Paracomplete truth theory with *KFS*-definable determinateness

Marcos Cramer

International Center for Computational Logic, TU Dresden, Germany

Abstract. One way to deal with the Liar paradox is the paracomplete approach to theories of truth that gives up proofs by contradiction and the Law of the Excluded Middle. This allows one to reject both the Liar sentence and its negation. The simplest paracomplete theory of truth is *KFS* due to Saul Kripke. At face value, this theory suffers from the problem that it cannot say anything about the Liar paradox, so a defender of this theory cannot explain their rejection of the Liar sentence within the language of *KFS*. This was one of the motivations for Hartry Field to extend *KFS* with a conditional that is not definable within *KFS*. With the help of this conditional, Field defines a determinateness operator that can be used to explain one's rejection of the Liar sentence within the object language of his theory. Field's determinateness operator can be transfinitely iterated to create stronger notions of determinateness required to explain the rejection of paradoxical sentences involving the determinateness operator. In this paper, we show that Field's complex extension of *KFS* is not required in order to express rejection of paradoxical sentences like the Liar sentence. Instead one can work with a transfinite hierarchy of determinateness operators that are definable in *KFS*. This allows for Field's philosophically appealing treatment of the Liar sentence, the truth-teller and strengthenings of the Liar sentence to be reproducible within the theory *KFS*, which is semantically much simpler than Field's extension of *KFS* with a conditional.

1 Introduction

In everyday conversations, in scientific texts and in philosophical discussions we often make use of the predicate "true". So to get a better understanding of how we communicate our ideas and how we judge the correctness of our arguments, it is desirable to have a reasonable *theory of truth*, i.e. a logical formalism that contains the predicate *True*, that captures the inferences involving this predicate that we would intuitively deem acceptable, and that satisfies further rationality criteria like consistency.

In any formalism that contains the predicate *True* and captures come basic arithmetical reasoning, one can construct a *Liar sentence*, i.e. a sentence that asserts of itself that it is not true. If we apply some classically valid inferences combined with some intuitive inferences for the predicate *True* to such a Liar sentence, we can derive an inconsistency, and thus by some further classically valid inferences, we can derive every sentence of the language, rendering the formalism practically useless. This is called the *Liar paradox*. Any reasonable theory of truth needs to handle the Liar paradox in some way by putting some restrictions either on the intuitive inferences for the predicate *True* or on the rules of classical logic. Field [4] proposes a so-called *paracomplete* theory of truth that deals with the Liar paradox by restricting classical logic to the strong Kleene logic K_3, in which proofs by contradiction are not unrestrictedly admissible and the Law of the Excluded Middle ($\varphi \vee \neg\varphi$ for any formula φ) does not hold unrestrictedly, but which differs from intuitionistic logic by still admitting double negation elimination and De Morgan's Laws.

This paracomplete approach can be traced back to the work of Kripke [6]. The formal theory of truth that is based on the semantic construction due to Kripke is usually called *KFS* and has K_3 as its underlying logic.

The stance towards the Liar paradox that a paracomplete theory of truth defends is that of rejecting both the Liar paradox and its negation. It seems desirable that this stance should be expressible and justifiable within the formal theory that the paracompletist puts forward. At face value, it seems that this cannot be achieved within *KFS*. To overcome this limitation of *KFS*, Field [4] has introduced a determinateness operator, which allows one to say that the Liar paradox is not determinately true as a justification for rejecting the Liar sentence. The determinateness operator can be transfinitely iterated to create stronger notions of determinateness required to explain the rejection of paradoxical sentences involving the determinateness operator.

While Field's theory of truth is based on *KFS*, his determinateness operator is based on a conditional $\rightarrow$ that is not definable within *KFS* but needs to be added to the theory. The semantics of the language involving both *True* and $\rightarrow$ is defined through a complex combination of a revision-rule construction for the semantics of the conditional $\rightarrow$ and Kripke's construction for the semantics of *True*.

In this paper we show that Field's complex extension of *KFS* is not required in order to express reasons for rejecting paradoxical sentences like the Liar sentence. Instead one can work with a transfinite hierarchy of determinateness operators that are definable in *KFS*. The definition of these determinateness operators is inspired by the well-founded semantics of logic programming [10,2]. Additionally to explaining how to define this hierarchy of determinateness operators within *KFS*, we show how it can be used to give a philosophically appealing object-language explanation of the rejection of paradoxical sentences like the Liar sentence, the truth-teller and strengthenings of the Liar sentence, just like Field does with the transfinitely iterable determinateness operator defined within his complex extension of *KFS* with a conditional.

2 The Liar Paradox

A Liar sentence is a sentence that asserts of itself that it is not true. An informal example of a Liar sentence is the following sentence that uses the determiner *this* to refer to itself:

This sentence is not true.

Given that the semantics of the word *this* depends a lot on the communicative context, logicians often prefer to work with more formal Liar sentences whose interpretation is completely independent of the communicative context. For this purpose, one usually works in a formal language that extends the standard first-order language of arithmetic $\mathcal{L}^{\text{arithm}}$ with a truth predicate *True*, yielding the extended first-order language $\mathcal{L}^{\text{arithm}}_{True}$. It is well known that assuming some basic formal theory of arithmetic, e.g. Peano Arithmetic, one can define a Gödel numbering of any recursively presented countable formal language, i.e. an encoding of the syntax of that language in terms of natural numbers which maps each formula φ of the formal language to a unique natural number $\#\varphi$ that can be used to talk about φ in the formal language. Intuitively, the intended meaning of $True(n)$ is that there exists a sentence φ of $\mathcal{L}^{\text{arithm}}_{True}$ such that $\#\varphi = n$ and φ is true.

In the language of arithmetic we can denote any natural number n with a term: With *succ* denoting the successor function, $\bar{n}$ denotes the term $succ(\dots succ(0)\dots)$ with n occurrences of *succ*. For a formula φ, we write $\langle\varphi\rangle$ for the term $\overline{\#\varphi}$ that refers to the Gödel number of φ.

It is well known that using a diagonalization technique due to Carnap, Gödel and Tarski one can construct a sentence $L \in \mathcal{L}^{\text{arithm}}_{True}$ for which one can prove the following formula in Peano Arithmetic (and indeed in various weaker theories of arithmetic):[1]

$$L \equiv \neg True(\langle L\rangle) \tag{1}$$

Given our intuitive interpretation of *True*, the sentence L is thus provably equivalent to the statement that L is not true. In other words, L is a Liar sentence, and unlike the informal Liar sentence presented above, it is a purely formal Liar sentence that does not depend on the interpretation of a context-dependent word like *this*.

Once we have constructed L, it seems like we can derive both $\neg L$ and L using standard rules of inference. We start with a proof by contradiction that establishes $\neg L$: Assume for a contradiction that L holds. In that case L is true, i.e. $True(\langle L\rangle)$ holds. But from (1) we get $True(\langle L\rangle) \supset \neg L$, so by modus ponens we get $\neg L$. This contradicts our assumption that L holds. This completes the proof by contradiction, i.e. we can retract the assumption and deduce $\neg L$. But from (1) we have $\neg L \supset True(\langle L\rangle)$, so by modus ponens we get $True(\langle L\rangle)$, i.e. we get that L is true. So we have deduced both $\neg L$ and L, a contradiction.

[1] As is usual in the literature on theories of truth that involve a conditional that is distinct from material implication, we use the symbol $\supset$ for material implication and the symbol $\equiv$ for material bi-implication, so that the symbol $\rightarrow$ is reserved for the conditional that is distinct from material implication, and the symbol $\leftrightarrow$ is reserved for the bidirectional application of this conditional.

This is what is commonly called the *Liar paradox*. If we try to formalize this apparent proof in the proof calculus of natural deduction, we see that apart from the explicitly stated rules *modus ponens* (also called ($\supset$-Elim)) and *proof by contradiction* (also called ($\neg$-Introd)), we implicitly made use of two further rules of inference that involve the predicate *True*:

$$\text{(T-Introd)} \qquad \varphi \vdash \mathit{True}(\langle\varphi\rangle)$$
$$\text{(T-Elim)} \qquad \mathit{True}(\langle\varphi\rangle) \vdash \varphi$$

These two rules are very compelling, because they seem to precisely characterize our intuitions about the meaning of the predicate *True*. What the Liar paradox shows is that these rules cannot be consistently combined with classical logic. Multiple avenues have been explored to deal with this:

- Those who want to keep classical logic fully in place need to reject at least one of (T-Introd) and (T-Elim). The most well-known theory of truth that works with classical logic is the so-called Kripke-Feferman theory KF, which accepts (T-Elim) but rejects (T-Introd) [3]. However, this theory has the awkward property of declaring "L, but L is not true".
- One can bite the bullet and accept that both L and $\neg L$ can be derived, but restrict classical logic so that this inconsistency does not lead to explosion, i.e. to the derivability of all sentences. This approach is called the *paraconsistent* approach and was first proposed by Priest [7].
- One can restrict the structural rules of inference that were left implicit in the above piece of informal reasoning and that allow one to use already derived sentences as premises for further derivations, as well as to use an assumption more than once in a derivation [8].
- One can give up one of those rules of inference that were explicitly used in the above elicitation of the Liar paradox. The most common rule of inference to be dropped is proof by contradiction ($\neg$-Introd). Dropping this rule is called the *paracomplete* approach, and it has recently gained traction due to Field's [4] defense of it.

In this paper we are working with a paracomplete approach to the Liar paradox, i.e. we are giving up proof by contradiction in its unrestricted form, which allows us to accept (T-Introd) and (T-Elim) unrestrictedly.

3 From Kripke to Field: Paracomplete Approaches to Semantic Paradoxes

Kripke [6] defined a construction that can be used to give a three-valued model-theoretic semantics for the language $\mathcal{L}^{\text{arithm}}_{\mathit{True}}$. This construction gives rise to the paracomplete theory *KFS* and is also at the heart of the paracomplete theory of truth presented by Field [4]. The same construction can also be used as the basis for approaches based on classical logic, e.g. for the Kripke-Feferman theory KF.

We will now sketch this construction and explain how it serves as a basis for a paracomplete theory of truth.

Following Field [4], we use $\{0, \frac{1}{2}, 1\}$ as the names for the three truth-values, in order to avoid confusion between the object language predicate *True* and the truth-value 1 that was called *true* by Kripke [6]. We assume that $\mathcal{L}^{\text{arithm}}$ contains the falsity constant $\bot$, the negation symbol $\neg$, the conjunction symbol $\wedge$ and the universal quantifier $\forall$. We write $(\varphi \vee \psi)$ for $\neg(\neg\varphi \wedge \neg\psi)$, $(\varphi \supset \psi)$ for $\neg(\varphi \wedge \neg\psi)$, and $\exists x\, \varphi$ for $\neg\forall x\, \neg\varphi$. We sometimes drop brackets when this does not cause confusion. As usual, we assume that $\mathcal{L}^{\text{arithm}}$ contains the equality symbol $=$ as its only predicate symbol and that it contains a constant symbol 0, a unary function symbol *succ* (*successor*) and two binary function symbols $+$ and $\cdot$, conventionally written in infix notation (e.g. $(s(0) \cdot (s(0) + s(0)))$). A countably infinite supply of variable symbols $(x, y, z, x_0, x_1, \dots)$ is assumed to be given. As usual, the constant symbol 0 and the variable symbols can be combined with the function symbols to form *terms*.

