(\omega) = \mathbf{a}_i^-$ implies $\sigma_i(\omega') = \mathbf{a}_i^-$, and there is at least one i such that $\sigma_i(\omega) \in \{\mathbf{a}_i^+, 1\}$ and $\sigma_i(\omega') = \mathbf{a}_i^-$.

The triple $\langle \Omega, \models, \prec_\sigma \rangle$ is a *preferential model* (cf. definition 2) and hence allows to define nonmonotonic inference of some quality. We show that σ-preferential reasoning follows the lines of System P for which the following preliminaries have to be deployed.
Since it is quite obvious that $\prec_\sigma$ is stoppered and transitive, we have the following lemma:

Lemma 1. *$\langle \Omega, \models, \prec_\sigma \rangle$ is a stoppered classical preferential model.*

Using the preference relation $\prec_\sigma$ we define a structual entailment relation according to definition 5.

Definition 10 (σ-structural inference).
Let A, B be formulas in $\mathfrak{L}$ and $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$. B can be structurally inferred*, or* σ-inferred*, from A, written as*

$$A \mathrel{|\!\sim^\sigma_\Delta} B \quad \textit{iff} \quad \forall\, \omega' :\ \omega' \models A\overline{B} \quad \exists\, \omega :\ \omega \models AB \quad \textit{with} \quad \omega \prec_\sigma \omega'.$$

We see that A σ-infers B if and only if for every world $\omega' \in Mod(A\overline{B})$ there is a σ-preferred world $\omega \in Mod(AB)$.

Example 2. We use the knowledge base Δ from example 1. By σ-structural inference, we find that flying birds are no penguins ($bf \mathrel{|\!\sim^\sigma_\Delta} \overline{p}$) since for every world ω' which is a model of pbf, namely $\omega' = pbf$, there is a world ω which is a model of $\overline{p}bf$, namely $\omega = \overline{p}bf$, for which we see, that $\sigma_1(\omega') = \mathbf{a}_i^+$, $\sigma_1(\omega) = \mathbf{a}_i^+$, $\sigma_2(\omega') = \mathbf{a}_i^-$, $\sigma_2(\omega) = 1$, $\sigma_3(\omega') = \mathbf{a}_i^+$ and $\sigma_3(\omega) = 1$ and therefore by the above definition it holds that $\omega \prec_\sigma \omega'$.

From lemma 1 and proposition 1, we obtain:

Proposition 3. *$\mathrel{|\!\sim^\sigma_\Delta}$ satisfies System P.*

However, there ist an odd finding, shown by the next example, which suggests that $\mathrel{|\!\sim^\sigma_\Delta}$ may violate System P.

Example 3 (Counterexample to System P-compliance of $\mathrel{|\!\sim^\sigma_\Delta}$?). We use the running example with the conditional structures from example 1. We would expect that $p \mathrel{|\!\sim^\sigma_\Delta} b$ since $(b|p) \in \Delta$, and $p \mathrel{|\!\sim^\sigma_\Delta} \overline{f}$ since $(\overline{f}\,|p) \in \Delta$, therefore by (CM) it should follow that $pb \mathrel{|\!\sim^\sigma_\Delta} \overline{f}$. But $\sigma_\Delta(pbf) = \mathbf{a}_1^+\mathbf{a}_2^-\mathbf{a}_3^+$ and $\sigma_\Delta(pb\overline{f}) = \mathbf{a}_1^-\mathbf{a}_2^+\mathbf{a}_3^+$, so $pb\overline{f} \not\prec_\sigma pbf$, hence $pb \not|\!\sim^\sigma_\Delta \overline{f}$.

Example 3 is, of course, no counterexample to the System P compliance of classical stoppered preferential models but to the assumption, that we may always conclude $A \mathrel{|\!\sim} B$ if $(B|A) \in \Delta$, which we formalised as property (DI).

Proposition 4. *$\mathrel{|\!\sim^\sigma_\Delta}$ does not comply with* (DI)

Proof. Example 1 is a counterexample for $\mathrel{|\!\sim^\sigma_\Delta}$and DI: We see that for the world $\omega' = pbf$ there is no $\omega \models p\overline{f}$ with $\omega \prec_\sigma \omega'$. So $p \mathrel{|\!\sim^\sigma_\Delta} \overline{f}$ does not hold.

By this, we see that $p \mathrel{|\!\sim^\sigma_\Delta} \overline{f}$ does not hold for the running example and because of that, example 3 is no counterexample to proposition 3.
It is apparent that this problem arises due to the incomparability of "alternating symbols" i.e. if for worlds ω, ω' we have $\sigma_i(\omega) = \mathbf{a}_i^+$, $\sigma_i(\omega') = \mathbf{a}_i^-$ and $\sigma_j(\omega) = \mathbf{a}_j^-$, $\sigma_j(\omega') = \mathbf{a}_j^+$ for at least one pair of $1 \leq i, j \leq n$, $i \neq j$. In this case, ω and ω' are *stucturally incomparable.*

6 Reasoning with c-representations

To solve the problem of structurally incomparable worlds that became evident in the previous section, we need to introduce some kind of weights for conditionals to compare the falsification of *different* conditionals.

