UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Pointer semantics with forward propagation

Ghosh, S.; Löwe, B.; Saraf, S.

DOI
10.2991/agi.2009.29
Publication date
2009
Document Version
Final published version
Published in
Advances in Intelligent Systems Research

Link to publication

Citation for published version (APA):
Ghosh, S., Löwe, B., & Saraf, S. (2009). Pointer semantics with forward propagation. *Advances in Intelligent Systems Research*, *8*, 37-42. https://doi.org/10.2991/agi.2009.29

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:26 Jan 2022

Pointer Semantics with Forward Propagation

Sujata Ghosh*
Center for Soft Computing Research
Indian Statistical Institute
Kolkata, West Bengal, India

Benedikt Löwe†
Institute for Logic,
Language and Computation
Universiteit van Amsterdam
1018 TV Amsterdam, The Netherlands

Sanchit Saraf‡
Department of Mathematics and Statistics
Indian Institute of Technology
Kanpur 208016, India

Abstract

In this paper, we will discuss a new approach to formally modelling belief change in systems of sentences with interdependency. Our approach is based on the paradigm called *pointer semantics* or *revision theory* which forms a fundamental way of successfully understanding the semantics of logic programming, but has also been used extensively in philosophical logic and other applications of logic. With a purely unidirectional (backward) flow of change, pointer semantics are not fit to deal with belief change. We propose an extension that allows flow of change in both directions in order to be applied for belief change.

Introduction

Pointer semantics

Pointer semantics are a formal propositional language for finitely many propositions $\{p_0, ..., p_n\}$ defined in terms of each other, and are a well-established tool in logic. The underlying idea of pointer semantics is to

"*iterate away uncertainties and keep the values that stabilize*": (#)

Starting with an initial hypothesis for the truth values that may be in conflict with each other, you apply the Tarskian definition of truth repeatedly and consider the iteration sequence. Those values that stabilize will be kept, the others discarded. The semantics based on (#) have been rediscovered several times independently and have found applications in various areas of foundational study and philosophy.[1] The language of pointer semantics is closely related to logic programming and is the logical reflection of the "Revision Theory of Truth" (GB93). The version of pointer semantics that we shall be using in this paper is essentially that of Haim Gaifman (Gai88; Gai92). The revision rules of pointer semantics let the truth value of a proposition depend on the values of those propositions it points to in the dependency graph (cf. Definition 1), so truth values "flow backward in the dependency graph".

Pointer semantics for belief systems

It has been suggested several times (e.g., cf. (Löw06)) that the underlying principle (#) could be used for other processes involving revision, e.g., the process of revising and updating our beliefs in the face of learning new and unexpected facts. But the mentioned "backward flow" of traditional pointer semantics makes sure that truth values of terminal nodes in the dependency graph (which would typically correspond to those newly learned atomic statements) will never change in the revision process. In the context of belief, we would need *forward propagation* of truth values along the dependency graph. This idea has been implemented in a formal system in (GLS07), but the system proposed by the authors was not easy to handle.

Aims, motivation and related work

Our aim is to provide a new framework for a semantics of belief change based on the general principle (#) using the standard definition of revision semantics (Definition 2). In this paper, we cannot give a motivation of the general framework of revision theory and refer the reader to the extensive philosophical discussion in (GB93) or the survey (Löw06). Our aim is to stay as close as possible to the spirit of that framework.

There are many competing formal frameworks that deal with the question "*How can we model rational belief change?*", most of which far more developed than what can be outlined in this short note. Transition systems and action description languages have been used in (HD05; HD07); there is a rich literature on using probabilistic logics for modelling belief change; in machine learning, dependency networks, Bayesian networks and Markov logic networks have been used (KDR07; HCH03; KD05); an argumentation theoretic approach can be found in (CnS07). This short list only scratches the surface of the vast literature of very good, intuitive and successful answers to our general

---

**Additional affiliation:* Department of Mathematics, Visva-Bharati, Santiniketan, India.

†Additional affiliations: Department Mathematik, Universität Hamburg, Hamburg, Germany; Mathematisches Institut, Rheinische Friedrich-Wilhelms-Universität Bonn, Bonn, Germany.

‡The third author would like to thank the Institute for Logic, Language and Computation (ILLC) of the *Universiteit van Amsterdam* and the *Department Mathematik* of the *Universität Hamburg* for their hospitality during his visit from May to July 2008. All three authors would like to thank Bjarni Hilmarsson (Amsterdam) for programming support.

[1] For more details, cf. the section 'The ubiquity of revision' in (Löw06, § 6).

question. Our approach of staying as close as possible to the pointer semantics paradigm of (#) cannot yet compete at the same level of depth and maturity at the moment. So, we need to explain why one should be interested in a new approach to belief change based on pointer semantics:

Traditional logic approaches to belief update are at the level of axiomatics of what we require of a belief change function, not at the more detailed level of how to actually make the decisions about the adequate belief change. For instance, if an agent believes $\{p \rightarrow q, p\}$ and learns $\{\neg q\}$, then the axiomatic prediction would be that either $p \rightarrow q$ or p has to be dropped, but without further details, it is difficult to predict which one.

