2015 IEEE Symposium Series on Computational Intelligence

A Computational Logic Approach to Human Spatial Reasoning

Emmanuelle-Anna Dietz, Steffen Hölldobler, Raphael Höps*
International Center for Computational Logic, TU Dresden, Germany

***Abstract*—We present a new approach with respect to spatial reasoning problems by using logic programs. Because the weak completion of a logic program admits a least model under the three-valued Łukasiewicz semantics and this semantics has been successfully applied to other human reasoning tasks, conditionals are evaluated under these least Ł-models. We show that the weak completion semantics can also handle spatial relations in a way humans do. In particular, we develop a computational logic approach to spatial reasoning and show that the weak completion semantics computes preferred mental models.**

I. Introduction

In the last century the classical (propositional) logic has played an important role as a normative concept for psychologists investigating human reasoning episodes. Psychological research, however, showed that humans systematically deviate from the classical logically correct answers.

Until now there are no widely accepted theories that express formal representations of human reasoning. Before we can formalize human behavior we need to understand how humans draw certain conclusions given some specific information. We address this issue by exploring current results from the literature. Conventional formal approaches such as classical logic are not appropriate for this purpose because they cannot deal with the elementary aspects that humans permanently need to reason with, for instance with incomplete information or non-monotonicity. The probably most well-known psychological experiment showing this is Byrne's suppression task [3]. Various formal alternatives have been proposed. However, most of them are purely theoretical and have never been applied to real case studies. Instead, for most of these theories, artificial examples have been constructed, which only show that this theory works within this very specific context. But what is the value of a theory for human reasoning that has never been tested on how humans actually reason?

Strube [29] claims that just modeling is not satisfying. He argues that knowledge engineering should also aim at being *cognitively adequate*. Accordingly, when evaluating computational approaches which try to explain human reasoning we insist on assessing their cognitive adequacy.

Adequacy originally has been defined in a linguistic context to compare and explain language theories and their properties, for which there are two different measures: *conceptual adequacy* and *inferential adequacy* [30]. Conceptual adequacy reflects in how far the language represents the content correctly. Inferential adequacy is about the procedural part when the language is applied on the content. Knauff, Rauh Schneider [19] and Knauff, Rauh and Renz [18] define cognitive adequacy in the setting of qualitative spatial reasoning, where they make a similar distinction: The degrees of conceptual adequacy reflects to which extent a system corresponds to human conceptual knowledge. Inferential adequacy focuses on the procedural part and indicates whether the reasoning process of a system is structured similarly to the way humans reason. This is analogous to the proposition made by Stenning and van Lambalgen in [27] and [28] to model human reasoning by a two step process: Firstly, human reasoning should be modeled by setting up an *appropriate representation* (conceptual adequacy) and, secondly, the *reasoning process* should be modeled with respect to this representation (inferential adequacy).

In the following we will focus on human spatial reasoning, where we investigate and formalize how new knowledge is understood given some information about a certain arrangement of objects. For instance, given the information, that a ferrari is left of a porsche and right of the porsche there is a beetle, we can without difficulty conclude that the ferrari is left of the beetle. But how exactly do we come to this conclusion? What happens if we have a set of premises with which more than one arrangement is possible? Consider the following example taken from Ragni and Knauff [25] which consists of four premises above the line and a conclusion below the line.

1. The ferrari is left of the porsche.
2. The beetle is right of the porsche.
3. The porsche is left of the hummer.
4. The hummer is left of the dodge.

C. The porsche is (necessarily)[1] left of the dodge.

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

The question which we shall be discussing in this paper is how to automatically construct what humans may have in mind while reading the premises. Accordingly, the conclusion should be evaluated in a similar way as humans do. For instance, the mental model theory [16] assumes that humans construct one model, verify the conclusion and possibly construct the next one. On the other hand, the preferred model theory [25] claims that humans only have exactly one so-called preferred model in mind. This requires less effort than constructing all possible models and in most situations of our everyday life this one model is absolutely enough to reason with. According to Ragni and Knauff [25], people will only think of alternatives,

*{dietz,sh}@iccl.tu-dresden.de. Authors are mentioned alphabetically.

[1]This means that the conclusion follows from all possible solutions.

978-1-4799-7560-0/15 $31.00
DOI 10.1109/SSCI.2015.229

IEEE computer society

if they are asked to search for other models. But even then, they do not randomly construct them from scratch. Instead of that, they first start generating models which are most similar to the preferred one by changing them as little as possible. This theory seems to be very promising and is empirically supported. Furthermore, it can give us an explanation about how people deal with ambiguity when modeling spatial reasoning problems and that, while evaluating a conclusion, they might come to conclusions which are wrong according to classical logic.

We will formalize the preferred model theory in a computational logic setting based on the *weak completion semantics*. This approach has its origin in the work by Stenning and van Lambalgen [28]. Their work was based on the Kripke's and Kleene's three-valued logic [17] and contained a technical bug which was corrected in [12] by considering the three-valued Łukasiewicz logic [21] instead. [12] also showed that each logic program as well as its weak completion admits a least model under Łukasiewicz logic, which can be computed by iterating Stenning and van Lambalgen's semantic operator [28]. Consequently, reasoning should be performed with respect to this least model.

Somewhat surprisingly, under the weak completion semantics we were able to adequately model the suppression task [5], [14] as well as the selection task [6], the belief-bias effect [24], contextual abductive reasoning with side effects [23] and conditionals [4]. Moreover, the approach can be implemented within a connectionist setting based on the core method as described in [1] and [13].

Our goal is to show that human spatial reasoning can also be adequately modeled under the weak completion semantics. In order to do so, we first discuss spatial relations in human reasoning and, in particular, the preferred model theory in Section II. Thereafter, we introduce the computational logic approach based on the weak completion semantics in Section III. Finally, Section IV shows how the preferred model theory can be implemented under the weak completion semantics.

II. Spatial Relations in Human Reasoning

We will first address the spatial reasoning problem following the introduction of [25]. We assume binary spatial relations between two objects and restrict ourselves in this paper to the *left* and *right* relations. We use the notation $\mathit{left}(X,Y)$ and $\mathit{right}(X,Y)$ to express that X is left of Y and X is right of Y, respectively. Reconsidering the example from the introduction we obtain:

Example 1:

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

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

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

A. Inference Rule Approach

The *inference rule approach*, as presented by Byrne and Johnson-Laird [2], is based on following assumptions:

1) Humans know a set of inference rules, which they can apply on the premises of the spatial reasoning problem in order to derive new knowledge.
2) In case they encounter the conclusion, it is proven. Only when all possibilities of applying the rules are exploited without proving the conclusion, the conclusion is refuted.
3) The order of the premises is not important.

