Gödel Sentences and Dialetheism

Gödel's famous *First Incompleteness Theorem* shows the presence of gaps of *Negation Incompleteness* in sufficiently expressive and (ω-)consistent formal systems. Graham Priest (in *In Contradiction*) put forth a special argument from Gödelian reasoning for dialetheism.

<u>The argument</u>: Supposing the *Church-Turing Thesis* the informal reasoning that the Gödel sentence G is true can be captured by a formal system N for naïve or informal proofs. Sentence G for some logical system expressive enough says of itself that it is not provable in that very system. By assumption N is the most expressive system, thus expressive enough. The crucial point: sentence G for the system N *is* provable in N. By the informal argument usually given for its truth: N as correct system cannot prove a sentence which says of itself that it is not provable in N, thus G is not provable in N, but this is what G says, so G is true. Now, this very argument can be made – by assuming that N is our formal system for *all* naïve and informal proofs – *in* N, thus G is provable in N, after all. And thus, negation incompleteness is *avoided* (as G is provable), at the cost of *inconsistency*: we have one provable sentence (G itself) saying that G is not provable, and one provable sentence saying that G is provable, as system N meets the conditions not only to express and represent the concept of provability, but to express – and even represent – the concept of truth as well. Sentence G, thus, is a paradox (a *dialetheia*) in N, with respect to the provability predicate.

Dialetheism accepting inconsistency in any case (dealing thus with the semantic antinomies) can reap the benefit of avoiding *Negation Incompleteness*. For Priest, thus, Gödel's argument shows a dilemma: either we achieve *Negation Completeness* or consistency. And, as we can reason to the truth of G, achieving *Negation Completeness* has to be our priority.

But isn't the contradiction engendered by G too much even for the dialetheist?
Having G means having a provable sentence claiming truly its own non-provability, and because G is provable ¬G is true, thus it should be *provable* in N as well, as we can informally reason to its truth, which should be captured in N.
As we want the paraconsistent equivalent to correctness N should not prove sentences that are *false only*. Paraconsistent system may prove sentences which are true and false at the same

time, but proving a sentence that is *only false* (i.e. not true at the same time) would be a *hyper-contradiction* putting the system used into doubt. G cannot not be *false only*, as it will then be provable given convention (T) for truth

(T) True("p") $\equiv$ p

from right to left (so-called 'T-in') and *Contraposition* (i.e. proving from the negation of its truth again $\neg$G, which says by *Double Negation Elimination* that G is provable in N, which by correctness of N means that G is true). Thus, if G is *at least* true, it is not provable, as it truly says, given (T) from left to right (so-called 'T-out'). As we can informally reason to G's truth, this reasoning is captured by a proof *in* N, N thus proving G, again. G being by disquotation also a paradox of truth (another *dialetheia*), because "Provable" applies to it as well as "$\neg$Provable", "True" applies as well as "$\neg$True".

Consider, now, the following critical argument:
Given Priest's line of argument, we have:

(1) $\vdash_N (\exists x)\text{Proof}(x,G)$

By paraconsistent correctness and T-out we thus get:

(2) $\vdash_N \neg(\exists x)\text{Proof}(x,G)$

This means that there is some number *m* which both codes a proof of G, by (1), and does not, by (2)! This is too strange as 'Proof' (i.e. being a proof) is a *primitive recursive*, algorithmic and deterministic relation (between one number coding the proof of a sentence and the number of that sentence). So, could we ever have (1) and (2) together? Contradiction in elementary arithmetic? – Some rhetorical questions: end of argument.

Reply: Without mystery we cannot give up the *Church-Turing Thesis*, and the existence of N. Representing the presence of a proof by 'Proof' should be available.
One measure might be a paraconsistent account of algorithms: one and the same deterministic algorithm providing different answers or an inconsistent answer on the same input. This is a *desideratum*, however. And it seems to clash with our intuitive notion of algorithm.
It is no option either to stress realism again (claiming provability to be different from truth) as G is available and provably *true* in N.
One may claim that *m* makes (1) true by *really coding* the proof of G, and (2) is thereby another provable sentence of N resulting in *saying of m falsely* that it does not code the proof of G. Thus, *m* turns out to be an inconsistent object in N. And (2) is another contradiction, its

truth being guaranteed by the provability of G. As there are other inconsistent objects in N already, this may not worry a dialetheist. The dialetheist may worry as coding a proof is a *primitive recursive* relation. So that saying of *m* that it does not code a proof of G contradicts the result of a respective algorithm. The algorithm, however, will in no way be *suppressed* in its workings: it delivers that *m* codes a proof of G, making (1) true. (2) just falsely contradicts this positive output. It cannot interrupt the algorithm which properly establishes (1).

A dialetheist can, therefore, embrace the truth of G, and so avoid Gödelian gaps of *Negation Incompleteness*. (2) is a price to be paid, supposedly without any detrimental consequences.