As discussed in (Löw06), pointer semantics carries a lot of information transcending the pure definition of the semantics (Definition 2): you can look at how fast stable values stabilize, at the various oscillation patterns of oscillating hypotheses, etc. This information can be used for definitions of *levels* or *types* of stability in order to help with prioritizing, promising to provide new insight in possible belief revision operators. These are tools that might be applicable directly in modelling belief change, or could serve as a subsidiary tool to support other (established) systems of formal belief change models for applications in artificial intelligence.

Overview of this paper

In our section "Definitions", we give the standard Definition 2 from pointer semantics (following (Löw06)) and define an algebra of pointer systems. The latter definition is new to this paper, allowing to state and prove Propositions 3 and 4 in the case of operators restricted to backward propagation. The central new definition of this paper is in the section "Belief Semantics with Forward Propagation". In the section "Properties of our Belief Semantics" we test our system in an example originally used in (GLS07) and finally see that our system is ostensibly non-logical as expected for a system is intended to model systems of belief.[2] We close with a discussion of future work in our final section.

Definitions

Abstract Pointer Semantics

Fix a finite set of propositional variables $\{\mathrm{p}_0, ..., \mathrm{p}_n\}$. An **expression** is just a propositional formula using $\wedge$, $\vee$, and $\neg$ and some of the propositional variables or the empty sequence, denoted by $\llcorner\!\lrcorner$.

We fix a finite algebra of truth values $\mathbb{T}$ with operations $\wedge$, $\vee$ and $\neg$ corresponding to the syntactic symbols. We assume a notion of order corresponding to information content that gives rise to a notion of **infimum** in the algebra of truth values, allowing to form $\inf(X)$ for some subset of $X \subseteq \mathbb{T}$. A truth value will represent the lowest information content (i.e., a least element in the given order); this truth value will be denoted by ½. We allow inf to be applied to the empty set and let $\inf \varnothing :=$ ½.

[2] "The fact that the logic of belief, even rational belief, does not meet principles of truth-functional deductive logic, should no longer surprise us (Gol75, p. 6)."

Our salient example is the algebra $\mathbb{T} := \{0, ½, 1\}$ with the following operations ("strong Kleene"):

$\wedge$	0	½	1
0	0	0	0
½	0	½	½
1	0	½	1

$\vee$	0	½	1
0	0	½	1
½	½	½	1
1	1	1	1

$\neg$	
0	1
½	½
1	0

.

The value ½ stands for ignorance, and thus the infimum is defined as $\inf(\{t\}) := t$, $\inf(\{½\} \cup X) :=$ ½, $\inf(\{0, 1\}) :=$ ½. This algebra of truth values will be used in this paper, even though the set-up in this section is more general.

If E is an expression and p_i is one of the propositional variables, then $\mathrm{p}_i \leftarrow E$ is a **clause**. We intuitively interpret $\mathrm{p}_i \leftarrow E$ as "p_i states E". If $E_0, ..., E_n$ are expressions, a set of clauses $\Sigma := \{\mathrm{p}_0 \leftarrow E_0, ..., \mathrm{p}_n \leftarrow E_n\}$ is called a **pointer system**. An **interpretation** is a function $I\colon \{\mathrm{p}_0, ..., \mathrm{p}_n\} \rightarrow \mathbb{T}$ assigning truth values to propositional letters. Note that if $\mathbb{T}$ is finite, the set of interpretations is a finite set (we shall use this later). A given interpretation I can be naturally extended to a function assigning truth values to all expressions (using the operations $\wedge$, $\vee$ and $\neg$ on $\mathbb{T}$). We denote this extended function with the same symbol I.

A clause can be transformed into an equation in $\mathbb{T}$: if $\mathrm{p}_i \leftarrow E$ is a clause, we can read it as an equation $\mathrm{p}_i = E$ in $\mathbb{T}$. If Q is such an equation, we say that an interpretation I is a **solution** of Q if plugging the values $\{I(\mathrm{p}_0), ..., I(\mathrm{p}_n)\}$ into the corresponding variables of the equation results in the same value left and right of the equals sign. An interpretation is a solution of a set of equations if it is a solution of each equation in the set.

A function mapping interpretations to interpretations is called a **revision function**; a family of these functions indexed by pointer systems is called a **revision operator**. If δ is a revision operator, we write δ_Σ for the revision function assigned to the pointer system Σ (and sometimes just write δ if Σ is clear from the context). We use the usual notation for iteration of revision functions, i.e., $\delta^0(I) := I$, $\delta^{n+1}(I) := \delta(\delta^n(I))$.

Definition 1 *Given a pointer system* $\{\mathrm{p}_0 \leftarrow E_0, ..., \mathrm{p}_n \leftarrow E_n\}$, *we define its* ***dependency graph*** *by letting* $\{0, ..., n\}$ *be the vertices and allowing an edge from* i *to* j *if* p_j *occurs in* E_i.[3]

Given a proposition p_i, arrows point to i from the propositions occurring in E_i, and thus we call a revision operator δ an **B-operator** (for "backward") if the value of $\delta(I)(\mathrm{p}_i)$ only depends on the values of $I(\mathrm{p}_j)$ for p_j occurring in E_i.