A *variable assignment* s is a function that assigns a natural number to each variable. Given a variable assignment s, a variable x and a natural number n, $s[x:n]$ denotes the variable assignment that coincides with s on all variables other than x and that assigns n to x. One can inductively define the interpretation t^s of a term t under a variable assignment s as follows:

$$\begin{aligned}
0^s &= \text{the natural number } 0 \\
x^s &= \text{the number that } s \text{ assigns to the variable } x \\
\mathit{succ}(t)^s &= \text{the successor of the natural number } t^s \\
(t_1 + t_2)^s &= \text{the sum of the natural number } t_1^s \text{ and the natural number } t_2^s \\
(t_1 \cdot t_2)^s &= \text{the product of the natural number } t_1^s \text{ and the natural number } t_2^s
\end{aligned}$$

Kripke's construction is based on a transfinite recursion which starts with assigning the truth-value $\frac{1}{2}$ to each formula and then recursively updates the truth-values of all formulas until a fixed point is reached after some transfinite number of iterations. At each step α of this transfinite recursion and for each variable assignment s, we assign to each formula $\varphi \in \mathcal{L}^{\text{arithm}}_{\mathit{True}}$ a truth-value $\varphi^{\alpha,s} \in \{0, \frac{1}{2}, 1\}$. The transfinite recursion is defined as follows:

$$\begin{aligned}
\varphi^{0,s} &= \tfrac{1}{2} \text{ for every } \varphi \in \mathcal{L}^{\text{arithm}}_{\mathit{True}} \text{ and every variable assignment } s \\
(t_1 = t_2)^{\alpha+1,s} &= \begin{cases} 1 \text{ if } t_1^s = t_2^s \\ 0 \text{ otherwise} \end{cases} \\
(\bot)^{\alpha+1,s} &= 0 \\
(\neg\varphi)^{\alpha+1,s} &= 1 - \varphi^{\alpha+1,s} \\
(\varphi \wedge \psi)^{\alpha+1,s} &= \min(\varphi^{\alpha+1,s}, \psi^{\alpha+1,s}) \\
(\forall x\, \varphi)^{\alpha+1,s} &= \min\{\varphi^{\alpha+1,s[x:n]} \mid n \in \mathbb{N}\} \\
(\mathit{True}(t))^{\alpha+1,s} &= \begin{cases} 1 & \text{if there is a sentence } \varphi \in \mathcal{L}^{\text{arithm}}_{\mathit{True}} \text{ with } t^s = \langle\varphi\rangle \text{ and } \varphi^{\alpha,s} = 1 \\ \tfrac{1}{2} & \text{if there is a sentence } \varphi \in \mathcal{L}^{\text{arithm}}_{\mathit{True}} \text{ with } t^s = \langle\varphi\rangle \text{ and } \varphi^{\alpha,s} = \tfrac{1}{2} \\ 0 & \text{otherwise} \end{cases}
\end{aligned}$$

$$\varphi^{\lambda,s} = \begin{cases} \nicefrac{1}{2} & \text{if } \lambda \text{ is a limit ordinal and } \varphi^{\alpha,s} = \nicefrac{1}{2} \text{ for all } \alpha < \lambda \\ 1 & \text{if } \lambda \text{ is a limit ordinal and } \varphi^{\alpha,s} = 1 \text{ for some } \alpha < \lambda \\ 0 & \text{otherwise, if } \lambda \text{ is a limit ordinal.} \end{cases}$$

Clearly for sentences (formulas without free variables) the variable assignment has no impact on the assigned truth-value, so we can write φ^{α} instead of $\varphi^{\alpha,s}$ when φ is a sentence.

One can easily see that this transfinite recursion is monotonic, i.e. if $\varphi^{\alpha,s} \neq \nicefrac{1}{2}$ for some ordinal α, then $\varphi^{\beta,s} = \varphi^{\alpha,s}$ for all $\beta \geq \alpha$. This together with the fact that $\mathcal{L}^{\text{arithm}}_{True}$ is countable implies that a fixed point is reached at some countable ordinal α_0, i.e. for each formula $\varphi \in \mathcal{L}^{\text{arithm}}_{True}$, each ordinal $\alpha \geq \alpha_0$ and each variable assignment s, $\varphi^{\alpha,s} = \varphi^{\alpha_0,s}$. The (ultimate) truth-value of a sentence $\varphi \in \mathcal{L}^{\text{arithm}}_{True}$, denoted as $|\varphi|$, is defined to be φ^{α_0}.

The idea behind paracomplete theories of truth like that of Field [4] is that the only sentences of $\mathcal{L}^{\text{arithm}}_{True}$ that we should accept are the ones that get assigned truth-value 1 in Kripke's construction, while sentences with truth-value 0 or ½ should be rejected. The theory comprising all the sentences with truth-value 1 in Kripke's construction is usually called *KFS*.

A sentence φ for which $(\varphi \vee \neg\varphi)$ is accepted is called *bivalent*. All sentences that do not involve the predicate *True* are bivalent in *KFS*. Also any sentence in which *True* is only applied to Gödel codes of sentences not involving *True* is bivalent. This process can be continued to the point that can be informally characterized by saying that all sentences in which there are no infinite nestings of the predicate *True* are bivalent. Note that "n is the Gödel code of a bivalent formula" is itself not generally bivalent, so when we want to bivalently restrict ourselves to bivalent formulas, we need to use a syntactic criterion like "the formula does not contain the predicate *True*" (but this way we always miss out some bivalent formulas).

Usually when we reject a certain statement, we can explain this rejection by explaining that we believe the negation of the statement in question, and maybe additionally give reasons for this belief in the statement's negation. For a defender of a paracomplete theory of truth, this kind of explanation for their rejection of L is not possible, because they also reject $\neg L$. So how could a paracompletist explain their rejection of L?

One thing that they could do is to step outside the object language $\mathcal{L}^{\text{arithm}}_{True}$ and use the metalinguistic vocabulary of Kripke's construction to explain that L does not get truth-value 1 in that construction. But this solution is unsatisfying, because it relies on going to a metalanguage rather than staying within a given language. This immediately raises the question why we don't immediately start with a language (e.g. the language of set theory) in which Kripke's construction can be performed. Actually Field [4] does start with the language of set theory, but in that case the set-class distinction implies that Kripke's construction cannot be performed with $\forall$ interpreted as unrestrictedly quantifying over all sets, but can only be performed with $\forall$ interpreted as quantifying over the members of a fixed set U. And no matter what set U we choose, we always get false sentences that have value 1 with respect to quantification over U. In that case Kripke's construction is not a trustworthy criterion for truth, so reference to it as an explication for one's rejection of a certain sentence is not convincing.

Instead of proposing such a metalinguistic response to the question of how to explain one's rejection of L, Field [4] introduces a determinateness operator D, where $D\varphi$ intuitively means 'determinately φ'. With the help of this operator, Field can say $\neg DL$, i.e. say that the Liar sentence L is not determinately true; this is taken to be a reason for rejecting L that is expressible in the object language. In Field's account, D is not a primitive notion, but is defined in terms of a non-material conditional $\rightarrow$ that Field introduces: $D\varphi$ is taken to mean $\varphi \wedge \neg(\varphi \rightarrow \neg\varphi)$ (or equivalently $\varphi \wedge (\top \rightarrow \varphi)$). The semantics of $\rightarrow$ is explicated through a transfinite revision-rule construction. This is a construction that has some resemblance to Kripke's construction, but it does not have the monotonicity property of that construction. Instead, truth-values of sentences involving $\rightarrow$ can oscillate between different truth-values. If such an oscillation occurs all the way towards a limit ordinal λ, then the truth-value at step λ will be ½.

Once the determinateness operator is introduced, one can form a strengthened Liar sentence L_1 that is provably equivalent to $\neg D\, True(\langle L_1\rangle)$. This brings up the question of what the status of this strengthened Liar sentence is. It turns out that accepting it or its negation would be problematic, but we cannot express rejection of it in the same way as in the case of the Liar sentence, because accepting $\neg DL_1$ would amount to accepting $\neg D\, True(\langle L_1\rangle)$, i.e. to accepting L_1. What we can do instead is to explain our rejection of L_1 by claiming $\neg DDL_1$, also written as $\neg D^2 L_1$. So iterating the determinateness operator yields a stronger notion of determinateness, and this stronger notion can be used to explain our stance towards a sentence involving a weaker notion of determinateness.

But then we can construct a sentence L_2 that is provably equivalent to $\neg D^2\, True(\langle L_2\rangle)$, and to explain our rejection of L_2 we need an even stronger notion of determinateness, namely D^3. Field shows that this process of iterating D can even be continued into the transfinite, but this involves some technical difficulties that go beyond the scope of this paper. It turns out that the question of how far precisely this can be meaningfully continued into the transfinite is also a very tricky one, as it touches on König's paradox of the least undefinable ordinal [9]. Field observes that for any given hereditarily definable ordinal α (i.e. for any α such that α and all its predecessors are definable), D^α is a definable and well-behaved operator. However, one cannot assume that "α is a hereditarily definable ordinal" is bivalent, because that would lead to a contradiction by König's paradox.

4 A Definable Transfinite Hierarchy of Determinateness Operators

In this section we show how a transfinite hierarchy of determinateness operators can be defined within *KFS*. Before giving the formal definition, let us first start with an informal motivation. We want to be able to say of some formulas that they have a determinate truth-value, namely being determinately true or determinately false, while saying of other formulas that they do not have a determinate truth-value. Additionally, we want this notion of determinateness to have a sensible compositional behavior with respect to the logical connectives and quantifiers, namely:

1. When t_1 and t_2 are variable-free terms that denote the same natural number, then $t_1 = t_2$ is determinately true.
2. When t_1 and t_2 are variable-free terms that denote different natural numbers, then $t_1 = t_2$ is determinately false.
3. $\bot$ is determinately false.
4. When φ is determinately true, $\neg\varphi$ is determinately false.
5. When φ is determinately false, $\neg\varphi$ is determinately true.
6. When φ is determinately false, $\varphi \wedge \psi$ is determinately false.
7. When ψ is determinately false, $\varphi \wedge \psi$ is determinately false.
8. When φ and ψ are both determinately true, $\varphi \wedge \psi$ is determinately true.
9. When $\varphi(t)$ is determinately false, $\forall x\, \varphi(x)$ is determinately false.
10. When $\varphi(\bar{n})$ is determinately true for all $n \in \mathbb{N}$, then $\forall x\, \varphi(x)$ is determinately true.
11. When n is the Gödel code of a determinately true sentence, then $\mathit{True}(n)$ is determinately true.
12. When n is the Gödel code of a determinately false sentence, then $\mathit{True}(n)$ is determinately false.

One can interpret the above criteria for "determinately true" and "determinately false" as an implicit inductive definition of these two notions. If one steps out of the object language of *KFS* and allows a metatheoretic definition, one can transform the inductive definition into an explicit definition: For this, one needs to choose for the extensions of "determinately true" and "determinately false" a pair of sets that satisfies the above criteria such that the sets are minimal with respect to set inclusion among the sets with this property. We cannot quantify over sets of formulas within *KFS*, so we cannot use this strategy to turn the implicit inductive definition into an explicit definition within *KFS*.

Denecker and Vennekens [2] show that various kinds of inductive definitions used in mathematics can be given a unified semantic account with the help of the well-founded semantics of logic programs. In the following we take inspiration from the formal definition of the well-founded semantics to transform the above implicit inductive definition of determinate truth and determinate falsity into an explicit definition of a hierarchy of determinateness operators Δ_α within *KFS*. Formulas of the form $\Delta_\alpha\varphi$ are not in general bivalent, but whenever they are bivalent, the truth-value of $\Delta_\alpha\varphi$ is identical to the one that gets assigned to the statement "φ is definitely true" in the metatheoretic explicit definition of "determinately true".

In order to explain how our definition is inspired by the well-founded semantics, we give a brief informal sketch of the definition of the well-founded semantics; readers interested in the formal details may consult the paper by Denecker and Vennekens [2].