Definition 11 (c-representations). *A c-representation [3] of a knowledge base* $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$ *is defined as an OCF of the form*

$$\kappa^c_\Delta(\omega) = \sum_{\substack{i=1 \\ \omega \models A_i\overline{B}_i}}^{n} \kappa_i^-, \qquad \kappa_i^- \in \mathbb{N}_0 \tag{2}$$

where the values κ_i^- *are penalty points for falsifying conditionals and have to be chosen to make* κ^c_Δ Δ*-admissible, i.e. for all* $1 \leq i \leq n$ *it holds that* $\kappa^c_\Delta \models (B_i|A_i)$ *which is the case if and only if*

$$\kappa_i > \min_{\omega \models A_iB_i} \left\{ \sum_{\substack{i \neq j \\ \omega \models A_j\overline{B}_j}} \kappa_j^- \right\} - \min_{\omega \models A_i\overline{B}_i} \left\{ \sum_{\substack{i \neq j \\ \omega \models A_j\overline{B}_j}} \kappa_j^- \right\}. \tag{3}$$

A minimal *c-representation is obtained by choosing* κ_i^- *minimally according to (3) for all* i, $1 \leq i \leq n$.

Example 4 (c-represented penguins). We use, from the introductory example, the knowledge base $\Delta = \{(f|b), (\overline{f}\,|p), (b|p)\}$. For the κ_i^- values of a c-representation we get, according to (3), $\kappa_1^- > 0$, $\kappa_2^- > \min\{\kappa_1^-, \kappa_3^-\}$ and $\kappa_3^- > \min\{\kappa_1^-, \kappa_2^-\}$. A minimal c-representation for Δ is calculated with $\kappa_1^- = 1$, $\kappa_2^- = \kappa_3^- = 2$ and the resulting ranking of worlds is shown in the following table.

ω	$\kappa^c_\Delta(\omega)$	ω	$\kappa^c_\Delta(\omega)$	ω	$\kappa^c_\Delta(\omega)$	ω	$\kappa^c_\Delta(\omega)$
pbf	2	$p\overline{b}\,f$	4	$\overline{p}\,bf$	0	$\overline{p}\,\overline{b}\,f$	0
$pb\overline{f}$	1	$p\overline{b}\,\overline{f}$	2	$\overline{p}\,b\overline{f}$	1	$\overline{p}\,\overline{b}\,\overline{f}$	0

For c-representations, we make use of the preference relation $<_\kappa$ and the inference relation $\mid\!\sim_\kappa$ defined for general OCF's κ (see section 4).

Definition 12 (Preference and inference by c-representation). *A world* $\omega \in \Omega$ *is* κ^c_Δ*-preferred to a world* $\omega' \in \Omega$ ($\omega <_{\kappa^c_\Delta} \omega'$) *if and only if* $\kappa^c_\Delta(\omega) < \kappa^c_\Delta(\omega')$. *For a knowledge base* Δ, *a formula* B *is* κ^c_Δ*-inferrable from* A ($A \mid\!\sim_{\kappa^c_\Delta} B$) *if and only if* $\kappa^c_\Delta(AB) < \kappa^c_\Delta(A\overline{B})$.

C-representations elaborate conditional structures in a more sophisticated way than structural inference and provide an inference relation that surpasses, e.g., System Z. For the axiomatic derivation of c-representations from conditional structures, the abelian group property of $\mathfrak{F}_\Delta$ is needed, for further information, please see [3].
Since every c-representation is an OCF, $\mid\!\sim_{\kappa^c_\Delta}$ inherits the properties of $\mid\!\sim_\kappa$.

Corollary 2. $\mid\!\sim_{\kappa^c_\Delta}$ *complies with System P and R.*

We introduced c-representations to solve the problem of structually incomparable worlds which arose in section 5. Indeed, c-representations employ numerical penalty values that may play the roles of weights. However, it is not at all clear that $\mid\!\sim_{\kappa^c_\Delta}$ refines $\mid\!\sim^\sigma_\Delta$. By the following example we see that $A \mid\!\sim^\sigma_\Delta B$ does not necessarily imply $A \mid\!\sim_{\kappa^c_\Delta} B$. So, with c-representations we have a different, high-quality inference, but the relevance for solving the addressed problem of $\mid\!\sim^\sigma_\Delta$is not obvious.

Proposition 5. *There are knowledge bases* Δ *such that* $A \mid\!\sim^\sigma_\Delta B$ *does not imply* $A \mid\!\sim_{\kappa^c_\Delta} B$.

Example 5. Let $\Delta = \{(b|a), (bc|a)\}$. A minimal c-representation is obtained from $\kappa_1^- = 0$, $\kappa_2^- = 1$. Ranks and conditional structures are shown below, we see that $a\overline{c} \mid\!\sim^{\sigma}_{\Delta} b$ but $a\overline{c} \not\mid\!\sim_{\kappa^c_\Delta} b$.