Fix Σ and δ. We call an interpretation I **(Σ, δ)-stable** if there is some k such that for all $n \geq k$, $\delta^n(I) = I$. We call I **(Σ, δ)-recurring** if for every k there is a $n \geq k$ such that $\delta^n(I) = I$. If Σ is fixed by the context, we drop it from the notation and call interpretations **δ-stable** and **δ-recurring**. If H is an interpretation, we consider the sequence $H^\infty := \{\delta^i(H)\,;\, i \in \mathbb{N}\}$ of interpretations occurring in the infinite iteration of δ on H. Clearly, if there is a stable interpretation in H^∞, then this is the only recurring interpretation in H^∞. We write $\mathrm{Rec}_{\Sigma,\delta}(H)$ for the set of recurring interpretations

[3] The relationship between pointer semantics and the dependency graph has been investigated in (Bol03).

in H^∞. Note that since the set of interpretations in finite, this set must be non-empty. If $I \in \mathrm{Rec}_{\Sigma,\delta}(H)$, then there are $i, j > 0$ such that $I = \delta^i(H) = \delta^{i+j}(H)$. Then for every $k < j$ and every n, we have $\delta^{i+k} = \delta^{i+n\cdot j+k}(H)$, so the sequence H^∞ exhibits a periodicity of length j (or a divisor of j). After the first occurrance of an $I \in \mathrm{Rec}_{\Sigma,\delta}(H)$, all further elements of H^∞ are recurring as well, and in particular, there is a recurring J such that $\delta(J) = I$. We shall call this an **I-predecessor** and will use this fact in our proofs.

Definition 2

$$[\![\Sigma, \mathrm{p}_i]\!]_{\delta,H} := \inf\{I(\mathrm{p}_i)\,;\, I \in \mathrm{Rec}_{\Sigma,\delta}(H)\}, \text{ and}$$
$$[\![\Sigma, \mathrm{p}_i]\!]_{\delta} := \inf\{I(\mathrm{p}_i)\,;\, \exists H(I \in \mathrm{Rec}_{\Sigma,\delta}(H))\}.$$

An algebra of pointer systems

In the language of abstract pointer systems, the possibility of complicated referential structures means that the individual proposition cannot be evaluated without its context.

As a consequence, the natural notion of logical operations is not that between propositions, but that between systems. If $\Sigma = \{\mathrm{p}_0 \leftarrow E_0, ..., \mathrm{p}_n \leftarrow E_n\}$ is a pointer system and $0 \le i \le n$, we define a pointer system that corresponds to the negation of p_i with one additional propositional variable $\mathrm{p}_\neg$,

$$\neg(\Sigma, \mathrm{p}_i) := \Sigma \cup \{\mathrm{p}_\neg \leftarrow \neg \mathrm{p}_i\}.$$

If we have two pointer systems

$$\Sigma_0 = \{\mathrm{p}_0 \leftarrow E_{0,0}, ..., \mathrm{p}_{n_0} \leftarrow E_{0,n_0}\}, \text{ and}$$
$$\Sigma_1 = \{\mathrm{p}_0 \leftarrow E_{1,0}, ..., \mathrm{p}_{n_1} \leftarrow E_{1,n_1}\},$$

we make their sets of propositions disjoint by considering a set $\{\mathrm{p}_0, ..., \mathrm{p}_{n_0}, \mathrm{p}_0^*, ..., \mathrm{p}_{n_1}^*, \mathrm{p}_*\}$ of $n_0 + n_1 + 2$ propositional variables. We then set

$$\Sigma_1^* := \{\mathrm{p}_0^* \leftarrow E_{1,0}, ..., \mathrm{p}_{n_1}^* \leftarrow E_{1,n_1}\}.$$

With this, we can now define two new pointer systems (with an additional propositional variable p_*):

$$(\Sigma_0, \mathrm{p}_i) \wedge (\Sigma_1, \mathrm{p}_j) := \Sigma_0 \cup \Sigma_1^* \cup \{\mathrm{p}_* \leftarrow \mathrm{p}_i \wedge \mathrm{p}_j^*\},$$
$$(\Sigma_0, \mathrm{p}_i) \vee (\Sigma_1, \mathrm{p}_j) := \Sigma_0 \cup \Sigma_1^* \cup \{\mathrm{p}_* \leftarrow \mathrm{p}_i \vee \mathrm{p}_j^*\}.$$

Logical properties of Gaifman pointer semantics

Fix a system $\Sigma = \{\mathrm{p}_0 \leftarrow E_0, ..., \mathrm{p}_n \leftarrow E_n\}$. A proposition p_i is called a **terminal node** if $E_i = \llcorner\!\lrcorner$; it is called a **source node** if p_i does not occur in any of the expressions $E_0, ..., E_n$. This corresponds directly to the properties of i in the dependency graph: p_i is a terminal node if and only if i has no outgoing edges in the dependency graph, and it is a source node if and only if i has no incoming edges in the dependency graph. The **Gaifman-Tarski operator** δ_B is defined as follows:

$$\delta_\mathrm{B}(I)(\mathrm{p}_i) := \begin{cases} I(E_i) & \text{if } \mathrm{p}_i \text{ is not terminal,} \\ I(\mathrm{p}_i) & \text{if } \mathrm{p}_i \text{ is terminal.} \end{cases}$$