In [2], a system with nine inference rules for spatial reasoning problems is specified, which does not only consider the *left* and *right* relations, but also the *frontOf* relation. As in this paper we will restrict ourselves to the *left* and the *right* relation, we will only consider a simplified version of the rule system. The relevant inference rules are the following:

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

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

B. Mental Model Theory

The *mental model theory* [16] does not assume that humans apply inference rules but that they construct so-called *mental models*. In spatial reasoning, a mental model is understood as the presentation of the spatial arrangements between objects that correspond to the premises. Consider the following example, again taken from [25], which is similar to Example 1 except from the third premise:

Example 2:

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

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

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

$\mathit{ferrari}\quad \mathit{porsche}\quad \mathit{beetle}\quad \mathit{hummer}\quad \mathit{dodge}$

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

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

The mental model theory assumes that when solving a spatial reasoning problem, humans behave as follows:

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

In this approach, deterministic problems are easier to solve than non-deterministic ones as then Step 3. can be omitted.

In the original version of the mental model theory as well as in the inference rule approach it is assumed that the order of the premises is irrelevant. However, [26] already refers to the studies made by [22] which show that the order actually influences the construction of mental models and, in particular, the construction of the first mental model in Step 1.

C. Preferred Model Theory

The preferred model theory [25] is based on the mental model theory. The major difference is the assumption that humans do not consider all but only certain mental models. These certain ones in turn depend on the order in which the premises are presented. In this section we will discuss this theory together with the computational system PRISM, developed by Ragni and Knauff [25] as well.

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

The preferred model theory assumes that the phases in which humans solve spatial reasoning problems, can be divided into a model construction, a model inspection and a model variation phase. The authors in [25] also present an implementation of the preferred model theory, PRISM. It only allows the following four types of premises for a spatial reasoning problem:

Type 1 Exactly the first premise.
Type 2 Premises containing exactly one new object and one which occurs already in the model so far.
Type 3 Premises, which contain two new objects, but which are not from type 1.
Type 4 Premises, which relate two objects occurring in two different *submodels* to each other, where submodels are already arrangements of objects which are not yet in relation to each other.

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

1) If the premise is of type 1 then the two objects will be placed directly next to each other.
2) If the premise is of type 2 then the new object will be inserted directly next to the already existing one provided that the space next to the existing one is free. If this space is already occupied then the new object is placed in the next available space; This is called *first free fit* or *3f-strategy*.
3) If the premise is of type 3 then a new arrangement – i.e., a new submodel – is constructed in which both objects are arranged directly next to each other.
4) If the premise is of type 4, the first arrangement will be placed directly next to the second arrangement; the object within these arrangements do not change their places.

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

$\mathit{ferrari} \quad \mathit{porsche}$

After reading the second premise, $\mathit{right}(\mathit{beetle}, \mathit{porsche})$, which is of type 2, the beetle is added next to the porsche as its right neighbor:

$\mathit{ferrari} \quad \mathit{porsche} \quad \mathit{beetle}$

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

$\mathit{ferrari} \quad \mathit{porsche} \quad \mathit{beetle} \quad \mathit{hummer}$

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

$\mathit{ferrari} \quad \mathit{porsche} \quad \mathit{beetle} \quad \mathit{hummer} \quad \mathit{dodge}$

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

$\mathit{ferrari} \quad \mathit{porsche} \quad \mathit{hummer} \quad \mathit{beetle} \quad \mathit{dodge}$
$\mathit{ferrari} \quad \mathit{porsche} \quad \mathit{hummer} \quad \mathit{dodge} \quad \mathit{beetle}$

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

Let us consider yet another example.

Example 3:

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

After reading the first premise, the ferrari and the porsche are placed next to each other:

$\mathit{ferrari} \quad \mathit{porsche}$

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

$\mathit{hummer} \quad \mathit{beetle}$

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

$$\mathit{ferrari} \quad \mathit{porsche} \quad \mathit{hummer} \quad \mathit{beetle}$$

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

$$\mathit{ferrari} \quad \mathit{hummer} \quad \mathit{beetle} \quad \mathit{porsche}$$

In this model, the conclusion does not hold anymore. However, the preferred model theory assumes that the majority of the people believes that the conclusion does hold. As we aim at modeling human reasoning, the first model corresponds to the conclusion, which we intend to model.

III. The Weak Completion Semantics

We assume the reader to be familiar with logic and logic programming, but we repeat basic notions and notations, which are based on [20] and [11]. A *(logic) program* is a finite set of (program) clauses of the form $A \leftarrow B_1 \wedge \ldots \wedge B_n$ where A is an atom and B_i, $1 \leq i \leq n$, are literals or of the form $\top$ and $\bot$, denoting truth- and falsehood, respectively. A is called *head* and $B_1 \wedge \ldots \wedge B_n$ is called *body* of the clause. We restrict terms to be constants and variables only, i.e., we consider so-called *data logic programs*. Clauses of the form $A \leftarrow \top$ and $A \leftarrow \bot$ are called *positive* and *negative facts*, respectively.

Let us clarify the notation defined until now by Example 1 from the previous section. The program, which contains all four premises is denoted as a set of positive facts about the *left* or *right* relation between the objects:

$$\begin{array}{lll} \mathcal{P}_{ex} = \{ \mathit{left}(\mathit{ferrari}, \mathit{porsche}) & \leftarrow & \top, \\ \quad \mathit{left}(\mathit{porsche}, \mathit{beetle}) & \leftarrow & \top,^2 \\ \quad \mathit{left}(\mathit{porsche}, \mathit{hummer}) & \leftarrow & \top, \\ \quad \mathit{left}(\mathit{hummer}, \mathit{dodge}) & \leftarrow & \top, \\ \quad \mathit{right}(X, Y) & \leftarrow & \mathit{left}(Y, X) \ \}^3 \end{array}$$

where the last clause is not a fact but simply a clause representing the symmetry between the *left* and *right* relations. Assume that the ferrari would actually not be left of the porsche. We could represent this as a negative fact:

$$\mathit{left}(\mathit{ferrari}, \mathit{porsche}) \leftarrow \bot.$$

In this paper we assume for each program that the alphabet consists precisely of the symbols mentioned in the program. When writing sets of literals we will omit curly brackets if the set has only one element.