An *inductive definition* is a set of clauses consisting of a definiendum called *head* and a definiens called *body*. The head is always an atomic formula, and all the predicates that appear in at least one head are considered to be simultaneously defined by this inductive definition. As an example, the above enumerated list can be read as a simultaneous inductive definition of "determinately true" and "determinately false", where each item represents a clause and in each clause, the part between "When" and the comma is the body and the part after the comma is the head (as the clause about $\perp$ shows, the head can also be empty, in which case it is considered to be always true). The well-founded model of an inductive definition can be defined as the limit of a *well-founded induction*, which is a transfinite sequence of approximations to the well-founded model. In each approximation, some atoms involving one of the defined predicates are already known to be true, other such atoms are already known to be false, and others still have unknown truth-value. At each successor step, we refine the previous approximation in one of two possible ways: If our current approximation makes the body of some clause true, we may add the head of that clause to the atoms that have been accepted to be true. And if adding some atoms to the set of atoms considered false results in all bodies that define those atoms to be false, then we may indeed add those atoms to the set of atoms considered false. We continue this process until no more refinement is possible, at which point the well-founded model of the inductive definition has been reached.

Note the asymmetry between making atoms true and making atoms false: We are free to assume that atoms are false, as long as this prophecy turns out to fulfill itself, whereas for considering something true at some step, we must have reasons for considering it true already at a previous step.

Inspired by the treatment of falsity in the definition of the well-founded semantics, we want to be able to say that a formula does not have a determinate truth-value if assuming it to not have a determinate truth-value turns out to be a self-fulfilling prophecy. For example, if we assume the Liar sentence L and the sentence $\mathit{True}(L)$ not to have a determinate truth-value, then for $n = \langle L \rangle$ the bodies of the clauses 11 and 12 in the above enumerated list are false, which confirms the indeterminateness of $\mathit{True}(\langle L \rangle)$, which in turn by clause 4 confirms the indeterminateness of $\neg \mathit{True}(\langle L \rangle)$, i.e. of L.

This motivates the definition of the functions *ConfNotDetTrue* and *ConfNotDetFalse*. The function *ConfNotDetTrue* maps a triple $(\varphi, \psi_0(x), \psi_1(x))$ consisting of a sentence φ and two formulas $\psi_0(x)$ and $\psi_1(x)$ to a sentence $\chi = \mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))$, where the intuition is that when χ is satisfied, then the assumption that all formulas satisfying ψ_0 are not determinately true and all formulas satisfying ψ_1 are not determinately false confirms that φ is not determinately true. Similarly the function *ConfNotDetFalse* maps a triple $(\varphi, \psi_0(x), \psi_1(x))$ consisting of a sentence φ and two formulas $\psi_0(x)$ and $\psi_1(x)$ to a sentence $\chi = \mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))$, where the intuition is that when χ is satisfied, then the assumption that all formulas satisfying ψ_0 are not determinately true and all formulas satisfying ψ_0 are not determinately false confirms that φ is not determinately false.

$$\begin{array}{lcl}
\mathit{ConfNotDetTrue}(t_1 = t_2, \psi_0(x), \psi_1(x)) & = & \neg t_1 = t_2 \\
\mathit{ConfNotDetTrue}(\bot, \psi_0(x), \psi_1(x)) & = & \neg\bot \\
\mathit{ConfNotDetTrue}(\neg\varphi, \psi_0(x), \psi_1(x)) & = & \psi_1(\langle\varphi\rangle) \\
\mathit{ConfNotDetTrue}(\varphi_1 \wedge \varphi_2, \psi_0(x), \psi_1(x)) & = & \psi_0(\langle\varphi_1\rangle) \vee \psi_0(\langle\varphi_2\rangle) \\
\mathit{ConfNotDetTrue}(\forall x\, \varphi(x), \psi_0(x), \psi_1(x)) & = & \exists n\, \psi_0(\langle\varphi(\bar{n})\rangle) \\
\mathit{ConfNotDetTrue}(\mathit{True}(t), \psi_0(x), \psi_1(x)) & = & \neg \mathit{sentence}(t) \vee \psi_0(t)
\end{array}$$

$$\begin{array}{lcl}
\mathit{ConfNotDetFalse}(t_1 = t_2, \psi_0(x), \psi_1(x)) & = & t_1 = t_2 \\
\mathit{ConfNotDetFalse}(\bot, \psi_0(x), \psi_1(x)) & = & \bot \\
\mathit{ConfNotDetFalse}(\neg\varphi, \psi_0(x), \psi_1(x)) & = & \psi_0(\langle\varphi\rangle) \\
\mathit{ConfNotDetFalse}(\varphi_1 \wedge \varphi_2, \psi_0(x), \psi_1(x)) & = & \psi_1(\langle\varphi_1\rangle) \wedge \psi_1(\langle\varphi_2\rangle) \\
\mathit{ConfNotDetFalse}(\forall x\, \varphi(x), \psi_0(x), \psi_1(x)) & = & \forall n\, \psi_1(\langle\varphi(\bar{n})\rangle) \\
\mathit{ConfNotDetFalse}(\mathit{True}(t), \psi_0(x), \psi_1(x)) & = & \mathit{sentence}(t) \wedge \psi_1(t)
\end{array}$$

Here $\mathit{sentence}(t)$ means that t denotes the Gödel code of a sentence, i.e. of a formula without free variables.

An important observation is that $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))$ and $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))$ are definable in *KFS*. In other words, $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \chi$ is merely a shorthand notation for a certain formula of $\mathcal{L}^{\text{arithm}}_{\mathit{True}}$ with four variables k, l, m and n, which refer to the Gödel codes of φ, $\psi_0(x)$, $\psi_1(x)$ and χ respectively, and similarly for *ConfNotDetFalse*. So the usage of these predicates does not constitute an expansion of the language $\mathcal{L}^{\text{arithm}}_{\mathit{True}}$.

Note that there is a close conceptual connection between these definitions of *ConfNotDetTrue* and *ConfNotDetFalse* and the informal inductive definition of “determinately true” and “determinately false” presented in the 12-item list at the beginning of this section. Consider for example item 5 from that list, according to which $\neg\varphi$ is determinately true when φ is determinately false. Since that 12-item list is to be read as an inductive definition and this item is the only one that gives a criterion for a formula of the form $\neg\varphi$ being determinately true, the assumption that φ is not determinately false allows us to confirm that $\neg\varphi$ is not determinately true. This is captured by the third line of the definition of *ConfNotDetTrue*: $\mathit{ConfNotDetTrue}(\neg\varphi, \psi_0(x), \psi_1(x))$ is the formula $\psi_1(\langle\varphi\rangle)$. So if $\mathit{ConfNotDetTrue}(\neg\varphi, \psi_0(x), \psi_1(x))$ is true and we additionally assume that all formulas satisfying ψ_1 are not determinately false, we are in effect assuming that φ is not determinately false, from which it follows that $\neg\varphi$ is not determinately true.

We want to use the functions *ConfNotDetTrue* and *ConfNotDetFalse* to single out the formulas that have truth-value ½. In light of paradox, this goal cannot be attained without restrictions. What we do instead is to approximate the set of formulas with truth-value ½ through a predicate $Ind(\alpha, n)$, which implies that n is the Gödel code of a formula with indeterminate truth-value, i.e. with truth-value ½. Here α is an ordinal notation that allows us to approximate the indended meaning of "indeterminate truth-value" ever more: The higher the value of α, the more indeterminate sentences are classified as indeterminate according to the criterion $Ind(\alpha, \langle\varphi\rangle)$.

In the definition of the predicate $Ind(\alpha, n)$, we will need to bivalently restrict ourselves to bivalent formulas. As explained before, this could be done by restricting ourselves to formulas that do not contain *True*. However, as we increase the value of α, we want to increase the number of bivalent formulas that we restrict ourselves to. This can be achieved by defining a notion of depth of a formula. A formula not containing *True* has depth 0, and a formula in which *True* is only applied to formulas of depth 0 has depth 1. This notion of depth can be extended into the transfinite as follows:

- $depth(0, n)$ denotes the *KFS* formalization of the statement "n if the Gödel code of a formula in $\mathcal{L}^{\text{arithm}}$".
- For every ordinal notation α, $depth(\alpha{+}1, n)$ denotes the *KFS* formalization of the statement "n is the Gödel code of a formula φ such that every subformula of φ of the form $True(t)$ only occurs within a subformula of φ of the form $depth(\alpha, t) \supset \psi[t]$".
- For every ordinal notation λ that denotes a limit ordinal, $depth(\lambda, n)$ denotes the *KFS* formalization of the statement "n is the Gödel code of a formula that satisfies $depth(\alpha, n)$ for some $\alpha < \lambda$".

Now we are ready to define the predicate *Ind* based on the functions *ConfNotDetTrue* and *ConfNotDetFalse* and the notion of the depth of a formula. Recall that the intuitive meaning of $Ind(\alpha, \langle\chi\rangle)$ is that χ has indeterminate truth-value. We define $Ind(\alpha, \langle\chi\rangle)$ to be the *KFS* formalization of the statement "There exist two formulas $\psi_0(x)$ and $\psi_1(x)$ of depth α such that $\psi_0(\langle\chi\rangle)$ and $\psi_1(\langle\chi\rangle)$ are true and for any formula φ, if $\psi_0(\langle\varphi\rangle)$ is true, then $ConfNotDetTrue(\varphi, \psi_0(x), \psi_1(x))$ is true, and if $\psi_1(\langle\varphi\rangle)$ is true then $ConfNotDetFalse(\varphi, \psi_0(x), \psi_1(x))$ is true."

Intuitively, this definition says that there are some formulas assumed to not be determinately true (namely those whose Gödel codes satisfy $\psi_0(x)$) and some further formulas assumed not to be detereminately false (namely those whose Gödel codes satisfy $\psi_1(x)$) such that χ is in both of these collections of formulas and that assuming these formulas to not be determinately true/false confirms that they are not determinately true/false. For choosing these two collections, we make use of the predicates $\psi_0(x)$ and $\psi_1(x)$ of depth α. The fact that they have depth α ensures that they are bivalent. If we allowed for arbitrary (possibly non-bivalent) formulas at this place, we would never be able to accept $\neg Ind(\alpha, \chi)$ for any χ, because a non-bivalent $\psi_0(x)$ or a non-bivalent $\psi_1(x)$ would ensure that the truth-value of $Ind(\alpha, \chi)$ in Kripke's construction is at most ½.

The following lemma formalizes the idea that $Ind(\alpha, \langle\varphi\rangle)$ expresses the indeterminateness of φ:

Lemma 1. *Suppose φ is a sentence such that $|Ind(\alpha, \langle\varphi\rangle)| = 1$. Then $|\varphi| = \frac{1}{2}$.*

Proof. The definition of *Ind* and the fact that $|Ind(\alpha, \langle\varphi\rangle)| = 1$ together imply that there exist formulas $\psi_0(x)$ and $\psi_1(x)$ of depth α such that the following properties are satisfied:

1. $|\psi_0(\langle\varphi\rangle)| = 1$.
2. $|\psi_1(\langle\varphi\rangle)| = 1$.
3. For any $\chi \in \mathcal{L}^{\text{arithm}}_{True}$, if $|\psi_0(\langle\chi\rangle)| = 1$ then $|ConfNotDetTrue(\chi, \psi_0(x), \psi_1(x))| = 1$.
4. For any $\chi \in \mathcal{L}^{\text{arithm}}_{True}$, if $|\psi_1(\langle\chi\rangle)| = 1$ then $|ConfNotDetFalse(\chi, \psi_0(x), \psi_1(x))| = 1$.