ω	$\kappa^c_\Delta(\omega)$	$\sigma_\Delta(\omega)$	ω	$\kappa^c_\Delta(\omega)$	$\sigma_\Delta(\omega)$	ω	$\kappa^c_\Delta(\omega)$	$\sigma_\Delta(\omega)$	ω	$\kappa^c_\Delta(\omega)$	$\sigma_\Delta(\omega)$
abc	0	$\mathbf{a}_1^+\mathbf{a}_2^+$	$ab\overline{c}$	1	$\mathbf{a}_1^+\mathbf{a}_2^-$	$\overline{a}bc$	0	1	$\overline{a}b\overline{c}$	0	1
$a\overline{b}c$	1	$\mathbf{a}_1^-\mathbf{a}_2^-$	$a\overline{b}\overline{c}$	1	$\mathbf{a}_1^-\mathbf{a}_2^-$	$\overline{a}\overline{b}c$	0	1	$\overline{a}\overline{b}\overline{c}$	0	1

The problem arises because the second rule, $(bc|a)$, also establishes the first rule, $(b|a)$, hence $\kappa_1^- = 0$. To approach this problem we examine the recently proposed postulate of *Enforcement* (ENF) [4]. This was proposed for *revisions* of ranking functions in the belief revision framework. We recall the necessary preliminaries from [4] in the following:
Let κ be a ranking function and $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$ be a knowledge base. Let $\kappa^* = \kappa * \Delta$ be the ranking function which results from revising the epistemic state κ by the new information Δ. As a quality criterion, (ENF) postulates that if for two worlds $\omega, \omega' \in \Omega$ it holds that $\omega \prec_\sigma \omega'$, then $\omega \leq_\kappa \omega'$ implies $\omega <_{\kappa^*} \omega'$. To use this postulate for inductive reasoning we revise the uniform ranking function κ_u, that is the ranking function for which $\kappa_u(\omega) = 0$ for all $\omega \in \Omega$, with the knowledge base Δ which we want to rely our reasoning on. For this special case, $\omega \leq_{\kappa_u} \omega'$ is trivially fulfilled for all $\omega, \omega' \in \Omega$ and (ENF) boils down to the following postulate:

Definition 13 (Enforcement for inductive reasoning (Ind-ENF)**).** *Let $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$. A Δ-admissible ranking function κ_Δ respects* (Ind-ENF) *if $\omega \prec_\sigma \omega'$ implies $\omega <_{\kappa_\Delta} \omega'$ for all $\omega, \omega' \in \Omega$.*

So a Δ-admissible ranking function κ_Δ respects (Ind-ENF) if the structural preference induced by σ_Δ is respected by κ_Δ. This leads to establishing conditional dependencies more thoroughly: (Ind-ENF) ensures that when learning $(b|a)$, also both conditionals $(b|ac)$ and $(b|a\overline{c})$ are established, as long as no other conditional in the knowledge base inhibits this. So, the problem shown in Example 5 does not occur.

Lemma 2. *If a Δ-admissible ranking function κ_Δ respects* (Ind-ENF)*, then $A \mid\!\sim^{\sigma}_{\Delta} B$ implies $A \mid\!\sim_{\kappa_\Delta} B$.*

Proof. By definition $A \mid\!\sim^{\sigma}_{\Delta} B$ iff for all $\omega' \models A\overline{B}$ there is an $\omega \models AB$ such that $\omega \prec_\sigma \omega'$. If (Ind-ENF) holds, then also $\omega <_{\kappa_\Delta} \omega'$. Hence $\kappa_\Delta(AB) < \kappa_\Delta(A\overline{B})$ and so have $A \mid\!\sim_{\kappa_\Delta} B$. □

We see that the OCF from example 5 does not respect (Ind-ENF) since $abc \prec_\sigma ab\overline{c}$ but $abc \not<_{\kappa^c_\Delta} ab\overline{c}$. The next proposition gives a simple criterion to check if a given c-representation satisfies (Ind-ENF).

Proposition 6. *Let $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$ be a conditional knowledge base. A c-representation κ^c_Δ with $\kappa_i^- > 0$ for all $1 \leq i \leq n$ respects* (Ind-ENF)*.*

Proof. Let $\Delta = \{(B_1|A_1), \ldots, (B_n|A_n)\} \subseteq (\mathfrak{L} \mid \mathfrak{L})$ with c-representation κ^c_Δ. We define $\mathcal{I}(\omega) = \{i \mid \omega \models A_i\overline{B}_i\} = \{i \mid \sigma_i(\omega) = \mathbf{a}_i^-\}$. If $\omega \prec_\sigma \omega'$, we have $\mathcal{I}(\omega) \subsetneq \mathcal{I}(\omega')$ and the difference $\kappa^c_\Delta(\omega') - \kappa^c_\Delta(\omega)$ for these worlds is

$$\kappa^c_\Delta(\omega') - \kappa^c_\Delta(\omega) = \sum_{i \in \mathcal{I}(\omega')} \kappa_i^- - \sum_{i \in \mathcal{I}(\omega)} \kappa_i^- = \sum_{i \in (\mathcal{I}(\omega') \setminus \mathcal{I}(\omega))} \kappa_i^- > 0,$$

since $\kappa_i^- > 0$ for all $1 \leq i \leq n$. Hence, (Ind-ENF) holds. □

Note, however, that postulating $\kappa_i^- > 0$ for each conditional $(B_i|A_i)$ in the knowledge base Δ is usually too strong, since Δ may contain equivalent conditionals.

Conclusion

In this paper, we presented an approach to base inductive conditional reasoning on structural arguments by observing which conditionals of the knowledge base are verified or falsified, respectively, by possible worlds. This induces a preference relation between possible worlds, and allows us to define a preferential entailment relation with nice properties. We also drew attention to the property of Direct Inference which links inference relations to conditional knowledge bases, and formalized an enforcement postulate for inductive reasoning which claims that structural differences between worlds must be reflected appropriately by the preference relation underlying preferential entailment. We applied these ideas to c-representations which allow for inductive conditional reasoning of high quality. A more thorough evaluation of c-representations that obey (Ind-ENF) is part of our ongoing research.