Note that this operator can be described as follows:

> "From the clause $\mathrm{p}_i \leftarrow E_i$ form the equation $\mathtt{Q}_i$ by replacing the occurrences of p_i on the right-hand side of the equality sign with the values $I(\mathrm{p}_i)$. If p_i is a terminal node, let $\delta(I)(\mathrm{p}_i) := I(\mathrm{p}_i)$. Otherwise, let I^* be the unique solution to the system of equations $\{\mathtt{Q}_0, ..., \mathtt{Q}_n\}$ and let $\delta(I)(\mathrm{p}_i) := I^*(\mathrm{p}_i)$." (*)

This more complicated description will provide the motivation for the forward propagation operator δ_F in the section "Belief semantics with forward propagation".

The operator δ_B gives rise to a *logical* system, as the semantics defined by δ_B are compatible with the operations in the algebra of pointer systems.

Proposition 3 *Let $\Sigma = \{\mathrm{p}_0 \leftarrow E_0, ..., \mathrm{p}_n \leftarrow E_n\}$ be a pointer system. For any $i \le n$, we have*

$$[\![\neg(\Sigma, \mathrm{p}_i)]\!]_{\delta_\mathrm{B}} = \neg[\![\Sigma, \mathrm{p}_i]\!]_{\delta_\mathrm{B}}.$$

Proof. In this proof, we shall denote interpretations for the set $\{\mathrm{p}_0, ..., \mathrm{p}_n\}$ by capital letters I and J and interpretations for the bigger set $\{\mathrm{p}_0, ..., \mathrm{p}_n, \mathrm{p}_\neg\}$ by letters $\hat{I}$ and $\hat{J}$. It is enough to show that if $\hat{I}$ is δ_B-recurring, then there is some δ_B-recurring J such that $\hat{I}(\mathrm{p}_\neg) = \neg J(\mathrm{p}_i)$. If I is δ_B-recurring, we call J an **I-predecessor** if J is also δ_B-recurring and $\delta_\mathrm{B}(J) = I$, and similarly for $\hat{I}$. It is easy to see that every δ_B-recurring I (or $\hat{I}$) has an I-predecessor (or $\hat{I}$-predecessor) which is not necessarily unique.

As δ_B is a B-operator, we have that if $\hat{J}$ is δ_B-recurring, then so is $J := \hat{J}{\restriction}\{\mathrm{p}_0, ..., \mathrm{p}_n\}$.

Now let $\hat{I}$ be δ_B-recurring and let $\hat{J}$ be one of its $\hat{I}$-predecessors. Then by the above, $J := \hat{J}{\restriction}\{\mathrm{p}_0, ..., \mathrm{p}_n\}$ is δ_B-recurring and

$$\hat{I}(\mathrm{p}_\neg) = \delta_\mathrm{B}(\hat{J})(\mathrm{p}_\neg) = \neg\hat{J}(\mathrm{p}_i) = \neg J(\mathrm{p}_i).$$

q.e.d.

Proposition 4 *Let $\Sigma_0 = \{\mathrm{p}_0 \leftarrow E_0, ..., \mathrm{p}_n \leftarrow E_n\}$ and $\Sigma_1 = \{\mathrm{p}_0 \leftarrow F_0, ..., \mathrm{p}_m \leftarrow F_m\}$ be pointer systems. For any $i, j \le n$, we have*

$$[\![(\Sigma_0, \mathrm{p}_i) \vee (\Sigma_1, \mathrm{p}_j)]\!]_{\delta_\mathrm{B}} = [\![\Sigma_0, \mathrm{p}_i]\!]_{\delta_\mathrm{B}} \vee [\![\Sigma_1, \mathrm{p}_j]\!]_{\delta_\mathrm{B}}.$$

Similarly for $\vee$ replaced by $\wedge$.

Proof. The basic idea is very similar to the proof of Proposition 3, except that we have to be a bit more careful to see how the two systems Σ_0 and Σ_1 can interact in the bigger system. We reserve letters I_0 and J_0 for the interpretations on Σ_0, I_1 and J_1 for those on Σ_1 and I and J for interpretations on the whole system, including p_*. If $[\![\Sigma_0, \mathrm{p}_1]\!] = 1$, then any δ_B-recurring I must have $I(\mathrm{p}_*) = 1$ by the $\vee$-analogue of the argument given in the proof of Proposition 3. Similarly, for $[\![\Sigma_1, \mathrm{p}_j]\!] = 1$ and the case that $[\![\Sigma_0, \mathrm{p}_i]\!] = [\![\Sigma_1, \mathrm{p}_j]\!] = 0$. This takes care of six of the nine possible cases.

If I_0 and I_1 are δ_B-recurring, then so is the function $I := \delta(I_0) \cup \delta(I_1) \cup \{\langle \mathrm{p}_*, I_0(\mathrm{p}_i) \vee I_1(\mathrm{p}_j)\rangle\}$ (if I_0 is k-periodic and I_1 is ℓ-periodic, then I is at most $k \cdot \ell$-periodic). In particular, if we have such an I_0 with $I_0(\mathrm{p}_i) = ½$ and an I_1 with $I_1(\mathrm{p}_j) \neq 1$, then $I(\mathrm{p}_*) = ½$ (and symmetrically for interchanged rôles). Similarly, if we have recurring interpretations for relevant values 0 and 1 for both small systems, we can put them together to δ_B-recurring interpretations with values 0 and 1 for the big system. This gives the truth value ½ for the disjunction in the remaining three cases.

q.e.d.