Let Γ_0 be the set of all sentences $\chi \in \mathcal{L}^{\text{arithm}}_{True}$ such that $|\psi_0(\langle\chi\rangle)| = 1$. Let Γ_1 be the set of all sentences $\chi \in \mathcal{L}^{\text{arithm}}_{True}$ such that $|\psi_1(\langle\chi\rangle)| = 1$.

We will now prove that for every ordinal β, we have the following two properties:

- For every $\chi \in \Gamma_0$, $\chi^\beta \in \{0, \frac{1}{2}\}$.
- For every $\chi \in \Gamma_1$, $\chi^\beta \in \{1, \frac{1}{2}\}$.

Note that these two properties together imply that for every ordinal β, $\varphi^\beta = \frac{1}{2}$, as required.

We prove these two properties by a simultaneous induction over β and the complexity of χ. So as an inductive hypothesis we assume that for every ordinal β' and every formula χ' such that either $\beta' < \beta$ or $\beta' = \beta$ and χ' is a subformula of χ, we have that if $\chi' \in \Gamma_0$, then $\chi'^{\beta'} \in \{0, \frac{1}{2}\}$, and we have that if $\chi' \in \Gamma_1$, then $\chi'^{\beta'} \in \{1, \frac{1}{2}\}$.

We now make the following case distinction:

Case 1: χ is of the form $t_1 = t_2$.
Assume $\chi \in \Gamma_0$, i.e. $|ConfNotDetTrue(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $ConfNotDetTrue(\chi, \psi_0(x), \psi_1(x))$ is $\neg t_1 = t_2$, so $\chi^\beta = |t_1 = t_2| = 0 \in \{0, \frac{1}{2}\}$, as required.

Now assume $\chi \in \Gamma_1$, i.e. $|ConfNotDetFalse(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $ConfNotDetFalse(\chi, \psi_0(x), \psi_1(x))$ is $t_1 = t_2$, so $\chi^\beta = |t_1 = t_2| = 1 \in \{1, \frac{1}{2}\}$, as required.

Case 2: χ is $\bot$.
Assume $\chi \in \Gamma_0$. Since $\chi = \bot$, $\chi^\beta = 0 \in \{0, \frac{1}{2}\}$, as required.

Note that $ConfNotDetFalse(\chi, \psi_0(x), \psi_1(x))$ is $\bot$, so we necessarily have that $|ConfNotDetFalse(\chi, \psi_0(x), \psi_1(x))| = 0 \neq 1$. Thus the case that $\chi \in \Gamma_1$ cannot arise.

Case 3: χ is of the form $\neg\chi_1$.
Assume $\chi \in \Gamma_0$, i.e. $|ConfNotDetTrue(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $ConfNotDetTrue(\chi, \psi_0(x), \psi_1(x))$ is $\psi_1(\langle\chi_1\rangle)$, so we have that $\chi_1 \in \Gamma_1$. By the inductive hypothesis, $\chi_1^\beta \in \{1, \frac{1}{2}\}$, i.e. $\chi^\beta \in \{0, \frac{1}{2}\}$, as required.

Now assume $\chi \in \Gamma_1$, i.e. $|\mathit{ConfNotDetFalse}(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $\mathit{ConfNotDetFalse}(\chi, \psi_0(x), \psi_1(x))$ is $\psi_0(\langle\chi_1\rangle)$, so we have that $\chi_1 \in \Gamma_0$. By the inductive hypothesis, $\chi_1^\beta \in \{0, ½\}$, i.e. $\chi^\beta \in \{1, ½\}$, as required.

Case 4: χ is of the form $(\chi_1 \wedge \chi_2)$.
Assume $\chi \in \Gamma_0$, i.e. $|\mathit{ConfNotDetTrue}(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $\mathit{ConfNotDetTrue}(\chi, \psi_0(x), \psi_1(x))$ is $\psi_0(\langle\chi_1\rangle) \vee \psi_0(\langle\chi_2\rangle)$. Thus either $|\psi_0(\langle\chi_1\rangle)| = 1$ or $|\psi_0(\langle\chi_2\rangle)| = 1$. So either $\chi_1 \in \Gamma_0$ or $\chi_2 \in \Gamma_0$. Then by the inductive hypothesis, either $\chi_1^\beta \in \{0, ½\}$ or $\chi_2^\beta \in \{0, ½\}$. This in turn implies that $\chi^\beta \in \{0, ½\}$, as required.

Now assume $\chi \in \Gamma_1$, i.e. $|\mathit{ConfNotDetFalse}(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $\mathit{ConfNotDetFalse}(\chi, \psi_0(x), \psi_1(x))$ is $\psi_1(\langle\chi_1\rangle) \wedge \psi_1(\langle\chi_2\rangle)$. Thus $|\psi_1(\langle\chi_1\rangle)| = 1$ and $|\psi_1(\langle\chi_2\rangle)| = 1$. So χ_1 and χ_2 are both in Γ_1. Then by the inductive hypothesis, $\chi_1^\beta \in \{1, ½\}$ and $\chi_2^\beta \in \{1, ½\}$. This in turn implies that $\chi^\beta \in \{1, ½\}$, as required.

Case 5: χ is of the form $\forall x\, \chi_1(x)$.
Assume $\chi \in \Gamma_0$, i.e. $|\mathit{ConfNotDetTrue}(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $\mathit{ConfNotDetTrue}(\chi, \psi_0(x), \psi_1(x))$ is $\exists n\, \psi_0(\langle\chi_1(\bar{n})\rangle)$. Thus for some natural number n, $|\psi_0(\langle\chi_1(\bar{n})\rangle)| = 1$. Fix such an n. Then $\chi_1(\bar{n}) \in \Gamma_0$. So by the inductive hypothesis, $\chi_1(\bar{n})^\beta \in \{0, ½\}$. This in turn implies that $\chi^\beta \in \{0, ½\}$, as required.

Now assume $\chi \in \Gamma_1$, i.e. $|\mathit{ConfNotDetFalse}(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $\mathit{ConfNotDetFalse}(\chi, \psi_0(x), \psi_1(x))$ is $\forall n\, \psi_1(\langle\chi_1(\bar{n})\rangle)$. So for every natural number n, we have $|\psi_1(\langle\chi_1(\bar{n})\rangle)| = 1$. Hence $\chi_1(\bar{n}) \in \Gamma_1$ for every $n \in \mathbb{N}$. Then by the inductive hypothesis, $\chi_1(\bar{n})^\beta \in \{1, ½\}$ for every $n \in \mathbb{N}$. This in turn implies that $\chi^\beta \in \{1, ½\}$, as required.

Case 6: χ is of the form $\mathit{True}(t)$.
If β is a successor ordinal, let β' be $\beta - 1$. If β is a limit ordinal, let β' be an arbitrary ordinal below β.

Assume $\chi \in \Gamma_0$, i.e. $|\mathit{ConfNotDetTrue}(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $\mathit{ConfNotDetTrue}(\chi, \psi_0(x), \psi_1(x))$ is $\neg \mathit{sentence}(t) \vee \psi_0(t)$. Since χ is a sentence, t does not contain variables. We distinguish two cases: If t does not denote the Gödel code of a sentence, then $\chi^\beta = 0$ if $\beta > 0$, and $\chi^\beta = ½$ if $\beta = 0$; so $\chi^\beta \in \{0, ½\}$, as required. For the second case, t denotes the Gödel code of a sentence, say χ'. Since we already know that $|\neg \mathit{sentence}(t) \vee \psi_0(t)| = 1$, this implies that $|\psi_0(t)| = 1$, i.e. $|\psi_0(\langle\chi'\rangle)| = 1$. Now by the induction hypothesis, $\chi'^{\beta'} \in \{0, ½\}$. So by the definition of Kripke's construction, $\mathit{True}(t)^{\beta'+1} \in \{0, ½\}$, i.e. $\chi^{\beta'+1} \in \{0, ½\}$. If β is a successor ordinal, this means that $\chi^\beta \in \{0, ½\}$, as required. If β is a limit ordinal, we have shown that for any $\beta' < \beta$, $\chi^{\beta'+1} \in \{0, ½\}$. This universal statement in turn implies that $\chi^\beta \in \{0, ½\}$, as required.

Now assume $\chi \in \Gamma_1$, i.e. $|\mathit{ConfNotDetFalse}(\chi, \psi_0(x), \psi_1(x))| = 1$. We have that $\mathit{ConfNotDetTrue}(\chi, \psi_0(x), \psi_1(x))$ is $\mathit{sentence}(t) \wedge \psi_1(t)$. So t denotes the Gödel code of a sentence, say χ', and $|\psi_1(\langle\chi'\rangle)| = 1$. Now by the induction hypothesis, $\chi'^{\beta'} \in \{1, ½\}$. Similarly as in the previous paragraph, this implies $\chi^\beta \in \{1, ½\}$, as required. □

For each ordinal notation α, we define the determinateness operator Δ_α as follows: $\Delta_\alpha\varphi$ is defined to be shorthand notation for $\varphi \wedge \neg Ind(\alpha, \langle\varphi\rangle)$. The following theorem ensures that $\neg\Delta_\alpha\varphi$ can be used to explain one's rejection of φ.

Theorem 1. *If* $|\neg\Delta_\alpha\varphi| = 1$, *then* $|\varphi| = 0$ *or* $|\varphi| = 1/2$.

Proof. By definition, $|\neg\Delta_\alpha\varphi| = |\neg(\varphi \wedge \neg Ind(\alpha, \langle\varphi\rangle))| = |\neg\varphi \vee Ind(\alpha, \langle\varphi\rangle)|$. So if $|\neg\Delta_\alpha\varphi| = 1$, then $|\neg\varphi| = 1$ or $|Ind(\alpha, \langle\varphi\rangle)| = 1$, i.e. by Lemma 1, $|\varphi| = 0$ or $|\varphi| = 1/2$. □

5 Explaining Rejection of the Liar Sentence

Now using the determinateness operator Δ_0, we can explain our rejection of the Liar sentence L by saying $\neg\Delta_0 L$. The statement $\neg\Delta_0 L$ can be made, because $|\neg\Delta_0 L| = 1$. Before giving a rigorous proof for the fact that $|\neg\Delta_0 L| = 1$, we first explain the intuition behind this. This intuitive explanation assumes that $\neg\, True(L)$ and L are identical rather than just equivalent over Peano Arithmetic. Define $\psi(x)$ to be the formula $x = \langle L\rangle \vee x = \langle True(\langle L\rangle)\rangle$. We make the following four observations, which follow directly from the definitions of *ConfNotDetTrue* and *ConfNotDetFalse*:

$$|ConfNotDetTrue(True(\langle L\rangle), \psi(x), \psi(x))| = 1$$
$$|ConfNotDetFalse(True(\langle L\rangle), \psi(x), \psi(x))| = 1$$
$$|ConfNotDetTrue(\neg\, True(\langle L\rangle), \psi(x), \psi(x))| = 1$$
$$|ConfNotDetFalse(\neg\, True(\langle L\rangle), \psi(x), \psi(x))| = 1$$

Further note that under the assumption that $\neg\, True(L)$ and L are identical, these four facts together with the definition of $Ind(\alpha, n)$ on page 11 imply that $Ind(0, \langle L\rangle)$ is satisfied, with $\psi(x)$ taking the role of $\psi_0(x)$ and $\psi_1(x)$.