Let $\mathcal{P}$ be a program. $g\mathcal{P}$ denotes the set of all ground instances of clauses occurring in $\mathcal{P}$. As the set of constants, predicate symbols and variables is finite, $\mathcal{P}$ is finite, and thus $g\mathcal{P}$ is finite as well. If $\mathcal{P}$ is a program, then $con(\mathcal{P})$ denotes the set of all constants occurring in $\mathcal{P}$. Consider again $\mathcal{P}_{ex}$:

$$con(\mathcal{P}_{ex}) = \{\mathit{ferrari}, \mathit{porsche}, \mathit{beetle}, \mathit{hummer}, \mathit{dodge}\}.$$

[2] Strictly speaking, we should write $\mathit{right}(\mathit{beetle}, \mathit{porsche}) \leftarrow \top$ instead. However, for a simplified representation of the programs in the following, we will only allow *left* relations as facts in the programs. Note that, because of the last clause, we can still conclude the corresponding *right* relation.

[3] This program is preliminary and does not contain all necessary clauses yet.

A ground atom A is *defined* in $g\mathcal{P}$ if and only if (in the following abbreviated with iff) $g\mathcal{P}$ contains a clause whose head is A; otherwise A is said to be *undefined*. Let $\mathcal{S}$ be a set of ground literals.

$$\mathit{def}(\mathcal{S}, \mathcal{P}) = \{A \leftarrow \mathit{body} \in g\mathcal{P} \mid A \in \mathcal{S} \vee \neg A \in \mathcal{S}\}$$

is called *definition* of $\mathcal{S}$. For instance

$$\begin{array}{l} \mathit{def}(\mathit{left}(\mathit{ferrari}, \mathit{porsche}), \mathcal{P}_{ex}) \\ = \ \{\mathit{left}(\mathit{ferrari}, \mathit{porsche}) \leftarrow \top\} \end{array}$$

On the other hand, $\mathit{def}(\mathit{left}(\mathit{porsche}, \mathit{porsche}), \mathcal{P}_{ex})$ is empty.

Let $\mathcal{P}$ be a program and consider the following transformation:

1) For each defined atom A, replace all clauses of the form $A \leftarrow \mathit{body}_1, \ldots, A \leftarrow \mathit{body}_m$ occurring in $g\mathcal{P}$ by $A \leftarrow \mathit{body}_1 \vee \ldots \vee \mathit{body}_m$.
2) If a ground atom A is undefined in $g\mathcal{P}$, then add $A \leftarrow \bot$.
3) Replace all occurrences of $\leftarrow$ by $\leftrightarrow$.

The ground set of formulas obtained by this transformation is called *completion* of $\mathcal{P}$, whereas the ground set of formulas obtained by applying only the steps 1. and 3. is called *weak completion* of $\mathcal{P}$ or $wc\mathcal{P}$. The weak completion of $\mathcal{P}_{ex}$ is:

$$\begin{array}{l} \{ \ \mathit{left}(\mathit{ferrari}, \mathit{porsche}) \leftrightarrow \top, \ \mathit{left}(\mathit{porsche}, \mathit{beetle}) \leftrightarrow \top, \\ \quad \mathit{left}(\mathit{porsche}, \mathit{hummer}) \leftrightarrow \top, \ \mathit{left}(\mathit{hummer}, \mathit{dodge}) \leftrightarrow \top\} \\ \cup \{\mathit{right}(o_1, o_2) \leftrightarrow \mathit{left}(o_2, o_1) \mid o_1, o_2 \in con(\mathcal{P}_{ex})\} \end{array}$$

We consider the three-valued Łukasiewicz (or Ł-) logic [21] (see Table I) and represent each *interpretation* I, i.e., each mapping from the set of formulas to the set of truth values, by a pair $\langle I^\top, I^\bot \rangle$ of ground atoms, where $I^\top = \{A \mid I(A) = \mathit{true}\}$, $I^\bot = \{A \mid I(A) = \mathit{false}\}$, $I^\top \cap I^\bot = \emptyset$. Atoms occurring neither in $I^\top$ nor in $I^\bot$ are mapped to *unknown*. Let $I = \langle I^\top, I^\bot \rangle$ and $J = \langle J^\top, J^\bot \rangle$ be two interpretations:

$$I \subseteq J \quad \text{iff} \quad I^\top \subseteq J^\top \text{ and } I^\bot \subseteq J^\bot.$$

A *model* of $\mathcal{P}$ is an interpretation which maps each clause occurring in $\mathcal{P}$ to *true*. I is the *least model* of $\mathcal{P}$ iff for any other model J of $\mathcal{P}$ it holds that $I \subseteq J$.

Under Ł-logic we find $F \wedge \top \equiv F \vee \bot \equiv F$ for each formula F, where $\equiv$ denotes semantic equivalence. Hence, occurrences of the symbols $\top$ and $\bot$ in the bodies of clauses can be restricted to those occurring in facts.

One should observe that in contrast to two-valued logic, $A \leftarrow B$ and $A \vee \neg B$ are not semantically equivalent under Ł-logic. Consider, for instance, an interpretation I such that $I(A) = I(B) = \mathit{unknown}$. Then, $I(A \vee \neg B) = \mathit{unknown}$ whereas $I(A \leftarrow B) = \mathit{true}$. It has been shown in [12] that logic programs as well as their weak completions admit a least model under Ł-logic. Moreover, the least Ł-model of $wc\mathcal{P}$ can be obtained as least fixed point of the following semantic operator, which is due to Stenning and van Lambalgen [28]: $\Phi_{\mathcal{P}}(\langle I^\top, I^\bot \rangle) = \langle J^\top, J^\bot \rangle$, where

$$\begin{array}{l} J^\top = \{A \mid A \leftarrow \mathit{body} \in \mathit{def}(A, \mathcal{P}) \text{ and } I(\mathit{body}) = \mathit{true}\}, \\ J^\bot = \{A \mid \mathit{def}(A, \mathcal{P}) \neq \emptyset \text{ and} \\ \qquad\qquad I(\mathit{body}) = \mathit{false} \text{ for all } A \leftarrow \mathit{body} \in \mathit{def}(A, \mathcal{P})\}. \end{array}$$

The *weak completion semantics* (WCS) is the approach to consider weakly completed logic programs and to reason with respect to the least Ł-models of these programs. We write

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

$\leftarrow$	$\top$	U	$\bot$
$\top$	$\top$	$\top$	$\top$
U	U	$\top$	$\top$
$\bot$	$\bot$	U	$\top$

$\leftrightarrow$	$\top$	U	$\bot$
$\top$	$\top$	U	$\bot$
U	U	$\top$	U
$\bot$	$\bot$	U	$\top$

TABLE I. TRUTH TABLES FOR THE Ł-SEMANTICS, WHERE WE HAVE USED $\top$, $\bot$ AND U INSTEAD OF *true*, *false* AND *unknown*, RESPECTIVELY.