H_0	$\delta_{\mathrm{B}}(H_0)$	$\delta_{\mathrm{F}}(H_0)$	H_1	$\delta_{\mathrm{B}}(H_1)$	$\delta_{\mathrm{F}}(H_1)$	H_2	$\delta_{\mathrm{B}}(H_2)$	$\delta_{\mathrm{F}}(H_2)$	H_3	$\delta_{\mathrm{B}}(H_3)$	$\delta_{\mathrm{F}}(H_3)$	H_4
0	½	0	0	1	0	½	1	½	1	1	1	1
1	1	½	1	1	½	1	1	½	1	1	½	1
0	0	½	0	0	½	0	0	½	0	0	½	0
½	0	½	0	0	½	0	0	½	0	0	0	0
½	½	½	½	1	½	1	1	1	1	1	1	1

Figure 1: The first three iterations of values of $H_0 = (0, 1, 0, ½, ½)$ up to the point of stability ($H_3 = (1, 1, 0, 0, 1)$).

Belief semantics with forward propagation

In (GLS07), the authors gave a revision operator that incorporated both backward and forward propagation. The value of $\delta(I)(\mathrm{p}_i)$ depended on the values of all $I(\mathrm{p}_j)$ such that j is connected to i in the dependency graph. Here, we split the operator in two parts: the backward part which is identical to the Gaifman-Tarski operator, and the forward part which we shall define now.

In analogy to the definition of δ_{B}, we define δ_{F} as follows. Given an interpretation I, we transform each clause $\mathrm{p}_i \leftarrow E_i$ of the system into an equation $\mathtt{Q}_i \equiv I(\mathrm{p}_i) = E_i$ where the occurrences of the p_i on the left-hand side of the equation are replaced by their I-values and the ones on the right-hand side are variables. We obtain a system $\{\mathtt{Q}_0, ..., \mathtt{Q}_n\}$ of $n+1$ equations in $\mathbb{T}$. Note that we cannot mimic the definition of δ_{B} directly: as opposed to the equations in that definition, the system $\{\mathtt{Q}_0, ..., \mathtt{Q}_n\}$ need not have a solution, and if it has one, it need not be unique. We therefore define: if p_i is a source node, then $\delta_{\mathrm{F}}(I)(\mathrm{p}_i) := I(\mathrm{p}_i)$. Otherwise, let S be the set of solutions to the system of equations $\{\mathtt{Q}_0, ..., \mathtt{Q}_n\}$ and let $\delta_{\mathrm{F}}(I)(\mathrm{p}_i) := \inf\{I(\mathrm{p}_i)\,;\, I \in S\}$ (remember that $\inf \varnothing = ½$). Note that this definition is literally the dual to definition (*) of δ_{B} (i.e., it is obtained from (*) by interchanging "right-hand side" by "left-hand side" and "terminal node" by "source node").

We now combine δ_{B} and δ_{F} to one operator δ_{T} by defining pointwise

$$\delta_{\mathrm{T}}(I)(\mathrm{p}_i) := \delta_{\mathrm{F}}(I)(\mathrm{p}_i) \otimes \delta_{\mathrm{B}}(I)(\mathrm{p}_i)$$

where $\otimes$ has the following truth table:[4]

$\otimes$	0	½	1
0	0	0	½
½	0	½	1
1	½	1	1

.

[4] The values for agreement ($0\otimes 0$, ½⊗½, and $1 \otimes 1$) are obvious choices. In case of complete disagreement ($0 \otimes 1$ and $1 \otimes 0$), you have little choice but give the value ½ of ignorance (otherwise, there would be a primacy of one direction over the other). For reasons of symmetry, this leaves two values ½⊗0 = 0⊗½ and ½⊗1 = 1 ⊗ ½ to be decided. We opted here for the most informative truth table that gives classical values the benefit of the doubt. The other options would be the tables

$\otimes_0$	0	½	1
0	0	½	½
½	½	½	1
1	½	1	1

$\otimes_1$	0	½	1
0	0	0	½
½	0	½	½
1	½	½	1

$\otimes_2$	0	½	1
0	0	½	½
f½	½	½	½
1	½	½	1

.

Each of these connectives will give rise to a slightly different semantics. We opted for the first connective $\otimes$, as the semantics based on the other three seem to have a tendency to stabilize on the value ½ very often (the safe option: in case of confusion, opt for ignorance).