However, in order to rigorously prove that $|\neg\Delta_0 L| = 1$, we may not use the assumption that $\neg\, True(L)$ and L are identical formulas, because they are not. For the proof to work without this assumption, we need to look into the details of how L is constructed using the diagonalization technique due to Carnap, Gödel and Tarski. For this we give a specific definition of L, which results from unpacking the standard diagonalization technique used to prove the existence of L. First we define the predicate $Q(x, y)$ to be the formalization of the following statement in the language of arithmetic: "x and y are Gödel codes of formulas φ and ψ respectively, and ψ can be obtained from φ by replacing all free instances of z in φ by $\bar{x}$." It is important to note that $Q(x, y)$ can be chosen to be a Σ_0 formula, i.e. a formula without unbounded quantifiers. Now L is defined to be the sentence $\neg\forall m\, \neg(Q(\langle\neg\forall m\, \neg(Q(z, m) \wedge \neg\, True(m))\rangle, m) \wedge \neg\, True(m))$.

In order to see that this formula does indeed work as a Liar sentence, it is helpful to read "$\neg\forall m\, \neg$" as "$\exists m$", i.e. to read the formula as $\exists m\, (Q(\langle\exists m\, (Q(z, m) \wedge \neg\, True(m))\rangle, m) \wedge \neg\, True(m))$. Now this tells us that there exists an m such that m is the Gödel code of the formula that we get when we replace all free occurrences of z in $\exists m\,(Q(z,m) \wedge \neg\,\mathit{True}(m))$ by $\langle\exists m\,(Q(z,m) \wedge \neg\,\mathit{True}(m))\rangle$, and such that $\neg\,\mathit{True}(m)$. But when we replace all free occurrences of z in $\exists m\,(Q(z,m) \wedge \neg\,\mathit{True}(m))$ by $\langle\exists m\,(Q(z,m) \wedge \neg\,\mathit{True}(m))\rangle$, we get $\exists m\,(Q(\langle\exists m\,(Q(z,m) \wedge \neg\,\mathit{True}(m))\rangle, m) \wedge \neg\,\mathit{True}(m))$, i.e. we get the formula that we started with. So the formula claims $\neg\,\mathit{True}(m)$ for its own Gödel code m.

Before presenting the rigorous proof that $|\neg\Delta_0 L| = 1$, we need to define the relation "φ is substitutionally contained in ψ", which is defined over pairs of arithmetic formulas φ and ψ. This relation is defined to be the transitive and reflexive closure of the relation "φ is directly substitutionally contained in ψ", which in turn is defined to hold just in case one of the three following conditions holds:

- ψ is of the form $\neg\varphi$.
- ψ is of the form $\varphi \wedge \chi$ or $\chi \wedge \varphi$ for some formula χ.
- ψ is of the form $\forall n\,\chi(n)$ and φ is of the form $\chi(\bar{k})$ for some natural number k.

Proposition 1. $|\neg\Delta_0 L| = 1$.

Proof. Note that by the definition of Δ_α, it is enough to prove that $|\mathit{Ind}(0, L)| = 1$. For this we need to show that there exist two formulas $\psi_0(x)$ and $\psi_1(x)$ in $\mathcal{L}^{\text{arithm}}$ such that $|\psi_0(\langle L\rangle)| = 1$ and $|\psi_1(\langle L\rangle)| = 1$ and for any formula φ, if $|\psi_0(\langle\varphi\rangle)| = 1$, then $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, and if $|\psi_1(\langle\varphi\rangle)| = 1$, then $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$.

We start by defining two formulas $\psi_0(x)$ and $\psi_1(x)$. The intuition here is that $\psi_0(x)$ is satisfied by certain numbers x that are Gödel codes of formulas that we know to be false or that we assume to have indefinite truth value. Similarly, the intuition for $\psi_1(x)$ is that it is satisfied by certain numbers x that are Gödel codes of formulas that we know to be true or that we assume to have indefinite truth value. The choice of true, false and presumably indefinite formulas included in each of these is chosen in such a way that these assumed truth values confirm themselves and confirm the indefiniteness of L.

For simplicity, we write $\mathbb{L}(x)$ as an abbreviation for $Q(\langle\neg\forall m\,\neg(Q(z,m) \wedge \neg\,\mathit{True}(m))\rangle, x)$. Note that the intuitive meaning of $\mathbb{L}(x)$ is that x is the Gödel code of L.

Define $\psi_0(x)$ to be the formalization of the statement "x is the Gödel code of one of the following formulas:

(a) L
(b) the formula φ' obtained by removing the initial $\neg$ from L
(c) $\neg\,\mathit{True}(\langle L\rangle)$
(d) $\mathit{True}(\langle L\rangle)$
(e) $\neg(\mathbb{L}(\langle L\rangle) \wedge \neg\,\mathit{True}(\langle L\rangle))$
(f) any formula of the form $\mathbb{L}(\bar{n}) \wedge \neg\,\mathit{True}(\bar{n})$, for any natural number n
(g) any false formula that is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$"

Due to the usage of "false" in the last item, one could think that $\psi_0(x)$ could only be formalized with the help of the predicate *True*. However, note that $\mathbb{L}(\bar{n})$ is a Σ_0 formula, and therefore also all formulas substitutionally contained in it are Σ_0 formulas. It is well-known that truth for Σ_0 formulas can be defined in the language of arithmetic, so this usage of the word "false" can be formalized in the language of arithmetic without usage of the predicate *True*.

Similarly, define $\psi_1(x)$ to be the formalization of the statement "x is the Gödel code of one of the following formulas:

(a) L
(b) the formula φ' obtained by removing the initial $\neg$ from L
(c) $\neg\mathit{True}(\langle L\rangle)$
(d) $\mathit{True}(\langle L\rangle)$
(e) $\mathbb{L}(\langle L\rangle) \wedge \neg\mathit{True}(\langle L\rangle)$
(f) any formula of the form $\neg(\mathbb{L}(\bar{n}) \wedge \neg\mathit{True}(\bar{n}))$, for any natural number n
(g) any true formula that is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$"

Just as $\psi_0(x)$, the formula $\psi_1(x)$ can be formalized in the language of arithmetic without usage of the predicate *True*.

We clearly have the required properties that $|\psi_0(\langle L\rangle)| = 1$ and $|\psi_1(\langle L\rangle)| = 1$. So all that remains to be shown are the following two properties:

1. For any $\varphi \in \mathcal{L}^{\text{arithm}}_{\mathit{True}}$, if $|\psi_0(\langle\varphi\rangle)| = 1$ then $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$.
2. For any $\varphi \in \mathcal{L}^{\text{arithm}}_{\mathit{True}}$, if $|\psi_1(\langle\varphi\rangle)| = 1$ then $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$.

In order to prove property 1, assume $|\psi_0(\langle\varphi\rangle)| = 1$. Now based on the definition of $\psi_0(x)$, we distinguish the following seven cases:

(a) $\varphi = L$. Since L begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(L, \psi_0(x), \psi_1(x)) = \psi_1(\langle\varphi'\rangle)$, where φ' is the formula obtained by removing the initial $\neg$ from L. But according to case (b) of the definition of $\psi_1(x)$, $|\psi_1(\langle\varphi'\rangle)| = 1$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.
(b) φ is the formula φ' obtained by removing the initial $\neg$ from L. Note that φ' is the formula $\forall m\, \neg(\mathbb{L}(m) \wedge \neg\mathit{True}(m))$. Since φ' begins with the symbol $\forall$, we need to apply the fifth line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(\varphi', \psi_0(x), \psi_1(x)) = \exists n\, \psi_0(\langle\neg(\mathbb{L}(n) \wedge \neg\mathit{True}(n))\rangle)$. Now by case (e) of the definition of ψ_0, we have that $|\psi_0(\langle\neg(\mathbb{L}(\langle L\rangle) \wedge \neg\mathit{True}(\langle L\rangle))\rangle)| = 1$, which in turn implies that $|\exists n\, \psi_0(\langle\neg(\mathbb{L}(n) \wedge \neg\mathit{True}(n))\rangle)| = 1$, i.e. that $|\mathit{ConfNotDetTrue}(\varphi', \psi_0(x), \psi_1(x))| = 1$, as required.
(c) $\varphi = \neg\mathit{True}(\langle L\rangle)$. Since $\neg\mathit{True}(\langle L\rangle)$ begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(\neg\mathit{True}(\langle L\rangle), \psi_0(x), \psi_1(x)) = \psi_1(\langle\mathit{True}(\langle L\rangle)\rangle)$. But

according to case (d) of the definition of $\psi_1(x)$, $|\psi_1(\langle \mathit{True}(\langle L\rangle)\rangle)| = 1$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(d) $\varphi = \mathit{True}(\langle L\rangle)$. Since $\mathit{True}(\langle L\rangle)$ begins with the symbol *True*, we need to apply the sixth line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(\mathit{True}(\langle L\rangle), \psi_0(x), \psi_1(x)) = \neg \mathit{sentence}(\langle L\rangle) \vee \psi_0(\langle L\rangle)$. According to case (a) of the definition of $\psi_0(x)$, $|\psi_0(\langle L\rangle)| = 1$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(e) $\varphi = \neg(\mathbb{L}(\langle L\rangle) \wedge \neg \mathit{True}(\langle L\rangle))$. Since φ begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \psi_1(\langle \mathbb{L}(\langle L\rangle) \wedge \neg \mathit{True}(\langle L\rangle)\rangle)$. But according to case (e) of the definition of $\psi_1(x)$, this implies that $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(f) φ is a formula of the form $\mathbb{L}(\bar{n}) \wedge \neg \mathit{True}(\bar{n})$, for some natural number n. Since the main connective of φ is $\wedge$, we need to apply the fourth line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \psi_0(\langle \mathbb{L}(\bar{n})\rangle) \vee \psi_0(\langle \neg \mathit{True}(\bar{n})\rangle)$. Now we need to distinguish two cases: If $n = \langle L\rangle$, then by case (c) of the definition of ψ_0, $|\psi_0(\langle \neg \mathit{True}(\bar{n})\rangle)| = 1$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required. If, on the other hand, $n \neq \langle L\rangle$, then by the definition of $Q(x, y)$, we get that $|\mathbb{L}(\bar{n})| = 0$, so by case (g) of the definition of ψ_0, we get that $|\psi_0(\langle \mathbb{L}(\bar{n})\rangle)| = 1$ (since every formula is substitutionally contained in itself), so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(g) φ is a false formula that is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$. We need to distinguish the following five subcases (note that φ cannot be of the form $\mathit{True}(t)$, because $Q(x, y)$ is an arithmetical formula):
1. φ is of the form $t_1 = t_2$. Then by the first line of the definition of *ConfNotDetTrue*, $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))$ is the formula $\neg t_1 = t_2$. Since $t_1 = t_2$ is the false formula φ, we know that $|\neg t_1 = t_2| = 1$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.
2. $\varphi = \bot$. Then by the second line of the definition of *ConfNotDetTrue*, we have that $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))$ is the formula $\neg\bot$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.
3. φ is of the form $\neg\varphi_1$. Then by the third line of the definition of *ConfNotDetTrue*, $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \psi_1(\langle\varphi_1\rangle)$. In this case φ_1 is true and is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$, so by case (g) of the definition of $\psi_1(x)$, $|\psi_1(\langle\varphi_1\rangle)| = 1$, i.e. $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.
4. φ is of the form $\varphi_1 \wedge \varphi_2$. Then by the fourth line of the definition of *ConfNotDetTrue*, $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \psi_0(\langle\varphi_1\rangle) \vee \psi_0(\langle\varphi_2\rangle)$. Since $|\varphi| = 0$, either $|\varphi_1| = 0$ or $|\varphi_2| = 0$. Without loss of generality, assume $|\varphi_1| = 0$. In this case, φ_1 is false and is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$, so by case (g) of the definition of $\psi_0(x)$, $|\psi_1(\langle\varphi_1\rangle)| = 1$, i.e. $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.
5. φ is of the form $\forall x\, \varphi_1(x)$. Then by the fifth line of the definition of *ConfNotDetTrue*, $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \exists n\, \psi_0(\langle\varphi_1(n)\rangle)$.