$\mathcal{P} \models_{wcs} F$ iff formula F holds in the least Ł-model of $wc\mathcal{P}$. As least Ł-model of $wc\mathcal{P}_{ex}$ we obtain $\langle I^\top, \emptyset\rangle$, where

$$I^\top = \{\ \mathit{left}(\mathit{ferrari}, \mathit{porsche}), \mathit{left}(\mathit{porsche}, \mathit{beetle}), \mathit{left}(\mathit{porsche}, \mathit{hummer}), \mathit{left}(\mathit{hummer}, \mathit{dodge}), \mathit{right}(\mathit{porsche}, \mathit{ferrari}), \mathit{right}(\mathit{beetle}, \mathit{porsche}), \mathit{right}(\mathit{hummer}, \mathit{porsche}), \mathit{right}(\mathit{dodge}, \mathit{hummer})\}.$$

The Φ operator differs from the semantic operator defined by Fitting [8] in the additional condition $\mathit{def}(A, \mathcal{P}) \neq \emptyset$ required in the definition of $J^\bot$. This condition states that A must be defined in order to be mapped to *false*, whereas in the Kripke-Kleene-semantics considered by Fitting an atom may be mapped to *false* even if it is undefined in the underlying program. This reflects precisely the difference between the weak completion and the completion semantics, namely in Step 2. of the program transformation. The Kripke-Kleene-semantics was also applied in [28]. However, as shown in [12] this semantics is not only the cause for a technical bug in one theorem of [28], but it also leads to a non-adequate model of some human reasoning episodes. Both, the technical bug as well as the non-adequate modeling, is avoided by using WCS.

As shown in [7], WCS is related to the well-founded semantics (WFS) as follows: Let $\mathcal{P}$ be a program which does not contain a positive loop and let

$$\mathcal{P}^+ = \mathcal{P} \setminus \{A \leftarrow \bot \mid A \leftarrow \bot \in \mathcal{P}\}.$$

Let u be a new nullary relation symbol not occurring in $\mathcal{P}$ and B be a ground atom in

$$\mathcal{P}^* = \mathcal{P}^+ \cup \{B \leftarrow u \mid \mathit{def}(B, \mathcal{P}) = \emptyset\} \cup \{u \leftarrow \neg u\}.$$

Then, the least Ł-model of $wc\mathcal{P}$ and the well-founded model for $\mathcal{P}^*$ coincide. The programs specified in [5] and in [6] to model the suppression and the selection task, respectively are acyclic and, thus, tight. Therefore, our results hold for both, WCS and WFS. The programs presented in the sequel of this paper are not tight. However, the positive cycles in our programs do not have any effect on the results as has been shown by Höps [15] and therefore they also hold for WFS.

IV. HUMAN SPATIAL REASONING UNDER WCS

Following the preferred model theory, we show how the preferred mental model of a spatial reasoning problem can be computed by logic programs under WCS. This approach covers the model construction and the model inspection phase.

A. Preferred Mental Models in Logic Programs

The running example in Section III shows us that relations between objects can be easily represented in logic programs. However, there is no straightforward way in which we can express the order in which the premises are read. But exactly this information is crucial if we want to formalize the preferred model theory. For this purpose, in the approach we will propose now, we explicitly express phases, where each premise is read at one particular phase. This allows us to define the order in which the premises are processed. In contrast to PRISM, we do not distinguish between the model construction and model inspection phase but process them at the same time.

Let $\mathcal{S}$ be a spatial reasoning problem. The program $\mathcal{P}_\mathcal{S}$ represents the premises of $\mathcal{S}$ and the necessary background knowledge in order to construct the preferred mental model. $\mathcal{P}_\mathcal{S}$ is called a PMM-program with respect to the problem $\mathcal{S}$. Within $\mathcal{P}_\mathcal{S}$ we will use the following notation with informal meaning as follows:

$l(X, Y, i)$	in phase i, X is placed to the left of Y,
$ln(X, Y, i)$	in phase i, X is the left neighbor of Y,
$ol(X, i)$	in phase i, directly left of X is occupied,
$or(X, i)$	in phase i, directly right of X is occupied,

where i starts with 1. In the following, n indicates the number of premises processed so far. Given a spatial reasoning problem $\mathcal{S}$, the corresponding program $\mathcal{P}_\mathcal{S}$ is constructed as follows.

1) We start by reading the premises. For each premise do: If the ith premise is of the form $\mathit{left}(o_1, o_2)$ or $\mathit{right}(o_1, o_2)$, then add

$$l(o_1, o_2, i) \leftarrow \top \quad \text{or} \quad l(o_2, o_1, i) \leftarrow \top,$$

respectively, to the (initially empty) program $\mathcal{P}_\mathcal{S}$, where o_1 and o_2 are assumed to be different objects.

2) We make a closed world assumption for the l relation in phase 1 as initially nothing is known about the spatial relation of objects.

$$\{l(o_1, o_2, 1) \leftarrow \bot \mid o_1, o_2 \in \mathit{con}(\mathcal{P}_\mathcal{S}), \mathit{diff}(o_1, o_2)\},$$

where diff specifies that its arguments are different objects. One should observe that programs are weakly completed, e.g., if the first premise of a spatial reasoning problem is of the form $\mathit{left}(\mathit{porsche}, \mathit{hummer})$ then the ground facts

$$l(\mathit{porsche}, \mathit{hummer}, 1) \leftarrow \top$$
$$l(\mathit{porsche}, \mathit{hummer}, 1) \leftarrow \bot$$

are generated in the first two steps, respectively. Their weak completion is

$$l(\mathit{porsche}, \mathit{hummer}, 1) \leftrightarrow \top \vee \bot \equiv l(\mathit{porsche}, \mathit{hummer}, 1) \leftrightarrow \top.$$

Under WCS, positive information overwrites negative information. In other words, there is no obligation to place o_1 to the left of o_2 in phase i unless explicitly stated in the ith premise.