Acknowledgment: We thank the anonymous referees for their valuable hints that helped us improving the paper. This work was supported by Grant KI $1413/5-1$ to Prof. Dr. Gabriele Kern-Isberner from the Deutsche Forschungsgemeinschaft (DFG) as part of the priority program "New Frameworks of Rationality" (SPP 1516). Christian Eichhorn is supported by this grant.

References

1. Pearl, J.: System Z: A natural ordering of defaults with tractable applications to nonmonotonic reasoning. In: Proceedings of the 3rd conference on Theoretical aspects of reasoning about knowledge. TARK '90, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (1990) 121–135
2. Spohn, W.: The Laws of Belief: Ranking Theory and Its Philosophical Applications. Oxford University Press (2012)
3. Kern-Isberner, G.: Conditionals in Nonmonotonic Reasoning and Belief Revision – Considering Conditionals as Agents. Volume 2087. Springer (2001)
4. Kern-Isberner, G., Krümpelmann, P.: A constructive approach to independent and evidence retaining belief revision by general information sets. In: Proceedings of the Twenty-Second international joint conference on Artificial Intelligence - Volume Two. IJCAI'11, AAAI Press (2011) 937–942
5. Kraus, S., Lehmann, D., Magidor, M.: Nonmonotonic reasoning, preferential models and cumulative logics. Artificial Intelligence Journal **44** (1990) 167–207
6. Pfeifer, N., Kleiter, G.D.: Nonmonotonicity and Human Probabilistic Reasoning. In Vejnarová, J., ed.: Proceedings of the 6th Workshop on Uncertainty Processing, Hejnice, Czech Republic, University of Economics, Prague (2003) 221–234
7. Makinson, D.: General patterns in nonmonotonic reasoning. In Gabbay, D.M., Hogger, C.J., Robinson, J.A., eds.: Handbook of logic in artificial intelligence and logic programming. Volume 3. Oxford University Press, Inc., New York, NY, USA (1994) 35–110
8. Costa, H.A., Parikh, R.: Conditional probability and defeasible inference. Journal of Philosophical Logic **34**(1) (2005) 97–113
9. Lukasiewicz, T.: Weak nonmonotonic probabilistic logics. Artif. Intell. **168**(1-2) (October 2005) 119–161

Towards a Declarative Approach to Model Human Reasoning with Nonmonotonic Logics

Christoph Wernhard
Technische Universität Dresden
christoph.wernhard@tu-dresden.de

Abstract. Stenning and van Lambalgen introduced an approach to model empirically studied human reasoning with nonmonotonic logics. Some of the research questions that have been brought up in this context concern the interplay of the open- and closed-world assumption, the suitability of particular logic programming semantics for the modeling of human reasoning, and the role of three-valued logic programming semantics and three-valued logics. We look into these questions from the view of a framework where logic programs that model human reasoning are represented declaratively and are mechanizable by classical formulas extended with certain second-order operators.

1 Introduction

When humans are presented with reasoning tasks, typical "fallacies" can be observed, conclusions that are not sound in a naive classical logic understanding. Such patterns of human reasoning are researched in cognitive science, with [1] being a landmark work. Recently, an approach to model such empirical observations by means of logic programs has been presented [18, 19]. It involves a two-step process, first the construction of a logic program from, for example, natural language statements and the choice of a logic programming semantics. Second, the actual reasoning, straightforwardly performed with respect to the program and chosen semantics. For modeling the findings from [1], a variant of the three-valued Fitting operator semantics had been originally suggested in [18] and corrected and developed further in [6–8, 2]. The following research questions were brought up, among others, by these works:

Q1: A particular variant of the Fitting operator semantics, along with a related variant of predicate completion, is required to model the results from [1]. What exactly are the roles of these variants? Are there analogue variants of other logic programming semantics?
Q2: Which logic programming semantics can be applied to model human reasoning according to the approach of [19]?
Q3: What are the roles of three-valued logic programming semantics and three-valued logics in the modeling of human reasoning?

We approach these questions here from a particular point of view where the logic programs that model human reasoning are represented by formulas of classical logic, extended by second-order operators for specific application patterns of predicate quantification [22]. A particular such pattern is predicate circumscription [14, 11], which allows to express formulas that have as models only those models of the argument formula which are minimal with respect to the extensions of a given set of predicates. For a formula with second-order operators, in many cases an equivalent formula without such operators can be computed by

eliminating these operators. With this approach, the relations between the nonmonotonic logic programming semantics and classical logic are established not by syntactic transformations, but by semantically defined second-order operators.

Typically, in logic programming the meaning of a predicate occurrence depends on its syntactic context, that is, whether it is in the head or in the body of a rule, or is subject to negation as failure. In our classical representations of programs, we distinguish these meanings by letting each original predicate p correspond to several predicates $p^0, p^1, \ldots$, one for each relevant such non-classical context. With this framework, several established logic programming semantics can be characterized as patterns in which circumscription and other second-order operators are applied, based on characterizations by means of syntactic translations, in particular, of predicate completion in terms of circumscription [10], stable models semantics in terms of circumscription [13, 12], and partial stable models semantics [16] in terms of two-valued stable models [9]. Consider for example the following normal logic program:

$$\begin{array}{l} l \leftarrow e,\ \mathsf{not}\ ab. \\ e \leftarrow \mathsf{true}. \end{array} \qquad \text{(i)}$$