Since $|\forall x\, \varphi_1(x)| = 0$, there is some natural number n such that $|\varphi_1(\bar{n})| = 0$. In this case $\varphi_1(\bar{n})$ is false and is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$, so by case (g) of the definition of $\psi_0(x)$, $|\psi_0(\langle\varphi_1(\bar{n})\rangle)| = 1$, i.e. $|\exists n\, \psi_0(\langle\varphi_1(n)\rangle)| = 1$, i.e. $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

In order to prove property 2, assume $|\psi_1(\langle\varphi\rangle)| = 1$. Now based on the definition of $\psi_1(x)$, we distinguish the following seven cases:

(a) $\varphi = L$. Since L begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(L, \psi_0(x), \psi_1(x)) = \psi_0(\langle\varphi'\rangle)$, where φ' is the formula obtained by removing the initial $\neg$ from L. But according to case (b) of the definition of $\psi_0(x)$, $|\psi_0(\langle\varphi'\rangle)| = 1$, so $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(b) φ is the formula φ' obtained by removing the initial $\neg$ from L. Note that φ' is the formula $\forall m\, \neg(\mathbb{L}(m) \wedge \neg\mathit{True}(m))$. Since φ' begins with the symbol $\forall$, we need to apply the fifth line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\varphi', \psi_0(x), \psi_1(x)) = \forall n\, \psi_1(\langle\neg(\mathbb{L}(n) \wedge \neg\mathit{True}(n))\rangle)$. Now by case (f) of the definition of ψ_1, we have that $|\psi_1(\langle\neg(\mathbb{L}(\bar{n}) \wedge \neg\mathit{True}(\bar{n}))\rangle)| = 1$ for every natural number n, which in turn implies that $|\forall n\, \psi_1(\langle\neg(\mathbb{L}(n) \wedge \neg\mathit{True}(n))\rangle)| = 1$, i.e. that $|\mathit{ConfNotDetFalse}(\varphi', \psi_0(x), \psi_1(x))| = 1$, as required.

(c) $\varphi = \neg\mathit{True}(\langle L\rangle)$. Since $\neg\mathit{True}(\langle L\rangle)$ begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\neg\mathit{True}(\langle L\rangle), \psi_0(x), \psi_1(x)) = \psi_0(\langle\mathit{True}(\langle L\rangle)\rangle)$. But according to case (d) of the definition of $\psi_0(x)$, $|\psi_1(\langle\mathit{True}(\langle L\rangle)\rangle)| = 1$, so $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(d) $\varphi = \mathit{True}(\langle L\rangle)$. Since $\mathit{True}(\langle L\rangle)$ begins with the symbol *True*, we need to apply the sixth line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\mathit{True}(\langle L\rangle), \psi_0(x), \psi_1(x)) = \mathit{sentence}(\langle L\rangle) \wedge \psi_1(\langle L\rangle)$. Clearly $|\mathit{sentence}(\langle L\rangle)| = 1$. Additionally, according to case (a) of the definition of $\psi_1(x)$, $|\psi_0(\langle L\rangle)| = 1$, so $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(e) $\varphi = \mathbb{L}(\langle L\rangle) \wedge \neg\mathit{True}(\langle L\rangle)$. Since the main connective of φ is $\wedge$, we need to apply the fourth line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \psi_1(\mathbb{L}(\langle L\rangle)) \wedge \psi_1(\neg\mathit{True}(\langle L\rangle))$. Note that $|\mathbb{L}(\langle L\rangle)| = 1$. So according to case (g) of the definition of $\psi_1(x)$, $|\psi_1(\mathbb{L}(\langle L\rangle))| = 1$. Furthermore, according to case (c) of the definition of $\psi_1(x)$, $|\psi_1(\neg\mathit{True}(\langle L\rangle))| = 1$. Thus $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(f) φ is of the form $\neg(\mathbb{L}(\bar{n}) \wedge \neg\mathit{True}(\bar{n}))$ for some natural number n. Since φ begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \psi_0(\mathbb{L}(\bar{n}) \wedge \neg\mathit{True}(\bar{n}))$. Now by case (f) of the definition of ψ_0, we get that $|\psi_0(\mathbb{L}(\bar{n}) \wedge \neg\mathit{True}(\bar{n}))| = 1$, so $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(g) φ is a true formula that is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$. We need to distinguish the following four subcases (note that φ cannot be of the form $\bot$, since we know φ is true, nor of the form $\mathit{True}(t)$, because $Q(x, y)$ is an arithmetical formula):
   1. φ is of the form $t_1 = t_2$. Then by the first line of the definition of *ConfNotDetFalse*, $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))$ is the formula $t_1 = t_2$. Since $t_1 = t_2$ is the true formula φ, we know that $|t_1 = t_2| = 1$, so $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.
   2. φ is of the form $\neg\varphi_1$. Then by the third line of the definition of *ConfNotDetFalse*, $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \psi_0(\langle\varphi_1\rangle)$. In this case φ_1 is false and is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$, so by case (g) of the definition of $\psi_0(x)$, $|\psi_0(\langle\varphi_1\rangle)| = 1$, i.e. $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.
   3. φ is of the form $\varphi_1 \wedge \varphi_2$. Then by the fourth line of the definition of *ConfNotDetFalse*, $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \psi_1(\langle\varphi_1\rangle) \wedge \psi_1(\langle\varphi_2\rangle)$. Since $|\varphi| = 1$, we know $|\varphi_1| = 1$ and $|\varphi_2| = 1$. So for each $i \in \{1, 2\}$, φ_i is true and is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$, so by case (g) of the definition of $\psi_1(x)$, $|\psi_1(\langle\varphi_i\rangle)| = 1$. Therefore $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.
   4. φ is of the form $\forall x\, \varphi_1(x)$. Then by the fifth line of the definition of *ConfNotDetFalse*, $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \forall n\, \psi_1(\langle\varphi_1(n)\rangle)$. Since $|\forall x\, \varphi_1(x)| = 1$, we have $|\varphi_1(\bar{n})| = 0$ for every natural number n. Every instance of $\varphi_1(\bar{n})$ is true and is substitutionally contained in a formula of the form $\mathbb{L}(\bar{n})$, so by case (g) of the definition of $\psi_1(x)$, $|\psi_1(\langle\varphi_1(\bar{n})\rangle)| = 1$. Therefore $|\forall n\, \psi_1(\langle\varphi_1(n)\rangle)| = 1$, i.e. $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

□

6 Explaining Rejection of the Truth-Teller Sentence

We will now look at how the determinateness operator can be used to explain the rejection of the truth-teller sentence. Intuitively, the truth-teller is a sentence that says of itself that it is true. The formal construction of the truth-teller sentence T is similar to that of the Liar sentence L, only that T is designed to be equivalent to $\mathit{True}(\langle T\rangle)$ over Peano Arithmetic, rather than to $\neg\mathit{True}(\langle T\rangle)$. One can easily see that $|T| = 1/2$, so T gets rejected by *KFS*. However, without the determinateness operator, it is not clear how the rejection of T could be explained within the language of *KFS*. But when we do make use of the determinateness operator Δ_0, we can indeed explain our rejection of T by saying $\neg\Delta_0 T$. The intuitive reason for why $|\neg\Delta_0 T| = 1$ is similar to the intuitive reason for why $|\neg\Delta_0 L| = 1$: The definitions of *ConfNotDetTrue* and *ConfNotDetFalse* imply the following two statements:

$$|\mathit{ConfNotDetTrue}(\mathit{True}(\langle T\rangle), x = \langle T\rangle, x = \langle T\rangle)| = 1$$
$$|\mathit{ConfNotDetFalse}(\mathit{True}(\langle T\rangle), x = \langle T\rangle, x = \langle T\rangle)| = 1$$

Now note that under the assumption that $\mathit{True}(T)$ and T are identical, these two facts together with the definition of $\mathit{Ind}(\alpha, n)$ on page 11 imply that $\mathit{Ind}(0, \langle T \rangle)$ is satisfied, with $x = \langle T \rangle$ taking the role of $\psi_0(x)$ and $\psi_1(x)$. This in turn implies that $|\neg \Delta_0 T| = 1$.

Similary as in the case of the Liar sentence, this intuitive explanation can be transformed into a formal proof that $|\neg \Delta_0 T| = 1$ that works without the assumption that $\mathit{True}(T)$ and T are identical formulas.

7 Explaining Rejection of Strengthened Liar Sentences

Once we have successfully dealt with the Liar sentence L as described above, the obvious next question to ask is what happens to a strengthened version of the Liar that makes use of Δ_0. One can construct a strengthened Liar sentence L'_1 that is equivalent to $\neg \Delta_0 \mathit{True}(\langle L'_1 \rangle)$ in classical Peano Arithmetic. Similarly to the case of the strengthened Liar sentence L_1 based on Field's determinateness operator D, one can show that $|L'_1| = 1/2$:

Proposition 2. $|L'_1| = 1/2$

Proof. By the construction of L'_1 and the definition of the truth value $|L'_1|$ based on Kripke's construction, it is clear that $|L'_1| = |\neg \Delta_0 \mathit{True}(\langle L'_1 \rangle)|$.

Assume for a contradiction that $|L'_1| = 1$. Then $|\neg \Delta_0 \mathit{True}(\langle L'_1 \rangle)| = 1$, so by Theorem 1, $|L'_1| = 0$ or $|L'_1| = 1/2$, which contradicts the assumption. So $|L'_1| \neq 1$.

Now assume for a contradiction that $|L'_1| = 0$. By the definition of Δ_0, this means that $|\Delta_0 \mathit{True}(\langle L'_1 \rangle)| = 0$, i.e. $|\neg \Delta_0 \mathit{True}(\langle L'_1 \rangle)| = 1$, i.e. $|L'_1| = 1$, which contradicts the assumption. So $|L'_1| \neq 0$. □

Due to Proposition 2, the rejection of L'_1 cannot be explained in the same way as the rejection of L, because $|\neg \Delta_0 L'_1| = |\neg \Delta_0 \mathit{True}(\langle L'_1 \rangle)| = |L'_1| = 1/2$. Unlike in the case of Field's approach, one cannot get around this problem by iterating the determinateness operator, because if $|\neg \Delta_0 \Delta_0 L'_1|$ were 1, then there would be witnesses $\psi_0(x)$ and $\psi_1(x)$ for $\mathit{Ind}(0, \langle \Delta_0 L'_1 \rangle)$, in which case $\psi'_0(x) := (x = \langle L'_1 \rangle \vee \psi_0(x))$ and $\psi'_1 := (x = \langle \Delta_0 \mathit{True}(\langle L'_1 \rangle) \rangle \vee x = \langle \mathit{True}(\langle L'_1 \rangle) \rangle \vee \psi_1(x))$ would be witnesses for $\mathit{Ind}(0, L'_1)$, which would be a contradiction.

What one can do instead is to use the stronger determinateness operator Δ_1 in order to explain the rejection of φ by stating $\neg \Delta_1 L'_1$. The following proposition establishes that $\neg \Delta_1 L'_1$ is indeed the case.

Proposition 3. $|\neg \Delta_1 L'_1| = 1$.

Proof. Note that by the definition of Δ_α, it is enough to prove that $|\mathit{Ind}(1, L'_1)| = 1$. While proving this, we will make the simplifying assumption that L'_1 is identical to $\neg \Delta_0 \mathit{True}(\langle L'_1 \rangle)$. This simplifying assumption is analogous to the assumption that we made when we gave a first sketch for why $|\neg \Delta_0 L| = 1$ before Proposition 1. This proof can be transformed into a more formal proof by following the idea of the proof of Proposiotion 1. But in order to not obfuscate the novel idea in this proof too much, we make the simplifying assumption here.