3) As at the beginning no objects have been placed, the space to the left and to the right of each object is initially empty:

$$\{ol(o, 1) \leftarrow \bot \mid o \in \mathit{con}(\mathcal{P}_\mathcal{S})\} \cup \{or(o, 1) \leftarrow \bot \mid o \in \mathit{con}(\mathcal{P}_\mathcal{S})\}$$

This corresponds to the closed world assumption with respect to the ol relation, which needs to be explicitly made under WCS. In the running example, we find that the space to the left and to the right of both cars, the porsche and the hummer, are empty in phase 1:

$$\{\ ol(porsche, 1) \leftarrow \bot,\ \ or(porsche, 1) \leftarrow \bot,\ \ ol(hummer, 1) \leftarrow \bot,\ \ or(hummer, 1) \leftarrow \bot\ \}$$

4) We start to place objects. If in phase i object o_1 should be placed to the left of object o_2 and the space to the left of o_1 as well as the space to the right of o_1 are empty, then o_1 is placed as the left neighbor of o_2:

$$\{ln(o_1, o_2, i) \leftarrow l(o_1, o_2, i) \wedge \neg ol(o_2, i) \wedge \neg or(o_1, i) \mid o_1, o_2 \in con(\mathcal{P}_\mathcal{S}), diff(o_1, o_2), i \in [1, n]\}$$

In the running example we obtain for phase 1:

$$ln(porsche,\ hummer, 1) \leftarrow l(porsche, hummer, 1) \wedge\ \neg ol(hummer, 1) \wedge \neg or(porsche, 1)$$

among others. Given 1., 2. and 3., the body of this clause will be *true* and, consequently, the porsche will be placed as the left neighbor of the hummer in phase 1.

5) Once an object o_1 has become the left neighbor of another object o_2 in phase i, this relation holds until the preferred mental model is constructed:

$$\{ln(\ o_1, o_2, i+1) \leftarrow\ ln(o_1, o2, i) \mid o_1, o_2 \in con(\mathcal{P}_\mathcal{S}), diff(o_1, o_2), i \in [1, n-1]\}$$

6) If o_1 has become the left neighbor of o_2 in phase i, then the space to the left of o_2 as well as the space to the right of o_1 are occupied in phase $i+1$:

$$\{ol(o_2, i+1) \leftarrow\ ln(o_1, o_2, i) \mid o_1, o_2 \in con(\mathcal{P}_\mathcal{S}), diff(o_1, o_2), i \in [1, n-1]\} \cup \{or(\ o_1, i+1) \leftarrow\ ln(o_1, o_2, i) \mid\ o_1, o_2 \in con(\mathcal{P}_\mathcal{S}), diff(o_1, o_2), i \in [1, n-1]\}$$

In combination with 5. the space to the left of o_2 and the space to the right of o_1 are occupied in all future phases. For example, after the porsche has been placed as left neighbor of the hummer in phase 1, the clauses

$$ol(hummer, 2) \leftarrow ln(porsche, hummer, 1)$$
$$or(porsche, 2) \leftarrow ln(porsche, hummer, 1)$$

determine that there is no space anymore immediately to the left of the hummer and immediately to the right of the porsche at phase 2.

7) If o_1 should be placed to the left of o_2 but there is already a left neighbor o_3 of o_2, then o_1 is placed to the left of o_3:

$$\{l\,(o_1, o_3, i+1) \leftarrow l(o_1, o_2, i+1) \wedge ln(o_3, o_2, i) \mid o_1, o_2, o_3 \in con(\mathcal{P}_\mathcal{S}), diff(o_1, o_2, o_3), i \in [1, n-1]\}$$

One should observe that this can only happen from phase 2 onwards, as in the first phase none of the objects has a left neighbor. This is the reason for writing $i+1$ in the atom $l(o_1, o_2, i+1)$ occurring in the bodies of the clauses.

8) Likewise, if o_1 should be placed to the left of o_2 but o_1 is already the left neighbor of some other object o_3, then o_3 should be placed to the left of o_2:

$$\{l\,(o_3, o_2, i+1) \leftarrow l(o_1, o_2, i+1) \wedge ln(o_1, o_3, i) \mid o_1, o_2, o_3 \in con(\mathcal{P}_\mathcal{S}), diff(o_1, o_2, o_3), i \in [1, n-1]\}$$

9) Finally, in order to determine whether the conclusion is true, we add the following clauses to $\mathcal{P}_\mathcal{S}$. If o_1 is the left neighbor of o_2 after processing all premises, then o_1 is to the left of o_2 in the preferred mental model:

$$\{left(o_1, o_2) \leftarrow ln(o_1, o_2, n) \mid o_1, o_2 \in con(\mathcal{P}_\mathcal{S}), diff(o_1, o_2)\}$$

10) The $left$ relation is transitive:

$$\{left(o_1, o_3) \leftarrow left(o_1, o_2) \wedge left(o_2, o_3) \mid o_1, o_2, o_3 \in con(\mathcal{P}_\mathcal{S}), diff(o_1, o_2, o_3)\}$$

11) The $right$ relation is the inverse of the $left$ relation:

$$\{right(o_1, o_2) \leftarrow left(o_2, o_1) \mid\ o_1, o_2 \in con(\mathcal{P}_\mathcal{S}), diff(o_1, o_2)\}$$

In each phase, one premise is read and understood as a request to place the mentioned objects in the required order. Objects are placed in the first available space like in the PRISM approach, where again in each phase exactly one request to place objects is processed and the objects in the request are placed. Once the least fixed point of $\Phi_{\mathcal{P}_\mathcal{S}}$ has been reached we can identify the preferred mental model: Given a problem $\mathcal{S}$, o_1 is the left neighbor of o_2 iff $ln(o_1, o_2, n)$ holds in the least fixed point. Additionally, queries involving the $left$ and $right$ relation can now be answered with respect to the preferred mental model of $\mathcal{S}$. This will be illustrated by two examples in the next subsection.