Properties of our belief semantics

As mentioned in the introduction, we should not be shocked to hear that a system modelling belief and belief change does not follow basic logical rules such as Propositions 3 and 4. Let us take the particular example of conjunction: the fact that belief is not closed under the standard logical rules for conjunction is known as the *preface paradox* and has been described by Kyburg as "conjunctivitis" (Kyb70). In other contexts (that of the modality of "ensuring that"), we have a problem with simple binary conjunctions (Sch08). Of course, the failure of certain logical rules in reasoning about belief is closely connected to the so-called "errors in reasoning" observed in experimental psychology, e.g., the famous Wason selection task (Was68). What constitutes rational belief in this context is an interesting question for modellers and philosophers alike (Ste97; Chr07; Cou08). Let us focus on some concrete examples to validate our claim that the semantics we propose do agree with intuitive understanding, and thus serve as a quasi-empirical test for our system as a formalization of reasoning in self-referential situations with evidence.

Concrete examples

So far, we have just given an abstract system of belief flow in our pointer systems. In order to check whether our system results in intuitively plausible results, we have to check a few examples. Keep in mind that our goal should be to model human reasoning behaviour in the presence of partially paradoxical situations. In this paper, we can only give a first attempt at testing the adequacy of our system: an empirical test against natural language intuitions on a much larger scale is needed. For this, also cf. our section "Discussion and Future Work".

The Liar As usual, the liar sentence is interpreted by the system $\Sigma := \{\mathrm{p}_0 \leftarrow \neg \mathrm{p}_0\}$. Since we have only one propositional variable, interpretations are just elements of $\mathbb{T} = \{0, ½, 1\}$. It is easy to see that $\delta_{\mathrm{B}}(0) = \delta_{\mathrm{F}}(0) = \delta_{\mathrm{T}}(0) = 1$, $\delta_{\mathrm{B}}(½) = \delta_{\mathrm{F}}(½) = \delta_{\mathrm{T}}(½) = ½$, and $\delta_{\mathrm{B}}(1) = \delta_{\mathrm{F}}(1) = \delta_{\mathrm{T}}(1) = 0$. This means that the δ_{T}-behaviour of the liar sentence is equal to the Gaifman-semantics behaviour.

The Miller-Jones Example Consider the following test example from (GLS07):

> Professors Jones, Miller and Smith are colleagues in a computer science department. Jones and Miller dislike each other without reservation and are very liberal in telling everyone else that "everything that the other one says is false". Smith just returned from a trip abroad and needs to find out about two committee meetings on Monday morning. He sends out e-mails to his colleagues and to the department secretary. He asks all three of them about the meeting of the faculty, and

H_0^*	$\delta_\mathrm{B}(H_0^*)$	$\delta_\mathrm{F}(H_0^*)$	H_1^*	$\delta_\mathrm{B}(H_1^*)$	$\delta_\mathrm{F}(H_1^*)$	H_2^*	$\delta_\mathrm{B}(H_2^*)$	$\delta_\mathrm{F}(H_2^*)$	H_3^*	$\delta_\mathrm{B}(H_3^*)$	$\delta_\mathrm{F}(H_3^*)$	H_4^*	$\delta_\mathrm{B}(H_4^*)$	$\delta_\mathrm{F}(H_4^*)$	H_5^*
0	½	0	0	1	0	½	1	½	1	1	1	1	1	1	1
1	1	½	1	1	½	1	1	½	1	1	½	1	1	½	1
0	0	½	0	0	½	0	0	½	0	0	½	0	0	½	0
½	0	½	0	0	½	0	0	½	0	0	0	0	0	0	0
0	½	½	½	½	½	½	1	½	1	1	½	1	1	1	1
1	1	½	1	1	½	1	1	½	1	1	½	1	1	½	1
0	0	½	0	0	½	0	0	½	0	0	½	0	0	½	0
½	0	1	½	0	½	0	0	½	0	0	0	0	0	0	0
½	½	½	½	½	½	½	½	½	½	1	½	1	1	1	1

Figure 2: The first three iterations of values of $H_0^* = (0, 1, 0, \frac{1}{2}, 0, 1, 0, \frac{1}{2}, \frac{1}{2})$ up to the point of stability ($H_4^* = (1, 1, 0, 0, 1, 1, 0, 0, 1)$).

> Jones and the secretary about the meeting of the library committee (of which Miller is not a member).
>
> Jones replies: *"We have the faculty meeting at 10am and the library committee meeting at 11am; by the way, don't believe anything that Miller says, as he is always wrong."*
>
> Miller replies: *"The faculty meeting was cancelled; by the way don't believe anything that Jones says, as he is always wrong."*
>
> The secretary replies: *"The faculty meeting is at 10 am and the library committee meeting is at 11 am. But I am sure that Professor Miller told you already as he is always such an accurate person and quick in answering e-mails: everything Miller says is correct."* (GLS07, p. 408)

Trying to analyse Smith's reasoning process after he returns from his trip, we can assume that he generally believes the secretary's opinions, and that he has no prior idea about the truth value of the statements "the faculty meeting is at 10am" and "the library meeting is at 11am" and the utterances of Miller and Jones. We have a vague intuition that tells us that in this hypothetical situation, Smith should at least come to the conclusion that the library meeting will be held at 11am (as there is positive, but no negative evidence). His beliefs about the faculty meeting are less straightforward, as there is some positive evidence, but also some negative evidence, and there is the confusing fact that the secretary supports Miller's statement despite the disagreement in truth value.