We need to show that there exist two formulas $\psi_0(x)$ and $\psi_1(x)$ of depth 1 such that $|\psi_0(\langle L_1' \rangle)| = 1$ and $|\psi_1(\langle L_1' \rangle)| = 1$ and for any formula φ, if $|\psi_0(\langle \varphi \rangle)| = 1$, then $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, and if $|\psi_1(\langle \varphi \rangle)| = 1$, then $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$.

Before defining $\psi_0(x)$ and $\psi_1(x)$, we first analyse the formal structure of $\mathit{Ind}(0, L_1')$. This statement has the logical form $\neg \forall w_0 \, \forall w_1 \, \tau(w_0, w_1)$. Note that all occurrences of *True* in $\tau(w_0, w_1)$ appear in subformulas of the form $\mathit{True}(t_0(w_i, t))$ for $i \in \{1, 2\}$, where $t_0(w, t)$ denotes $\langle \psi(t) \rangle$ for any w denoting the Gödel code of a formula $\psi(x)$.

Define $\psi_0(x)$ to be the formalization of the statement "x is the Gödel code of one of the following formulas:

(a) L_1'
(b) $\Delta_0 \, \mathit{True}(\langle L_1' \rangle)$
(c) $\mathit{True}(\langle L_1' \rangle)$
(d) $\mathit{Ind}(0, \langle \mathit{True}(\langle L_1' \rangle) \rangle)$
(e) any false formula in which all occurrences of *True* appear in subformula of the form $\mathit{True}(t_0(\bar{n}, t))$, where n is the Gödel code of a formula $\psi(x)$ of depth 0"

Due to the usage of the word "false" in the last item, it may seem that we need a formula of depth 2 to formalize $\psi_0(x)$. However, we can actually formalize $\psi_0(x)$ with a formula of depth 1 by noting that the last item could have equivalently been formulated as follows:

– "any formula χ satisfying the following two criteria:
  (i) All occurrences of *True* in χ appear in subformula of the form $\mathit{True}(t_0(\bar{n}, t))$, where n is the Gödel code of a formula $\psi(x)$ of depth 0.
  (ii) The depth 0 formula χ' that is the result of replacing each subformula of the form $\mathit{True}(t_0(\bar{n}, t))$ in χ by $\psi(t)$, where $\psi(x)$ is the formula whose Gödel code is n, is true."

Similarly, define $\psi_1(x)$ to be the formalization of the statement "x is the Gödel code of one of the following formulas:

(a) L_1'
(b) $\Delta_0 \, \mathit{True}(\langle L_1' \rangle)$
(c) $\mathit{True}(\langle L_1' \rangle)$
(d) $\neg \mathit{Ind}(0, \langle \mathit{True}(\langle L_1' \rangle) \rangle)$
(e) $\forall w_1 \, \forall w_2 \, \tau(w_1, w_2)$
(f) any formula of the form $\forall w_2 \, \tau(\bar{n}, w_2)$ for a natural number n
(g) any true formula in which all occurrences of *True* appear in subformula of the form $\mathit{True}(t_0(\bar{n}, t))$, where n is the Gödel code of a formula $\psi(x)$ of depth 0"

Just as in the case of $\psi_0(x)$, $\psi_1(x)$ can be formalized as a formula of depth 1.

We clearly have the required properties that $|\psi_0(\langle L_1' \rangle)| = 1$ and $|\psi_1(\langle L_1' \rangle)| = 1$. So all that remains to be shown are the following two properties:

1. For any $\varphi \in \mathcal{L}^{\text{arithm}}_{True}$, if $|\psi_0(\langle\varphi\rangle)| = 1$ then $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$.
2. For any $\varphi \in \mathcal{L}^{\text{arithm}}_{True}$, if $|\psi_1(\langle\varphi\rangle)| = 1$ then $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$.

In order to prove property 1, assume $|\psi_0(\langle\varphi\rangle)| = 1$. Now based on the definition of $\psi_0(x)$, we distinguish the following five cases:

(a) $\varphi = L_1'$. By the simplifying assumption, this means that $\varphi = \neg\Delta_0\, \mathit{True}(\langle L_1'\rangle)$. Since φ begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \psi_1(\langle\Delta_0\, \mathit{True}(\langle L_1'\rangle)\rangle)$. But according to case (b) of the definition of $\psi_1(x)$, $|\psi_1(\langle\Delta_0\, \mathit{True}(\langle L_1'\rangle)\rangle)| = 1$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(b) $\varphi = \Delta_0\, \mathit{True}(\langle L_1'\rangle)$. By the definition of Δ_0, this means that $\varphi = \mathit{True}(\langle L_1'\rangle) \wedge \neg \mathit{Ind}(0, \langle \mathit{True}(\langle L_1'\rangle)\rangle)$. Since the main connective of φ is $\wedge$, we need to apply the fourth line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \psi_0(\langle \mathit{True}(\langle L_1'\rangle)\rangle) \vee \psi_0(\neg \mathit{Ind}(0, \langle \mathit{True}(\langle L_1'\rangle)\rangle))$. But according to case (c) of the definition of $\psi_0(x)$, $|\psi_0(\langle \mathit{True}(\langle L_1'\rangle)\rangle)| = 1$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(c) $\varphi = \mathit{True}(\langle L_1'\rangle)$. Since $\mathit{True}(\langle L_1'\rangle)$ begins with the symbol *True*, we need to apply the sixth line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(\mathit{True}(\langle L_1'\rangle), \psi_0(x), \psi_1(x)) = \neg \mathit{sentence}(\langle L_1'\rangle) \vee \psi_0(\langle L_1'\rangle)$. According to case (a) of the definition of $\psi_0(x)$, $|\psi_0(\langle L_1'\rangle)| = 1$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(d) $\varphi = \mathit{Ind}(0, \langle \mathit{True}(\langle L_1'\rangle)\rangle)$. In this case, $\varphi = \neg\forall w_1\, \forall w_2\, \tau(w_1, w_2)$. Since φ begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetTrue*, according to which $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \psi_1(\langle\forall w_1\, \forall w_2\, \tau(w_1, w_2)\rangle)$. But according to case (e) of the definition of $\psi_1(x)$, $|\psi_1(\langle\forall w_1\, \forall w_2\, \tau(w_1, w_2)\rangle)| = 1$, so $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(e) φ is a false formula in which all occurrences of *True* appear in subformula of the form $\mathit{True}(t_0(\bar{n}, t))$, where n is the Gödel code of a formula $\psi(x)$ of depth 0. In this case, the fact that $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$ can be proven in a similar way as case (g) for property 1 in the proof of Proposition 1. However, additionally to the five cases considered there, we now have a sixth case, namely that φ is of the form $\mathit{True}(t_0(\bar{n}, t))$, where n is the Gödel code of a formula $\psi(x)$ of depth 0. In this case, we have that $\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x)) = \neg \mathit{sentence}(t_0(\bar{n}, t)) \vee \psi_0(t_0(\bar{n}, t)) = \neg \mathit{sentence}(\langle\psi(t)\rangle) \vee \psi_0(\langle\psi(t)\rangle)$. Since φ is false, $\psi(t)$ is false, so $|\psi_0(\psi(t))| = 1$ by case (e) of the definition of $\psi_0(x)$. Thus $|\mathit{ConfNotDetTrue}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

In order to prove property 2, assume $|\psi_1(\langle\varphi\rangle)| = 1$. Now based on the definition of $\psi_1(x)$, we distinguish the following seven cases:

(a) $\varphi = L_1'$. By the simplifying assumption, this means that $\varphi = \neg\Delta_0\, \mathit{True}(\langle L_1'\rangle)$. Since φ begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \psi_0(\langle\Delta_0\, \mathit{True}(\langle L_1'\rangle)\rangle)$. But according to case (b) of the definition of $\psi_0(x)$, $|\psi_0(\langle\Delta_0\, \mathit{True}(\langle L_1'\rangle)\rangle)| = 1$, so $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(b) $\varphi = \Delta_0\, \mathit{True}(\langle L_1'\rangle)$. By the definition of Δ_0, this means that $\varphi = \mathit{True}(\langle L_1'\rangle) \wedge \neg \mathit{Ind}(0, \langle \mathit{True}(\langle L_1'\rangle)\rangle)$. Since the main connective of φ is $\wedge$, we need to apply the fourth line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \psi_1(\langle \mathit{True}(\langle L_1'\rangle)\rangle) \wedge \psi_1(\neg \mathit{Ind}(0, \langle \mathit{True}(\langle L_1'\rangle)\rangle))$. According to case (c) of the definition of $\psi_1(x)$, $|\psi_1(\langle \mathit{True}(\langle L_1'\rangle)\rangle)| = 1$; and according to case (d) of the definition of $\psi_1(x)$, $|\psi_1(\langle\neg \mathit{Ind}(0, \langle \mathit{True}(\langle L_1'\rangle)\rangle)\rangle)| = 1$. Therefore $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(c) $\varphi = \mathit{True}(\langle L_1'\rangle)$. Since $\mathit{True}(\langle L_1'\rangle)$ begins with the symbol *True*, we need to apply the sixth line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\mathit{True}(\langle L_1'\rangle), \psi_0(x), \psi_1(x)) = \mathit{sentence}(\langle L_1'\rangle) \wedge \psi_0(\langle L_1'\rangle)$. Clearly $|\mathit{sentence}(\langle L_1'\rangle)| = 1$. Furthermore, according to case (a) of the definition of $\psi_1(x)$, $|\psi_1(\langle L_1'\rangle)| = 1$, so $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(d) $\varphi = \neg \mathit{Ind}(0, \langle \mathit{True}(\langle L_1'\rangle)\rangle)$. Since φ begins with the symbol $\neg$, we need to apply the third line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \psi_0(\langle \mathit{Ind}(0, \langle \mathit{True}(\langle L_1'\rangle)\rangle)\rangle)$. But according to case (d) of the definition of $\psi_0(x)$, $|\psi_0(\langle \mathit{Ind}(0, \langle \mathit{True}(\langle L_1'\rangle)\rangle)\rangle)| = 1$, so $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(e) $\varphi = \forall w_1 \forall w_2\, \tau(w_1, w_2)$. Since φ begins with the symbol $\forall$, we need to apply the fifth line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \forall n\, \psi_1(\langle\forall w_2\, \tau(\bar{n}, w_2)\rangle)$. But according to case (f) of the definition of $\psi_1(x)$, $|\psi_1(\langle\forall w_2\, \tau(\bar{n}, w_2)\rangle)| = 1$ for any natural number n, so $|\forall n\, \psi_1(\langle\forall w_2\, \tau(\bar{n}, w_2)\rangle)| = 1$, i.e. $|\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(f) φ is of the form $\forall w_2\, \tau(\bar{n}, w_2)$ for a natural number n. Since φ begins with the symbol $\forall$, we need to apply the fifth line of the definition of *ConfNotDetFalse*, according to which $\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x)) = \forall m\, \psi_1(\langle\tau(\bar{n}, \bar{m})\rangle)$. Now suppose for a contradiction that there is some m such that $|\tau(\bar{n}, \bar{m})| \neq 1$. Since $\tau(\bar{n}, \bar{m})$ has depth 1, it cannot have truth-value ½, so this implies $|\tau(\bar{n}, \bar{m})| = 0$. This in turn means that $|\forall w_0 \forall w_1 \tau(w_0, w_1)| = 0$, i.e. $|\mathit{Ind}(0, L_1')| = 1$, so $|\Delta_0 L_1'| = 0$, i.e. $|L_1'| = |\neg\Delta_0 L_1'| = 1$, contradicting Proposition 2. Thus for every natural number m, $|\tau(\bar{n}, \bar{m})| = 1$. This together with case (g) of the definition of $\psi_1(x)$ and the fact that all occurrences of *True* in $\tau(\bar{n}, \bar{m})$ appear in subformulas of the form $\mathit{True}(t_0(w, t))$ implies that $|\psi_1(\langle\tau(\bar{n}, \bar{m})\rangle)| = 1$ for all natural numbers m. Thus $|\forall m\, \psi_1(\langle\tau(\bar{n}, \bar{m})\rangle)| = |\mathit{ConfNotDetFalse}(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