B. Examples

We consider the following spatial reasoning problem:

Example 4:

1.	$left(porsche, hummer)$
2.	$left(dodge, hummer)$
$C.$	$left(dodge, porsche)$

Let $\mathcal{P}_4$ be the logic program corresponding to Example 4 and $\Phi_{\mathcal{P}_4}$ the corresponding semantic operator. To save space we abbreviate the constants representing cars by their first letter, i.e., d, h and p are abbreviations for $dodge$, $hummer$ and $porsche$, respectively. In Table II we illustrate the computation of the least fixed point of $\Phi_{\mathcal{P}_4}$ step by step, where $\Phi_{\mathcal{P}_4} \uparrow n$ denotes I after the nth iteration of $\Phi_{\mathcal{P}_4}$, $\Phi \uparrow 0 = \langle \emptyset, \emptyset \rangle$ and $\Phi \uparrow (i+1) = \Phi(\Phi \uparrow i)$ for all $i > 0$. Focusing on atoms which are mapped to true, i.e., on $I^\top$, we find:

- In the first iteration of the $\Phi_{\mathcal{P}_4}$ operator the requests to place the porsche to the left of the hummer in phase 1 and the dodge to the left of the hummer in phase 2 are recorded.
- In the second iteration of $\Phi_{\mathcal{P}_4}$ the porsche becomes the left neighbor of the hummer in phase 1.
- In the third iteration of $\Phi_{\mathcal{P}_4}$ we learn that the space to the left of the hummer as well as the space to the right of the porsche are occupied in phase 2. As the porsche is the left neighbor of the hummer in phase 1, this relationship is preserved in phase 2 and the dodge must be placed to the left of the porsche in phase 2.
- In the forth iteration of $\Phi_{\mathcal{P}_4}$ the dodge becomes the left neighbor of the porsche in phase 2 and we find that the porsche and the hummer are in the $left$ relation.
- In the fifth iteration of $\Phi_{\mathcal{P}_4}$ we find that the dodge and the porsche are in the $left$ relation, whereas the hummer and the porsche are in the $right$ relation.

iteration	$I^\top$	$I^\perp$	#
$\Phi_{\mathcal{P}_4}\uparrow 0$	$\emptyset$	$\emptyset$	
$\Phi_{\mathcal{P}_4}\uparrow 1$	$\boldsymbol{l(p,h,1)}$		1.
	$\boldsymbol{l(d,h,2)}$		1.
		$l(d,h,1), l(d,p,1), l(h,d,1)$	2.
		$l(h,p,1), l(p,d,1)$	2.
		$ol(d,1), ol(h,1), ol(p,1)$	3.
		$or(d,1), or(h,1), or(p,1)$	3.
$\Phi_{\mathcal{P}_4}\uparrow 2$	$\boldsymbol{ln(p,h,1)}$	$ln(d,h,1), ln(d,p,1), ln(h,d,1)$	4.
		$ln(h,p,1), ln(p,d,1)$	4.
$\Phi_{\mathcal{P}_4}\uparrow 3$	$ln(p,h,2)$		5.
	$ol(h,2)$	$ol(d,2), ol(p,2)$	6.
	$or(p,2)$	$or(d,2), or(h,2)$	6.
	$\boldsymbol{l(d,p,2)}$	$l(h,p,2), l(p,d,2), l(p,h,2)$	7.,8.
$\Phi_{\mathcal{P}_4}\uparrow 4$	$\boldsymbol{ln(d,p,2)}$	$ln(d,h,2), ln(h,p,2), ln(p,d,2)$	4.
		$l(h,d,2)$	7.,8.
	$left(p,h)$		9.
$\Phi_{\mathcal{P}_4}\uparrow 5$		$ln(h,d,2)$	4.
	$\boldsymbol{left(d,p)}$		9.
	$right(h,p)$		11.
$\Phi_{\mathcal{P}_4}\uparrow 6$	$left(d,h)$		10.
	$right(p,d)$		11.
$\Phi_{\mathcal{P}_4}\uparrow 7$	$right(h,d)$		11.

TABLE II. THE COMPUTATION OF THE LEAST FIXED POINT OF $\Phi_{\mathcal{P}_4}$, WHERE IN EACH ITERATION ONLY ATOMS ARE LISTED WHICH APPEAR IN $I^\top$ AND $I^\perp$ FOR THE FIRST TIME. # LISTS THE CLAUSES RESPONSIBLE FOR ADDING AN ATOM TO $I^\top$ OR $I^\perp$.

- In the sixth iteration of $\Phi_{\mathcal{P}_4}$ we find by transitivity that the dodge and the hummer are in the *left* relation, whereas the porsche and the dodge are in the *right* relation.
- Finally, in the seventh iteration of $\Phi_{\mathcal{P}_4}$ we find that the hummer and the dodge are in the *right* relation.

C holds in the preferred model; the dodge is to the left of the porsche.

We return to Example 3 from Section II-C which contains premises from type 3 and 4, i.e., premises that generate submodels. Let $\mathcal{P}_3$ be the logic program corresponding to Example 3 and $\Phi_{\mathcal{P}_3}$ be the corresponding semantic operator. Again to save space, we abbreviate the constants representing *beetle*, *hummer*, *ferrari* and *porsche*, by their first letter, i.e., b, h, f and p, respectively. In Table III we depict the computation of the least fixed point of $\Phi_{\mathcal{P}_3}$. For $I^\top$ we find:

- In the first iteration the three requests to place objects are recorded.
- In the second and the forth iteration, the ferrari becomes the left neighbor of the porsche and the hummer becomes the left neighbor of the beetle, respectively, thus generating two submodels which are not connected at this step.
- In the fifth and the sixth iteration the request to place the ferrari to the left of the beetle ($l(f,b,3)$) is processed. This generates $l(f,h,3)$ and, thereafter, to $l(p,h,3)$.
- The porsche becomes the left neighbor of the hummer in the seventh iteration leading to the preferred mental model.