It can be represented by the classical propositional formula $(l^0 \leftarrow e^0 \wedge \neg ab^1) \wedge e^0$, where its stable model semantics can be rendered by the second-order formula

$$\mathsf{rename}_{1\backslash 0}(\mathsf{circ}_{(0\cap \mathsf{POS})\cup 1}((l^0 \leftarrow e^0 \wedge \neg ab^1) \wedge e^0)). \qquad \text{(ii)}$$

The second-order operator $\mathsf{rename}_{1\backslash 0}$ expresses systematic renaming of predicates with superscript 1 to their counterparts with superscript 0. The other second-order operator $\mathsf{circ}_{(0\cap \mathsf{POS})\cup 1}$ expresses parallel predicate circumscription [11] of the predicates with superscript 0 while leaving predicates with superscript 1 fixed. (See [20] for precise specifications of these operators.) Eliminating the second-order operators in (ii) yields the classical propositional formula $(e^0 \wedge l^0 \wedge \neg ab^0)$ which is equivalent to (ii) and whose models correspond to the stable models of the original program (i), that is, the single stable model $\{e, l\}$.

Envisaged benefits of this approach, in particular in the context of modeling human reasoning, include the following:

- The second-order formulas that express logic programs which in turn model human reasoning tasks provide a *declarative view of human reasoning*. With these formulas, different operational methods to construct them and to perform reasoning can be associated, in analogy to calculi. Calculi follow different paradigms, such as "model-based" versus "rule-based". Similarly, different paradigms in human reasoning such as "mental models" versus "rules" or neural approaches can be related to a single declarative representation.
- The framework allows *mechanization at all levels*, not just execution of the logic program in a way that simulates human reasoning. Also the metalevel of the characterizations of nonmonotonic semantics is mechanizable. Automated deduction systems can be applied to reason about features of human reasoning and to systematize them.
- Humans apply different ways of reasoning. The proposed approach allows to take this into account by allowing to express *different nonmonotonic semantics within a single framework*, where they can be used together. In addition, features like disjunctive heads, negation as failure in the head and first-order quantification can straightforwardly be incorporated.

– With the proposed approach, nonmonotonic semantics are in essence *reduced to combinations of a few general operators and principles*, like circumscription applied to predicate occurrences. It may be of interest to investigate whether patterns at this level of combination of general operators match observed patterns of human reasoning.

The rest of the paper is organized around the three mentioned research questions: Question Q1 is addressed in Section 2, *Integration of Open- and Closed-World Reasoning*, question Q2 in Section 3, *Logic Programming Semantics for Modeling Human Reasoning*, and question Q3 in Sections 4 and 5 about three-valued logic programming semantics and three-valued logics, respectively, for modeling human reasoning.

2 Integration of Open- and Closed-World Reasoning

According to the approach of [19], it is essential for the application of logic programming to model human reasoning that some predicates are subject to the closed-world assumption and others to the open-world assumption. With predicate circumscription, selective closed-world reasoning can be expressed naturally by specifying which predicates are to be minimized and which are fixed. Other nonmonotonic semantics have to be generalized such that they allow selective closed-word reasoning. Consider the following program, which models the two conditionals *"if she has an essay to write she will study in the library"* and *"if she has a textbook to read she will study in the library"* from [1] according to [19], where l stands for *"she will study in the library"*, e for *"she has an essay to write"*, and t for *"she has a textbook to read"*.

$$\begin{array}{l} l \leftarrow e,\ \mathsf{not}\ ab_1. \\ l \leftarrow t,\ \mathsf{not}\ ab_2. \end{array} \qquad \text{(iii)}$$

According to [19], abnormality predicates (ab_1, ab_2) and predicates occurring in a rule head (l) are considered closed-world, and the remaining predicates (e, t) open-world. Now, the negative fact *"she does not have an essay to write"* is added to the human reasoning scenario. Thus e should be set to false. Since this can not be expressed in the syntax of normal logic programs, in [19] a special syntax for negative facts is provided that allows to write $e \leftarrow \mathsf{false}$. If we want to stay within normal logic programs, the negated e can be expressed simply by considering e as subject to the closed-world assumption, leaving just t open-world. The stable model of program (iii) with respect to t considered as open-world can be expressed by the following second-order formula:

$$\mathsf{rename}_{1\backslash 0}(\mathsf{circ}_{(0\cap \mathsf{POS})\cup 1\cup\{t^0\}}((l^0 \leftarrow e^0 \wedge \neg ab_1^1) \wedge (l^0 \leftarrow t^0 \wedge \neg ab_2^1))). \qquad \text{(iv)}$$

The open-world predicate t is passed as parameter to the circumscription operator to the effect that t is considered with respect to the circumscription as fixed. Elimination of the second-order operators in (iv) yields

$$(l^0 \wedge t^0 \wedge \neg e^0 \wedge \neg ab_1^0 \wedge \neg ab_2^0) \vee (\neg l^0 \wedge \neg t^0 \wedge \neg e^0 \wedge \neg ab_1^0 \wedge \neg ab_2^0), \qquad \text{(v)}$$

corresponding to two stable models: $\{l, t\}$ and $\{\}$. Thus $\neg l$, is not a consequence of program (iii) under stable models semantics with t considered open-world, matching the empiric results reported in [1], where in that scenario only 5% of the subjects conclude *she will not study in the library.*

Further logic programming semantics can be generalized to take open-world predicates into account analogously to the stable models semantics. This holds in particular for the supported models semantics and for three-valued generalizations of these two-valued semantics: the partial stable models semantics [16], the related well-founded semantics, and semantics based on the Fitting operator [4]. In all cases, the open-world predicates are passed as fixed predicates to an occurrence of the circumscription operator [20].