In (GLS07, p. 409), the authors analysed this example with their real-valued model and ended up with a stable solution in which Smith accepted both appointments and took Jones's side (disbelieving Miller). In our system, we now get the following analysis: A pointer system formulation is given as follows.

$$\begin{array}{lllllll} \mathrm{p}_0 & \leftarrow & \neg\mathrm{p}_1 \wedge \neg\mathrm{p}_4, & \mathrm{p}_1 & \leftarrow & \neg\mathrm{p}_0 \wedge \mathrm{p}_2 \wedge \mathrm{p}_4, \\ \mathrm{p}_2 & \leftarrow & \textvisiblespace, & \mathrm{p}_3 & \leftarrow & \mathrm{p}_0 \wedge \mathrm{p}_2 \wedge \mathrm{p}_4, \\ \mathrm{p}_4 & \leftarrow & \textvisiblespace, & & & \end{array}$$

where p_0 is Miller's utterance, p_1 is Jones's utterance, p_2 is "the library meeting will take place at 11am", p_3 is the secretary's utterance, and p_4 is the "the faculty meeting will take place at 10am".

We identify our starting hypothesis with $H := (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 1, \frac{1}{2})$ (here, as usual, we identify an interpretation with its sequence of values in the order of the indices of the propositional letters). Then $\delta_\mathrm{B}(H) = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ and $\delta_\mathrm{F}(H) = (\frac{1}{2}, \frac{1}{2}, 1, 1, \frac{1}{2})$, so that we get $H' := \delta_\mathrm{T}(H) = (\frac{1}{2}, \frac{1}{2}, 1, 1, \frac{1}{2})$. Then, in the second iteration step, $\delta_\mathrm{B}(H') = (\frac{1}{2}, \frac{1}{2}, 1, \frac{1}{2}, \frac{1}{2})$ and $\delta_\mathrm{F}(H') = (\frac{1}{2}, \frac{1}{2}, 1, 1, \frac{1}{2})$, so we obtain stability at $\delta_\mathrm{T}(H') = H'$.

Examples of nonlogical behaviour

In what follows, we investigate some logical properties of the belief semantics, viz. negation and disjunction, focussing on stable hypotheses. To some extent, our results show that the operator δ_T is rather far from the logical properties of δ_B discussed in Propositions 3 and 4.

Negation Consider the pointer system Σ given by

$$\begin{array}{llllll} \mathrm{p}_0 & \leftarrow & \neg\mathrm{p}_3, & \mathrm{p}_1 & \leftarrow & \textvisiblespace, \\ \mathrm{p}_2 & \leftarrow & \textvisiblespace, & \mathrm{p}_3 & \leftarrow & \mathrm{p}_1 \wedge \mathrm{p}_2. \end{array}$$

The interpretation $H := (0, 1, 0, \frac{1}{2})$ is δ_T-stable, as $\delta_\mathrm{B}(H) = (\frac{1}{2}, 1, 0, 0)$, $\delta_\mathrm{F}(H) = (0, \frac{1}{2}, \frac{1}{2}, 1)$, and thus $\delta_\mathrm{T}(H) = H$.

Now let us consider the system $\neg(\Sigma, \mathrm{p}_3)$. Remember from the proof of Proposition 3 that stable interpretations for the small system could be extended to stable interpretations for the big system by plugging in the expected value for $\mathrm{p}_\neg$. So, in this particular case, the stable value for p_3 is ½, so we would expect that by extending H by $H_0(\mathrm{p}_\neg) := \frac{1}{2}$, we would get another stable interpretation.

But this is not the case, as the table of iterated values given in Figure 1 shows. Note that H_0 is not even recurring.

Disjunction Consider the pointer systems Σ and Σ^* and their disjunction $(\Sigma, \mathrm{p}_4) \vee (\Sigma^*, \mathrm{p}_1^*)$ given as follows:

$$\begin{array}{llllll} \mathrm{p}_0 & \leftarrow & \neg\mathrm{p}_3, & \mathrm{p}_0^* & \leftarrow & \neg\mathrm{p}_3^*, \\ \mathrm{p}_1 & \leftarrow & \textvisiblespace & \mathrm{p}_1^*, & \leftarrow & \textvisiblespace, \\ \mathrm{p}_2 & \leftarrow & \textvisiblespace & \mathrm{p}_2^*, & \leftarrow & \textvisiblespace, \\ \mathrm{p}_3 & \leftarrow & \mathrm{p}_1 \wedge \mathrm{p}_2, & \mathrm{p}_3^* & \leftarrow & \mathrm{p}_1^* \wedge \mathrm{p}_2^*, \\ \mathrm{p}_* & \leftarrow & \mathrm{p}_4 \vee \mathrm{p}_1^*. & & & \end{array}$$

Note that Σ and Σ^* are the same system up to isomorphism and that Σ is the system from the previous example. We already know that the interpretation $H = (0, 1, 0, \frac{1}{2})$ is δ_T-stable (therefore, it is δ_T-stable for both Σ and Σ^* in the appropriate reading.

The natural extension of H to the full system with nine propositional variables would be $H_0^* := (0, 1, 0, \frac{1}{2}, 0, 1, 0, \frac{1}{2}, \frac{1}{2})$, as p_* should take the value $H(\mathrm{p}_4) \vee H(\mathrm{p}_1^*) = \frac{1}{2} \vee 0 = \frac{1}{2}$. However, we see in Figure 2 that this interpretation is not stable (or even recurring).