(g) φ is a true formula in which all occurrences of *True* appear in subformula of the form $\mathit{True}(t_0(\bar{n}, t))$, where n is the Gödel code of a formula $\psi(x)$ of depth 0. In this case, the fact that $|ConfNotDetTrue(\varphi, \psi_0(x), \psi_1(x))| = 1$ can be proven in a similar way as case (g) for property 2 in the proof of Proposition 1. However, additionally to the five cases considered there, we now have a sixth case, namely that φ is of the form $True(t_0(\bar{n}, t))$, where n is the Gödel code of a formula $\psi(x)$ of depth 0. In this case, we have that $ConfNotDetTrue(\varphi, \psi_0(x), \psi_1(x)) = sentence(t_0(\bar{n}, t)) \wedge \psi_0(t_0(\bar{n}, t)) = sentence(\langle\psi(t)\rangle) \wedge \psi_0(\langle\psi(t)\rangle)$. Clearly, $|sentence(\langle\psi(t)\rangle)| = 1$. Furthermore, since φ is true, $\psi(t)$ is true, so $|\psi_1(\psi(t))| = 1$ by case (g) of the definition of $\psi_1(x)$. Thus $|ConfNotDetFalse(\varphi, \psi_0(x), \psi_1(x))| = 1$, as required.

□

Similarly as in the case of the transfinite iterations of Field's determinateness operator, we can explain the rejection of ever-stronger strengthened Liar sentences that use ever-higher-indexed determinateness operators by always using a determinateness operator of a higher index than the one used in the strengthened Liar sentence whose rejection we want to explain.

8 Conditional reasoning

Field motivates his extension of *KFS* to a theory with a conditional not only through the fact that this allows him to express his rejection of the Liar sentence in the object language, but also based on the argument that it is desirable to have a conditional that satisfies certain basic properties that one would expect of a conditional but that are not satisfied by the material implication $\supset$ of *KFS*, e.g. that $\varphi \to \varphi$ and $\varphi \to (\varphi \vee \psi)$ are logical truths for any choice of φ and ψ. In this section, we present some objections to Field's motivation for extending *KFS* with such a conditional, and give an outline of what kind of conditional we believe could be introduced to model conditional reasoning. The formal definition of a conditional along the lines that we sketch here is left to future work.

Prima facie it would be desirable to have a conditional that satisfies both conditional proof (the rule of $\to$-introduction of natural deduction, i.e. the principle that whenever we have $\Gamma, \varphi \vdash \psi$, we also have $\Gamma \vdash \varphi \to \psi$) and modus ponens, i.e. the rule $\varphi, \varphi \to \psi \vdash \psi$. Due to Curry's paradox ("If this sentence is true, then the earth is flat"), we know that a theory of truth cannot consistently allow for both of these features of the conditional if it validates the rules $\varphi \vdash True(\langle\varphi\rangle)$ and $True(\langle\varphi\rangle) \vdash \varphi$ as well as the usual structural rules of logic. So in the paracomplete setting, where we do indeed want to validate the rules $\varphi \vdash True(\langle\varphi\rangle)$ and $True(\langle\varphi\rangle) \vdash \varphi$ as well as the usual structural rules, we need to give up some prima facie desirable features of the conditional. The material implication $\supset$ of *KFS* satisfies modus ponens, but is does not satisfy conditional proof, and does not even satisfy principles like having $\varphi \to \varphi$ and $\varphi \to (\varphi \vee \psi)$ as logical truths for any choice of φ and ψ. In his monograph *Saving Truth from Paradox* [4], Field defines a conditional which just like the material implication satisfies modus ponens and does not satisfy conditional proof, but satisfies some desirable principles not satisfied by the material implication, for example that for any formulas φ and ψ, $\varphi \to \varphi$ and $\varphi \to (\varphi \vee \psi)$ are logical truths, symbolically expressed as $\vdash \varphi \to \varphi$ and $\vdash \varphi \to (\varphi \vee \psi)$.

We would like to point out that principles like $\vdash \varphi \to \varphi$ and $\vdash \varphi \to (\varphi \vee \psi)$ are derivable from conditional proof, so a conditional that satisfies conditional proof would automatically satisfy these principles. In *Saving Truth from Paradox* [4], Field does not provide any philosophical argument why $\varphi \to \varphi$ and $\varphi \to (\varphi \vee \psi)$ should be accepted while other similar principles following from conditional proof should be rejected. We therefore propose to look at another possible avenue, namely to fully accept conditional proof and therefore also principles like $\vdash \varphi \to \varphi$ and $\vdash \varphi \to (\varphi \vee \psi)$ that follow from it, but to restrict the applicability of modus ponens in order to save the consistency of the theory in light of Curry's paradox.

It might sound counterintuitive to limit modus ponens. In order to explain how one can make sense of such a limitation, we would like to compare conditional statements of the form "If φ then ψ" (denoted as $\varphi \to \psi$) with validity statements like "It is logically valid to conclude ψ from the assumption φ", which as a meta-level claim is usually denoted as $\varphi \vdash \psi$, whereas as an object-level claim, it is denoted as $Val(\langle\varphi\rangle, \langle\psi\rangle)$.

Beall and Murzi [1] introduce a principle called VD that can be seen as the analogue of modus ponens for validity:

$$\text{(VD)} \qquad \varphi,\, Val(\langle\varphi\rangle, \langle\psi\rangle) \vdash \psi$$

In his response to Beall and Murzi, Field [5] himself points out that this principle is highly problematic. He compares it to the principle that if $\vdash Val(\langle\varphi\rangle, \langle\psi\rangle)$, then $\varphi \vdash \psi$, and points out that VD is stronger and more problematic than this principle, because VD concerns what follows from the assumption of $Val(\langle\varphi\rangle, \langle\psi\rangle)$, whereas the other principle is only applicable when $Val(\langle\varphi\rangle, \langle\psi\rangle)$ is unconditionally established.

The basic idea of our approach to conditional reasoning is to limit modus ponens, analogously to the rejection of VD. Of course validity assertions and conditional assertions behave differently in many respects. Most notably, $\varphi \to \psi$ may be true even when $Val(\langle\varphi\rangle, \langle\psi\rangle)$ is not, because the truth of $\varphi \to \psi$ may depend on a non-logical truth that can be used for deriving ψ from φ. However, we think that this difference between validity assertions and conditional assertions is orthogonal to the analogy between modus ponens and VD. In other words, we believe that it is possible to develop a formal theory of conditional reasoning that accounts for the just-mentioned difference between $\varphi \to \psi$ and $Val(\langle\varphi\rangle, \langle\psi\rangle)$, but that keeps intact the analogy between modus ponens and VD and thus rejects certain instances of modus ponens. In particular, such a theory would still accept the principle that when $\vdash \varphi \to \psi$, then $\varphi \vdash \psi$, and it would furthermore accept instances of modus ponens, i.e. of the rule $\varphi, \varphi \to \psi \vdash \psi$, in which φ is known to be *grounded* with respect to the conditional $\to$. By φ being *grounded* with respect to the conditional $\to$ we mean that when one replaces sub-formulas of φ of the form $True(t)$ by formulas χ whose Gödel codes are potential referents of t, and iterates this replacement process transfinitely, thus potentially producing a “formula” with an infinitely deep syntactical structure, one never gets an infinite nesting of the conditional connective $\rightarrow$.

One can easily see that the Curry sentence is not grounded with respect to the conditional $\rightarrow$, so that this limited form of modus ponens cannot be applied to it. Thus this limitation of modus ponens is indeed enough to avoid Curry's paradox, even when conditional proof is permitted without limitations. On the other hand, a conditional with these features cannot be used to define a determinacy operator in the way Field has done it.

Developing the details of this envisioned formal theory of conditional reasoning goes beyond the scope of this paper. The reason why we sketched this theory here was to clarify why we believe that the problem of developing a formal notion of determinateness in order to explain the rejection of certain formulas whose negation cannot be asserted can be viewed as independent from the problem of developing a theory of conditional reasoning. This paper has focused on the first of these two problems, and only sketched a potential way of tackling the second problem, whose formal details are therefore left to future work.

9 Conclusion

Just like Field [4], we have defined a determinateness operator to explain the rejection of the Liar sentence within the object language of our theory of truth. Unlike Field's determinateness operator, our determinateness operator can be defined within the theory *KFS* and does not require *KFS* to be extended by a conditional that is undefinable within *KFS*. Field's approach, on the other hand, is based on the idea of extending *KFS* through a semantic construction that involves the combination of a revision-rule construction for the semantics of the conditional $\rightarrow$ and Kripke's construction for the semantics of *True*, so the overall semantic construction is rather complicated. Our approach, on the other hand, can be completely developed within the theory *KFS* that comes out of Kripke's construction. Thus we avoid the complications of the extended theory for the conditional that Field has developed while achieving an equally good resolution of the Liar paradox and strengthened versions of it.

Unlike Field's determinateness operator, our determinateness operator cannot be strengthened by iterating it. Instead, we have defined a transfinite hierarchy of ever stronger determinateness operators that are not definable as iterations of the weakest determinateness operator. With this transfinite hierarchy of determinateness operators we can explain the Liar paradox and strengthened versions of it in much the same way as Field does, only that we use our stronger determinateness operators where Field uses iterations of his determinateness operator.

Finally we have sketched our take on conditional reasoning in order to explain why we consider the development of a formal notion of determinateness a separate problem from the development of a theory of conditional reasoning. While this paper provides a detailed formal account of how to develop a hierarchy of determinateness operators within *KFS*, the development of a theory of conditional reasoning was only sketched and is therefore left for future work.

References

1. Beall, J., Murzi, J.: Two flavors of curry's paradox. The Journal of Philosophy **110**(3), 143–165 (2013)
2. Denecker, M., Vennekens, J.: The Well-Founded Semantics Is the Principle of Inductive Definition, Revisited. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR 2014, Vienna, Austria, July 20-24, 2014 (2014)
3. Feferman, S.: Reflecting on incompleteness. The Journal of Symbolic Logic **56**(1), 1–49 (1991)
4. Field, H., et al.: Saving truth from paradox. Oxford University Press (2008)
5. Field, H., et al.: Disarming a paradox of validity. Notre Dame Journal of Formal Logic **58**(1), 1–19 (2017)
6. Kripke, S.: Outline of a theory of truth. The journal of philosophy **72**(19), 690–716 (1976)
7. Priest, G., et al.: In contradiction. Oxford University Press (2006)
8. Ripley, D.: Comparing substructural theories of truth. Ergo, an Open Access Journal of Philosophy **2** (2015)
9. Simmons, K.: Paradoxes of denotation. Philosophical Studies: An International Journal for Philosophy in the Analytic Tradition **76**(1), 71–106 (1994)
10. Van Gelder, A., Ross, K.A., Schlipf, J.S.: The well-founded semantics for general logic programs. Journal of the ACM (JACM) **38**(3), 619–649 (1991)