There are alternate ways of expressing the incorporation of the required open-world reasoning into logic programs: In [19], a variant of predicate completion is used, called *weak completion* in [6], where predicates that do not occur in a head are exempt from completion. This works for the completion based semantics such as supported models and the three-valued semantics based on the Fitting operator, however it is not straightforwardly adaptable to rules with first-order variables [20]. Another possibility is to use the standard versions of the logic programming semantics and extended programs by special rules to encode that certain predicates are open-world: For the completion based semantics this can be achieved by adding $p \leftarrow p$ for each open-world predicate p. For the stable and the partial stable models semantics, and for the completion based semantics as well, this can be achieved by adding two rules $p \leftarrow \neg not\_p$ and $not\_p \leftarrow \neg p$, where not_p is a fresh predicate, for each open-world predicate p. Existential predicate quantification can be applied to the newly generated $not\_$ predicates, such that they do not occur in the final results. For all these variants, equivalence to the circumscription based representation can be shown [20].

3 Logic Programming Semantics for Modeling Human Reasoning

In the literature, so far only a single logic programming semantics has been investigated in the context of modeling human reasoning according to [19]: The least fixed point of the Fitting operator (modified to take open-world predicates into account) and its rendering as least model of the program completion viewed as formula in a three-valued logic. The program representation of a human reasoning scenario is considered adequate with respect to the empiric results if certain conclusions drawn or not drawn by a significant number of human subjects correspond to facts that are entailed or not entailed, respectively, by the program, like $\neg l$ in the example above.

In Section 2 we already have seen an example that has been evaluated with a different logic programming semantics, the two-valued stable models semantics, generalized to take open-world predicates into account. It can be shown that for the scenarios from [1] studied in [19] with this semantics the same adequacy results as for the three-valued Fitting operator based semantics can be obtained. For the logic programs according to [19], the supported models semantics, again generalized to take open-world predicates into account, yields exactly the same models as the stable models semantics. Considering three-valued semantics, the adequacy results obtained for the Fitting operator based semantics can also be obtained with the partial stable models semantics and the well-founded semantics, when these are generalized to take open-world predicates into account. These correspondences have been shown for "forward-reasoning" tasks, that is,

modus ponens and *denial of the antecedent* in [20]. Combining this with results from [8, 21], they are expected also to hold for abduction based "backward reasoning", that is, *modus tollens* and *affirmation of the consequent*.

That the stable models semantics yields the same models as the supported models semantics does not come as a surprise, since by Fages' theorem [3] both semantics are identical for programs that are *tight*, that is, do not involve "positive loops". The experiments discussed in [19] all lead to tight programs. It is not difficult to transfer Fages' theorem to a three-valued setting, where for tight programs the models represented by the fixed-points of the Fitting operator are exactly the partial stable models. Accordingly, the model represented by the least fixed-point of the Fitting operator is the well-founded model.

It seems currently unclear whether there are interesting scenarios of human reasoning that would lead to logic programs which are not tight or to programs of richer classes, for example by permitting disjunctive rules, where the completion based semantics might differ from those based on stable models.

4 Three-Valued Logic Programming Semantics for Modeling Human Reasoning

In human reasoning positive and negative knowledge is apparently not handled symmetrically. The Fitting operator [4] represents a particular asymmetric way of inferring positive information (heads of rules whose body is verified) and negative information (negated heads in case the bodies of all rules with the atom as head are falsified). As shown in [19], this matches the empirical results from [1] about the suppression task when open-world predicates are properly considered.

The Fitting operator semantics and the partial stable models semantics can be expressed with second-order operators in two-valued classical logic [20], where the representation of the partial stable models semantics is based on a translation into the two-valued stable models semantics [9]. The following formulas show how the Fitting operator semantics is rendered in our framework for the example program (iii), with t considered as open-world:

$$\begin{aligned}
&\mathsf{circ}_{\mathsf{INFMIN}}(\mathsf{fitting}_{\{t^0,t^1,t^2\}}((l^0 \leftarrow e^2 \wedge \neg ab_1^1) \wedge (l^0 \leftarrow t^2 \wedge \neg ab_2^1))) && (1)\\
\equiv\ &\mathsf{circ}_{\mathsf{INFMIN}}(\mathsf{CONS} \wedge (l^0 \leftarrow e^0 \wedge \neg ab_1^1) \wedge (l^0 \leftarrow t^0 \wedge \neg ab_2^1) \wedge && (2)\\
&\qquad (l^1 \rightarrow (e^1 \wedge \neg ab_1^0) \vee (t^1 \wedge \neg ab_2^0)) \wedge\\
&\qquad (e^1 \rightarrow \mathsf{false}) \wedge (ab_1^1 \rightarrow \mathsf{false}) \wedge (ab_2^1 \rightarrow \mathsf{false}))\\
\equiv\ &\mathsf{circ}_{\mathsf{INFMIN}}(\neg ab_1^0 \wedge \neg ab_1^1 \wedge \neg ab_2^1 \wedge \neg ab_2^0 \wedge \neg e^0 \wedge \neg e^1 \wedge && (3)\\
&\qquad ((l^0 \wedge l^1 \wedge t^0 \wedge t^1) \vee && \text{(vi)}\\
&\qquad (l^0 \wedge l^1 \wedge \neg t^0 \wedge t^1) \vee\\
&\qquad (\neg l^0 \wedge l^1 \wedge \neg t^0 \wedge t^1) \vee\\
&\qquad (\neg l^0 \wedge \neg l^1 \wedge \neg t^0 \wedge t^1) \vee\\
&\qquad (\neg l^0 \wedge \neg l^1 \wedge \neg t^0 \wedge \neg t^1)))\\
\equiv\ &\neg ab_1^0 \wedge \neg ab_1^1 \wedge \neg ab_2^1 \wedge \neg ab_2^0 \wedge \neg e^0 \wedge \neg e^1 \wedge \neg l^0 \wedge l^1 \wedge \neg t^0 \wedge t^1. && (4)
\end{aligned}$$