C holds in the preferred model; the porsche is to the left of the beetle.

iteration	$I^\top$	$I^\perp$	#
$\Phi_{\mathcal{P}_3}\uparrow 0$	$\emptyset$	$\emptyset$	
$\Phi_{\mathcal{P}_3}\uparrow 1$	$\boldsymbol{l(f,p,1)}$		1.
	$\boldsymbol{l(h,b,2)}$		1.
	$\boldsymbol{l(f,b,3)}$		1.
		$l(b,f,1), l(b,h,1), l(b,p,1), l(f,b,1),$	2.
		$l(f,h,1), l(h,b,1) l(h,f,1), l(h,p,1),$	2.
		$l(p,b,1), l(p,f,1), l(p,h,1)$	2.
		$ol(b,1), ol(f,1), ol(h,1), ol(p,1)$	3.
		$or(b,1), or(f,1), or(h,1), or(p,1)$	3.
$\Phi_{\mathcal{P}_3}\uparrow 2$	$\boldsymbol{ln(f,p,1)}$	$ln(b,f,1), ln(b,h,1), ln(b,p,1), ln(f,b,1)$	4.
		$ln(f,h,1), ln(h,b,1), ln(h,f,1) ln(h,p,1)$	4.
		$ln(p,b,1), ln(p,f,1), ln(p,h,1)$	4.
$\Phi_{\mathcal{P}_3}\uparrow 3$	$ln(f,p,2)$		5.
	$ol(p,2)$	$ol(b,2), ol(f,2), ol(h,2)$	6.
	$or(f,2)$	$or(b,2), or(h,2), or(p,2)$	6.
		$l(b,h,2), l(b,p,2), l(f,b,2), l(f,h,2)$	7.,8.
		$l(f,p,2), l(h,p,2), l(p,b,2)$	7.,8.
$\Phi_{\mathcal{P}_3}\uparrow 4$		$ln(b,h,2), ln(b,p,2), ln(f,b,2), ln(f,h,2)$	4.
	$\boldsymbol{ln(h,b,2)}$	$ln(h,p,2), ln(p,b,2), ln(p,f,2)$	4.
	$ln(f,p,3)$		5.
	$ol(p,3)$		6.
	$or(f,3)$		6.
		$l(b,f,2), l(h,f,2), l(p,h,2)$	7.,8.
$\Phi_{\mathcal{P}_3}\uparrow 5$	$ln(h,b,3)$	$ln(b,p,3), ln(f,b,3), ln(f,h,3)$	5.
		$ln(h,d,2), ln(b,f,2), ln(h,f,2),$	5.
		$ln(h,p,3), ln(p,h,2)$	5.
	$ol(b,3)$		6.
	$or(h,3)$		6.
	$\boldsymbol{l(f,h,3)}$	$l(h,p,3), l(p,f,3)$	7.,8.
	$left(f,p)$		9.
$\Phi_{\mathcal{P}_3}\uparrow 6$		$ln(p,b,3), ln(p,f,3)$	5.
		$ol(f,3), ol(h,3), or(b,3), or(p,3)$	6.
	$\boldsymbol{l(p,h,3)}$	$l(b,h,3), l(b,p,3), l(f,p,3),$	7,8.
		$l(h,b,3), l(h,f,3), l(p,b,3), ln(h,f,3)$	7.,8.
	$left(h,b)$		9.
	$right(p,f)$		11.
$\Phi_{\mathcal{P}_3}\uparrow 7$	$\boldsymbol{ln(p,h,3)}$	$ln(b,h,3)$	4.
		$l(b,f,3)$	7.,8.
	$right(b,h)$		11.
$\Phi_{\mathcal{P}_3}\uparrow 8$		$ln(b,f,3)$	5.
	$left(p,h)$		9.
$\Phi_{\mathcal{P}_3}\uparrow 9$	$left(f,h)$		10.
	$\boldsymbol{left(p,b)}$		10.
	$right(h,p)$		11.
$\Phi_{\mathcal{P}_3}\uparrow 10$	$left(f,b)$		10.
	$right(b,p)$		11.
	$right(h,f)$		11.
$\Phi_{\mathcal{P}_3}\uparrow 11$	$right(b,f)$		11.

TABLE III. THE COMPUTATION OF THE LEAST FIXED POINT OF $\Phi_{\mathcal{P}_3}$, WHERE IN EACH ITERATION ONLY ATOMS ARE LISTED WHICH APPEAR IN $I^\top$ AND $I^\perp$ FOR THE FIRST TIME. # LISTS THE CLAUSES RESPONSIBLE FOR ADDING AN ATOM TO $I^\top$ OR $I^\perp$.

V. CONCLUSION

We have shown that our computational logic approach based on the weak completion semantics can compute preferred mental models for spatial reasoning problems. We have restricted our presentation to the *left* and *right* relation, but the formalization can be extended to include additional ones like the *front* or the *back* relations. Likewise, we should be able to handle the four cardinal directions. Different than other approaches such as described in [10], the preferred model theory explains how a model is constructed and seems to be able to predict conclusions humans make given a spatial reasoning problem. This allows us to understand how they influence the model construction, as we have shown by Example 3 and 4.

Höps has shown in [15] that although the logic programs in this paper are not tight anymore, the relationship between the weak completion and the well-founded semantics mentioned in Section III can be preserved and, hence, preferred mental models can also be computed within state-of-the-art reasoning systems based on answer set programming like CLINGO [9]. Thus, large scale applications seem to be feasible.

As shown in [13] the least fixed point of the Φ operator can be computed in a connectionist setting based on the core method [1]. In other words, the core method allows one to compute and to reason with respect to preferred mental models within a purely connectionist network.

It appears that the weak completion semantics can cover a wide range of human reasoning episodes. As already mentioned in the introduction, it can handle the suppression as well as the selection task, the belief-bias effect, contextual abductive reasoning with side effects and indicative conditionals. With the results presented in this paper, these tasks can now be combined with reasoning over spatial relationships. We are unaware of any other logic based approach which covers such a wide range of human reasoning episodes.

ACKNOWLEDGMENTS

Many thanks to Marco Ragni and the reviewers.

REFERENCES

[1] S. Bader and S. Hölldobler. The core method: Connectionist model generation. In S. Kollias, A. Stafylopatis, W. Duch, and E. Oja, editors, *Artificial Neural Networks - ICANN 2006*, volume 4132 of *Lecture Notes in Computer Science*, pages 1–13. Springer Berlin Heidelberg, 2006.

[2] R. Byrne and Johnson-Laird. Spatial reasoning. In *Journal of Memory and Language*, volume 28, pages 564 – 575, 1989.

