The inherent risks in using a name-forming function at object language level

Ferenc András
http://ferenc.andrasek.hu

The Reasoner 9/5, May 2015

The truth problem is one of the central problems of philosophy. Nowadays, every major theory of truth that applies to formal languages utilizes devices referring to formulae. Such devices include name-forming functions. The theory of truth discussed in this paper applies to strict formal logic languages, the critique of which must, therefore, also obey mathematical rigour. This is why I have used formal logic derivations below rather than the argumentation of ordinary language.

The first derivation below demonstrates that some name-forming functions produce an antinomy. However, in the first instance we can straightforwardly escape from the trap by denying the existence of that given function. In the second derivation, however, I prove that the citation function can also produce antinomy. Furthermore, in this case we cannot easily escape from the trap, because denying the existence of the citation function is counterintuitive.

In mathematics it is often difficult to understand that two formulae are equivalent, although it is evident that a formula is always equivalent to itself. The axiom that follows (Axiom A) makes a similar claim in the domain of logical formulae.

Let L be a first-order logic language, including the one-to-one name-forming function ξ, which is a mapping from formulae to names. Let ξ^{-1} be the inverse of ξ. The iteration of the operator ξ is acceptable. If z is a formula name of L, then $\xi^{-1}(z)$ is a formula of L. There are no restrictions for ξ, and hence, Axiom A, which is a version of Tarski's β, is intuitive:

(A) $\exists\xi$((ξ is a one-to-one function of L in the domain of L WFFs) & $\forall x \forall y$(if x and y are L sentence names and $x = y$, then $(\xi^{-1}(x) \leftrightarrow \xi^{-1}(y))$))

The following derivations are based on Quine's deduction technique outlined in his book"Methods of Logic". What follows is a proof by contradiction: the asterisk indicates the original premise which is assumed, and all subsequent consequences of that premise. The absence of an asterisk prefix to (7) indicates that that line not depend on previous premises, but instead holds absolutely. The absence of an asterisk thus indicates a claim to validity. The argument, and specifically premise (2), is a formulation of Tarski's derivation of an antinomy from the unrestricted use of quasi-quotation. (For further clarification of this point and its relation to Tarksi see the discussion that follows this derivation.)

There are no specific restrictions for ξ; so, we apply premise (2):

*(2)	$\exists x$(x is a sentence name of L &	
	$x = \gamma(\forall y(x = y \rightarrow \sim \gamma^{-1}(y))))$	(A)
	γ is such a bijective function	
*(3)	$z = \gamma(\forall y(z = y \rightarrow \sim \gamma^{-1}(y)))$	(2) z
*(4)	$\gamma^{-1}(z) \leftrightarrow \forall y(z = y \rightarrow \sim \gamma^{-1}(y))$	(A) (3)
*(5)	$\gamma^{-1}(z) \leftrightarrow (z = z \rightarrow \sim \gamma^{-1}(z))$	(4) $y = z$
*(6)	$\gamma^{-1}(z) \leftrightarrow \sim \gamma^{-1}(z)$	(5)
(7)	If (A) and (2) are true, then $\gamma^{-1}(z) \leftrightarrow \sim \gamma^{-1}(z)$	(A) (2)
(8)	If (A) is true, then there is no such γ function.	(7)

Since axiom (A) is a plausible assumption, it is preferable to deny the existence of function γ and preserve (A). Nonetheless, it is clear that this argument is readily transformable to another inference based on another name-forming function. What if we substitute function γ with the so-called citation function?

Tarski designated the " $(\lambda x)\ulcorner x \urcorner$ " function as the "citation function" or "quasi-quotation," distinguishing the usage of the citation function from the normal usage of quotation marks. The quasi-quotation is only one possible name-forming function among many others, like Gödel numbering.