In the classical representation of the logic program in formula (1), three roles of predicate occurrences are distinguished: In the head, subjected to negation as failure, and in the positive body, superscripted by 0, 1, 2, respectively. The operator $\mathsf{fitting}$ is a shorthand for a certain combination of second-order operators that renders the semantics of the Fitting operator. Its subscript argument in (1) specifies that t is to be considered open-world. (See [20] for precise definitions.)

Step (2) shows the formula (1) after eliminating fitting, which can be considered as the result of a program transformation: CONS, a shorthand for an axiom that excludes certain unwanted models as discussed below, a classical representation of the program with head and positive body superscripted with 0 and the negative body with 1, and the converses of the completion of the latter variant of the program with superscripts flipped.

The predicate superscripts now indicate the contribution to three-valued models: An interpretation assigns the "three-valued" truth value TRUE to an atom p if it is a model of $p^0 \wedge p^1$. It assigns FALSE to p if it is a model of $\neg p^0 \wedge \neg p^1$, and it assigns UNDEFINED to p if it a model of $\neg p^0 \wedge p^1$. The remaining possibility that it is a model $p^0 \wedge \neg p^1$ is excluded by the axiom CONS. In step (3) the argument formula of the circumscription is syntactically simplified, indicating its five three-valued models. Finally, in step (4) the circumscription operator is eliminated, rendering the selection of the *least* model of the Fitting operator. The symbol INFMIN is a shorthand for a parameter that specifies that predicates with superscript 0 are minimized while those with superscript 1 are maximized (our version of predicate circumscription allows maximization, dual to minimization [22]). A model of a formula circumscribed in this way is an *informationally minimal* model of the circumscribed formula, since viewed as three-valued, there is no other model of the formula whose assignments of atoms to FALSE and to TRUE (but not to UNDEFINED) are properly contained in those assignments of the first model.

With this approach, "positive" and "negative" aspects correspond to differently superscripted predicates. In combination they yield three-valued truth values. The same combinations are obtained for the partial stable models semantics based on the translation in [9], where it is suggested to consider the predicates superscripted with 1 as representing potential truth. It may be of interest to investigate whether there are correspondences to observed human reasoning at the level of this "lower layer" of components with different status, representing "true" and "potentially true" knowledge.

5 Three-Valued Logics for Modeling Human Reasoning

The least model of the Fitting operator applied to a normal logic program is the unique informationally minimal model of the program's completion under a certain three-valued logic [4]. This also applies to the considered generalizations in which open-world predicates are taken into account. In the corresponding three-valued logic the semantics of the biconditional must yield TRUE if and only if both argument formulas have the same truth value. This is the case for the semantics of the biconditional considered in [4], where FALSE is the value in all other cases, and also for the biconditional derived from Łukasiewicz's implication, considered in [6], where UNDEFINED is the value if the truth value of exactly one argument is UNDEFINED, and FALSE is the value in the remaining cases. With both variants of the three-valued biconditional, formulas with the syntactic form of a completed normal logic program have exactly the same three-valued models. Similarly, if normal logic programs themselves are considered as three-valued formulas, the models with respect of the implication seq_3 [5], the analogue to the mentioned biconditional considered in [4], used in [15] for logic programs, are exactly the models obtained with Łukasiewicz's implication.

In Section 4 we have seen an assignment of three-valued truth values to *atomic* formulas, by two-valued interpretations over atoms decorated with superscripts 0 and 1. This principle can be extended to *complex* formulas of certain three-valued logics. In [20] such an extension is given for the three-valued logic S_3 [17], the logic applied in [4] to render the semantics of the Fitting operator: A valuation function from S_3 formulas and two-valued interpretations over atoms with superscripts 0 and 1 onto three truth values is specified, where the values of atoms are assigned as described in Section 4 and the values of complex formulas according to the involved three valued connectives. The valuation function is complemented by a translation function that maps an S_3-formula to a classical propositional formula whose predicates are decorated with superscripts 0 and 1 in a compatible way. That is, an interpretation over the superscripted atoms is a model of the translated S_3 formula if and only if the valuation function applied to the formula and the interpretation yields TRUE. In the following example, the part of the argument formula of the circumscription that follows CONS is the value of this translation function, applied to the completion of the program (iii), extended by the rule $t \leftarrow t$ which expresses that t is open-world, viewed as an S_3-formula:

$$\begin{aligned}\mathsf{circ}_{\mathsf{INFMIN}}(&\mathsf{CONS} \wedge \\ &(l^0 \leftrightarrow (e^0 \wedge \neg ab_1^1) \vee (t^0 \wedge \neg ab_2^1)) \wedge (t^0 \leftrightarrow t^0) \wedge (e^0 \leftrightarrow \mathsf{false}) \wedge \\ &(ab_1^0 \leftrightarrow \mathsf{false}) \wedge (ab_2^0 \leftrightarrow \mathsf{false}) \wedge \\ &(l^1 \leftrightarrow (e^1 \wedge \neg ab_1^0) \vee (t^1 \wedge \neg ab_2^0)) \wedge (t^1 \leftrightarrow t^1) \wedge (e^1 \leftrightarrow \mathsf{false}) \wedge \\ &(ab_1^1 \leftrightarrow \mathsf{false}) \wedge (ab_2^1 \leftrightarrow \mathsf{false})).\end{aligned} \tag{vii}$$