Discussion and future work

Testing the behaviour of our system on the liar sentence and one additional example cannot be enough as an empirical test of the adequacy of our system. After testing more examples and having developed some theoretical insight into the system and its properties, we would consider testing the system experimentally by designing situations in which people reason about beliefs in self-referential situations with evidence, and then compare the predictions of our system to the actual behaviour of human agents.

Such an experimental test should not be done with just one system, but with a class of systems. We have already discussed that our choice of the connective $\otimes$ combining δ_{B} and δ_{F} to δ_{T} was not unique. Similarly, the rules for how to handle multiple solutions ("take the pointwise infimum") in the case of forward propagation are not the only way to deal with this formally. One natural alternative option would be to split the sequence H^{∞} into multiple sequences if there are multiple solutions. For instance, if we are trying to calculate $\delta_{\mathrm{F}}(H)$ and we have multiple solutions to the set of equations, then $\delta_{\mathrm{F}}(H)$ becomes a set of interpretations (possibly giving rise to different recurrences and stabilities, depending on which possibility you follow). There are many variants that could be defined, but the final arbiter for whether these systems are adequate descriptions of reasoning processes will have to be the experimental test.

References

Thomas Bolander. *Logical Theories for Agent Introspection*. PhD thesis, Technical University of Denmark, 2003.

David Christensen. *Putting Logic in its place. Formal Constraints on Rational Belief.* Oxford University Press, 2007.

Carlos Iván Chesñevar and Guillermo Ricardo Simari. A lattice-based approach to computing warranted beliefs in skeptical argumentation frameworks. 2007. In (Vel07, pp. 280–285).

Marian E. Counihan. *Looking for logic in all the wrong places: an investigation of language, literacy and logic in reasoning*. PhD thesis, Universiteit van Amsterdam, 2008. ILLC Publications DS-2008-10.

Haim Gaifman. Operational pointer semantics: Solution to self-referential puzzles I. In Moshe Y. Vardi, editor, *Proceedings of the 2nd Conference on Theoretical Aspects of Reasoning about Knowledge, Pacific Grove, CA, March 1988*, pages 43–59. Morgan Kaufmann, 1988.

Haim Gaifman. Pointers to truth. *Journal of Philosophy*, 89(5):223–261, 1992.

Anil Gupta and Nuel Belnap. *The revision theory of truth*. MIT Press, 1993.

Sujata Ghosh, Benedikt Löwe, and Erik Scorelle. Belief flow in assertion networks. In Uta Priss, Simon Polovina, and Richard Hill, editors, *Conceptual Structures: Knowledge Architectures for Smart Applications, 15th International Conference on Conceptual Structures, ICCS 2007, Sheffield, UK, July 22-27, 2007, Proceedings*, volume 4604 of *Lecture Notes in Computer Science*, pages 401–414. Springer, 2007.

Alan H. Goldman. A note on the conjunctivity of knowledge. *Analysis*, 36:5–9, 1975.

Geoff Hulten, David Maxwell Chickering, and David Heckerman. Learning Bayesian networks from dependency networks: A preliminary study. In Christopher M. Bishop and Brendan J. Frey, editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*. Society for Artificial Intelligence and Statistics, 2003.

Aaron Hunter and James P. Delgrande. Iterated belief change: A transition system approach. In Leslie Pack Kaelbling and Alessandro Saffiotti, editors, *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30-August 5, 2005*, pages 460–465. Professional Book Center, 2005.

Aaron Hunter and James P. Delgrande. An action description language for iterated belief change. 2007. In (Vel07, pp. 2498–2503).

Stanley Kok and Pedro Domingos. Learning the structure of markov logic networks. In Luc De Raedt and Stefan Wrobel, editors, *Proceedings of the 22nd International Machine Learning Conference*, pages 441–448. ACM Press, 2005.

Kristian Kerstin and Luc De Raedt. Bayesian logic programming: Theory and tool. In Lise Getoor and Ben Taskar, editors, *Introduction to Statistical Relational Learning*. MIT Press, 2007.

Henry Kyburg. Conjunctivitis. In Marshall Swain, editor, *Induction, Acceptance, and Rational Belief*, pages 55–82. Reidel, 1970.

Benedikt Löwe. Revision forever! In Henrik Schärfe, Pascal Hitzler, and Peter Øhrstrøm, editors, *Conceptual Structures: Inspiration and Application, 14th International Conference on Conceptual Structures, ICCS 2006, Aalborg, Denmark, July 16-21, 2006, Proceedings*, volume 4068 of *Lecture Notes in Computer Science*, pages 22–36. Springer, 2006.

Benjamin Schnieder. On what we can ensure. *Synthese*, 162:101–115, 2008.

Edward Stein. *Without good reason. The rationality debate in philosophy and cognitive science*. Clarendon Library of Logic and Philosophy. Clarendon Press, 1997.

Manuela M. Veloso, editor. *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*. AAAI Press, 2007.

Peter Wason. Reasoning about a rule. *Quarterly Journal of Experimental Psychology*, 20(3):273–281, 1968.