[3] R. M. J. Byrne. Suppressing valid inferences with conditionals. *Cognition*, 31:61–83, 1989.

[4] E.-A. Dietz and S. Hölldobler. A new computational logic approach to reason with conditionals. In F. Calimeri, G. Ianni, and M. Truszczynski, editors, *Logic Programming and Nonmonotonic Reasoning, 13th International Conference, LPNMR*, volume 9345 of *Lecture Notes in Artificial Intelligence*, pages 265–278. Springer, 2015.

[5] E.-A. Dietz, S. Hölldobler, and M. Ragni. A computational logic approach to the suppression task. In N. Miyake, D. Peebles, and R. P. Cooper, editors, *Proceedings of the 34th Annual Conference of the Cognitive Science Society, CogSci 2013*, pages 1500–1505. Austin, TX: Cognitive Science Society, 2012.

[6] E.-A. Dietz, S. Hölldobler, and M. Ragni. A computational logic approach to the abstract and the social case of the selection task. In *Proceedings of the 11th International Symposium on Logical Formalizations of Commonsense Reasoning, COMMONSENSE 2013*, Aeya Nappa, Cyprus, 2013.

[7] E.-A. Dietz, S. Hölldobler, and C. Wernhard. Modelling the suppression task under weak completion and well-founded semantics. *Journal of Applied Non-Classical Logics*, 24:61–85, 2014.

[8] M. Fitting. A Kripke-Kleene semantics for logic programs. *Journal of Logic Programming*, 2(4):295–312, 1985.

[9] M. Gebser, R. Kaminski, B. Kaufmann, and T. Schaub. *Clingo* = ASP + control: Preliminary report. In M. Leuschel and T. Schrijvers, editors, *Technical Communications of the Thirtieth International Conference on Logic Programming (ICLP'14)*, volume arXiv:1405.3694v1, 2014. Theory and Practice of Logic Programming, Online Supplement.

[10] G. P. Goodwin and P. Johnson-Laird. Reasoning about relations. *Psychological review*, 112(2):468, 2005.

[11] S. Hölldobler. *Logik und Logikprogrammierung 1: Grundlagen*. Kolleg Synchron. Synchron, 2009.

[12] S. Hölldobler and C. D. Kencana Ramli. Logic programs under three-valued Łukasiewicz semantics. In P. M. Hill and D. S. Warren, editors, *Logic Programming, 25th International Conference, ICLP 2009*, volume 5649 of *Lecture Notes in Computer Science*, pages 464–478, Heidelberg, 2009. Springer.

[13] S. Hölldobler and C. D. Kencana Ramli. Logics and networks for human reasoning. In C. Alippi, M. M. Polycarpou, C. G. Panayiotou, and G. Ellinas, editors, *International Conference on Artificial Neural Networks, ICANN 2009, Part II*, volume 5769 of *Lecture Notes in Computer Science*, pages 85–94, Heidelberg, 2009. Springer.

[14] S. Hölldobler, T. Philipp, and C. Wernhard. An abductive model for human reasoning. In *Logical Formalizations of Commonsense Reasoning, Papers from the AAAI 2011 Spring Symposium*, AAAI Spring Symposium Series Technical Reports, pages 135–138, Cambridge, MA, 2011. AAAI Press.

[15] R. Höps. Menschliches räumliches Schließen und Ansätze aus der Computational Logic. Bachelor's thesis, TU Dresden, Germany, 2014.

[16] P. N. Johnson-Laird. *Mental models: towards a cognitive science of language, inference, and consciousness*. Harvard University Press, Cambridge, MA, 1983.

[17] S. C. Kleene. *Introduction to Metamathematics*. North-Holland, Amsterdam, 1952.

[18] M. Knauff, R. Rauh, and J. Renz. A cognitive assessment of topological spatial relations: Results from an empirical investigation. In *Proceedings of the 3rd International Conference on Spatial Information Theory, COSIT'97*, volume 1329 of *Lecture Notes in Computer Science*, pages 193–206, Heidelberg, 1997. Springer.

[19] M. Knauff, R. Rauh, and C. Schlieder. Preferred mental models in qualitative spatial reasoning: A cognitive assessment of Allen's calculus. In *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society, CogSci 1995*, pages 200–205, Hillsdale, NJ, 1995. Lawrence Erlbaum.

[20] J. W. Lloyd. *Foundations of Logic Programming*. Springer-Verlag New York, Inc., New York, NY, USA, 1984.

[21] J. Łukasiewicz. O logice trójwartościowej. *Ruch Filozoficzny*, 5:169–171, 1920. English translation: On three-valued logic. In: Łukasiewicz J. and Borkowski L. (ed.). (1990). *Selected Works*, Amsterdam: North Holland, pp. 87–88.

[22] K. Manktelow. *Reasoning and Thinking*. Psychology press, 2000.

[23] L. M. Pereira, E. Dietz, and S. Hölldobler. Contextual abductive reasoning with side-effects. *Journal of Theory and Practice of Logic Programming (TPLP)*, 14(4-5):633–648, 2014.

[24] L. M. Pereira, E.-A. Dietz, and S. Hölldobler. A computational logic approach to the belief bias effect. In *14th International Conference on Principles of Knowledge Representation and Reasoning (KR 2014)*, 2014. short paper.

[25] M. Ragni and M. Knauff. A theory and a computational model of spatial reasoning with preferred mental models. In *Psychological Review 2013*, volume 120, pages 561 – 588, 2013.

[26] M. Ragni, M. Knauff, and B. Nebel. A computational model for spatial reasoning with mental models. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society*, pages 1797–1802, 2005.

[27] K. Stenning and M. van Lambalgen. Semantic interpretation as computation in nonmonotonic logic: The real meaning of the suppression task. *Cognitive Science*, 6(29):916–960, 2005.

[28] K. Stenning and M. van Lambalgen. *Human Reasoning and Cognitive Science*. A Bradford Book. MIT Press, Cambridge, MA, 2008.

[29] G. Strube. The role of cognitive science in knowledge engineering. In F. Schmalhofer, G. Strube, and T. Wetter, editors, *Contemporary Knowledge Engineering and Cognition, First Joint Workshop*, volume 622 of *Lecture Notes in Computer Science*, pages 159–174. Springer, Heidelberg, 1992.

[30] G. Strube. *Wörterbuch der Kognitionswissenschaft*. Klett-Cotta, Stuttgart, 1996.