The translation function yields two instances of classical representations of the completed program, with superscripts distinguishing predicate occurrences subjected to negation as failure, flipped in the two instances. It can be shown that the formula (vii) is equivalent to (vi) [20]. The translation extends the compositional view of three truth values in terms of two values and superscripted atoms outlined in Section 4 to complex logic formulas.

6 Conclusion

We have looked into the modeling of human reasoning tasks by logic programs proposed in [18, 19] and developed further in [6–8, 2] from the perspective of a framework where nonmonotonic semantics are represented in classical logic extended by some second-order operators. These classical formulas provide a view on logic programming and the modeled human reasoning that is not tied to a particular way of processing, that allows to represent the interplay of different nonmonotonic semantics, and that allows formalization and mechanization not only on the "object-level" by computing outcomes of human reasoning, but also on the "meta-level" of reasoning about features and principles of human reasoning.

The framework lets general features and requirements of the approach to model human reasoning by logic programming become apparent, such as the interplay of open- and closed-world assumption, which can be straightforwardly expressed by parameterizing circumscription appropriately, from where it transfers to several other logic programming semantics by expressing them in terms of circumscription. The inspection of three-valued logic programming semantics and three-valued logics applied to model human reasoning leads to "lower

level" representations, where combinable pieces of knowledge (atomic formulas) are associated with different aspects, like being *true* versus being *potentially true*. This suggests future investigations in the area of human reasoning to see whether some of its features can be analogously explained by combination of related primitives.

References

1. R. M. J. Byrne. Suppressing valid inferences with conditionals. *Cognition*, 31:61–83, 1989.
2. E.-A. Dietz, S. Hölldobler, and M. Ragni. A computational logic approach to the suppression task. In *CogSci 2012*, 2012. To appear.
3. F. Fages. Consistency of Clark's completion and existence of stable models. *Journal of Methods of Logic in Computer Science*, (1):51–60, 1994.
4. M. Fitting. A Kripke-Kleene semantics for logic programs. *Journal of Logic Programming*, 2(4):295–312, 1985.
5. S. Gottwald. *A Treatise on Many-Valued Logics*, volume 9 of *Studies in Logic and Computation*. Research Studies Press, Baldock, UK, 2001.
6. S. Hölldobler and C. D. P. Kencana Ramli. Logic programs under three-valued Łukasiewicz semantics. In *ICLP 2009*, volume 5649 of *LNCS*, pages 464–478. Springer, 2009.
7. S. Hölldobler and C. D. P. Kencana Ramli. Logics and networks for human reasoning. In *ICANN'09*, pages 85–94, 2009.
8. S. Hölldobler, T. Philipp, and C. Wernhard. An abductive model for human reasoning (poster paper). In *Logical Formalizations of Commonsense Reasoning*, AAAI Spring Symposium Series, pages 135–138. AAAI Press, 2011.
9. T. Janhunen, I. Niemelä, D. Seipel, P. Simons, and J.-H. You. Unfolding partiality and disjunctions in stable model semantics. *ACM Transactions on Computational Logic*, 7(1):1–37, 2006.
10. J. Lee and F. Lin. Loop formulas for circumscription. *Artificial Intelligence*, 170:160–185, 2006.
11. V. Lifschitz. Circumscription. In *Handbook of Logic in AI and Logic Programming*, volume 3, pages 298–352. Oxford University Press, Oxford, 1994.
12. V. Lifschitz. Twelve definitions of a stable model. In *ICLP 2008*, volume 5366 of *LNCS*, pages 37–51. Springer, 2008.
13. F. Lin. *A Study of Nonmonotonic Reasoning*. PhD thesis, Stanford Univ., 1991.
14. J. McCarthy. Circumscription – a form of non-monotonic reasoning. *Artificial Intelligence*, 13:27–39, 1980.
15. T. Przymusinski. Every logic program has a natural stratification and an iterated fixed point model. In *PODS 89*, pages 11–21. ACM SIGACT-SIGMOD, 1989.
16. T. Przymusinski. Well-founded semantics coincides with three-valued stable semantics. *Fundamenta Informaticae*, 13(4):445–464, 1990.
17. N. Rescher. *Many-Valued Logic*. McGraw-Hill, New York, 1969.
18. K. Stenning and M. van Lambalgen. Semantic interpretation as computation in nonmonotonic logic: The real meaning of the suppression task. *Cognitive Science*, (29):916–960, 2005.
19. K. Stenning and M. van Lambalgen. *Human Reasoning and Cognitive Science*. MIT Press, Cambridge, MA, 2008.
20. 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, 2011.
21. C. Wernhard. The globally weakest sufficient condition as basis for abduction in logic programming. Unpublished manuscript, 2012.
22. C. Wernhard. Projection and scope-determined circumscription. *Journal of Symbolic Computation*, 47:1089–1108, 2012.