In the domain of well-formed formulas (WFFs), the quasi-quotation function is not a partial function; however, other name-forming functions can be partial functions. This means that one can form an individual constant of any WFF or term by applying quasi-quotation, and can quote quote-names; however, other name-forming functions leave one's hands tied: in other cases, it is not permitted to iterate the use of name-forming functions. It must be noted that there is no problem in using quasi-quotation at the metalanguage level in the domain of formulas of the object language. In this case, $\ulcorner p \urcorner$ is not a name of object language names, but a name of metalanguage names. In another case, if one applies a formal logic language including quasi-quotation at the object language level, then one has to handle this device very carefully. We know from Tarski that quasi-quotation itself is a very risky device which can produce antinomy. Tarski only sketched the argument:

> Let the symbol 'c' be a typographical abbreviation of the expression 'the sentence printed on this page, line 6 from the top'. We consider the following statement: for all p, if c is identical with the sentence 'p', then not p ... We establish empirically:
> (α) the sentence 'for all p, if c is identical with the sentence 'p', then not p' is identical with c.
> In addition we make only a single supplementary assumption which concerns the quotation-function and seems to raise no doubts:
> (β) for all p and q, if sentence 'p' is identical with sentence 'q', then p if and only if q.
> By means of elementary logical laws we easily derive a contradiction from the premises (α) and (β) (Alfred Tarski, 1936: The concept of truth in formalized languages. In Logic, Semantics, Metamathematics, OUP, p.162). Although Tarski's argument has been reconstructed before, the reconstruction below appears to be novel.

Let " $(\lambda x)\ulcorner x \urcorner^{-1}$ " symbolize the denotation function: if x is a formula name of L, then $\ulcorner x \urcorner^{-1}$ is a formula of L. Let us then consider the following inference:

(B) ((($\lambda x)\ulcorner x \urcorner$ is a one-to-one citation function of L in the domain of L WFFs) & $\forall x \forall y$(if x and y are L sentence names and $x = y$, then $\ulcorner x \urcorner^{-1} \leftrightarrow \ulcorner y \urcorner^{-1}$))

There are no specific restrictions for $(\lambda x)\ulcorner x \urcorner$ name-forming function, so we apply premise (2):

*(2)	$\exists x(x$ is a sentence name of L & $x = \ulcorner\ \forall y(x = y \to \sim \ulcorner y \urcorner^{-1})\urcorner)$	(B)
*(3)	$z = \ulcorner(\forall y(z = y \to \sim \ulcorner y \urcorner^{-1})\urcorner$	(2) z
*(4)	$\ulcorner z \urcorner^{-1} \leftrightarrow \forall y(z = y \to \sim \ulcorner y \urcorner^{-1})$	(B) (3)
*(5)	$\ulcorner z \urcorner^{-1} \leftrightarrow (z = z \to \sim \ulcorner z \urcorner^{-1})$	(4) $y = z$
*(6)	$\ulcorner z \urcorner^{-1} \leftrightarrow \sim \ulcorner z \urcorner^{-1}$	(5)
(7)	If (B) and (2) are true, then $\ulcorner z \urcorner^{-1} \leftrightarrow \sim \ulcorner z \urcorner^{-1}$	(B) (2)
(8)	If (B) is true, then there is no such $(\lambda x)\ulcorner x \urcorner$ name forming function.	(7)

Thus, applying the citation function as a name-forming device at the object language level does indeed result in an antinomy. It follows from the above-mentioned inference that any theory including (B) – similar to the Revision Theory of Truth – is inconsistent. Gupta and Belnap declare in their seminal work: " . . . L contains for each sentence A a quotational name 'A'. The interpretation I assigns to the name 'A' the sentence A." (Anil Gupta and Nuel Belnap, 1993: The Revision Theory of Truth, The MIT Press, p.75). Philip Kremer says: "L^- will have a quote name 'A' for every sentence A of L^-." (2014: The Revision Theory of Truth, *The Stanford Encyclopedia of Philosophy*, Summer 2014 Edition).

The conclusion is that it is advisable to use name-forming devices at the metalanguage level; otherwise, at the object language level, one must carefully restrict the domain of the name-forming function to avoid its iterated usage. (I thankfully acknowledge the helpful assistance received from Peter Fekete.)