The Revenge Form of Fitch’s Paradox for Russellian Typing Knowledge

Jiří Raclavský

October 29, 2015

Abstract

It is already known that typing knowledge is capable of resolving Fitch’s knowability paradox. As I have argued elsewhere, Russellian typing knowledge is immune to the recently raised criticism of the typing approach. The present paper focuses on a special form of the criticism which is based on a revenge form of the paradox; the revenge problem was suggested by Williamson and Hart. The basic idea of the main form of a revenge Fitch’s paradox utilizes quantification over type levels. However, the formalism evoked in the criticism is ambivalent. Suggesting also a proper method of quantification over types, I examine several possible readings of the revenge paradox. As I show in detail, if such readings were viable, they would violate the typing rules in a direct manner. Hence, there is no revenge for the Russellian typing approach to Fitch’s knowability paradox.

Keywords: Fitch’s knowability paradox; revenge; Russellian typing knowledge; ramified hierarchy of types; quantification over types; Russell; Church.

1 Introduction

As is well known, Fitch’s knowability paradox, hereafter FP, was discovered by Church (2009, originally 1945) when reviewing a paper by Fitch who published the paradox in (1963). FP has been solved in a number of different ways. One of them is typing knowledge according to which there is a whole hierarchy of knowledge operators (K-operators). The approach had already been considered by Church (2009), but it was first elaborated by Williamson (2000). Paseau (2008) and Linsky (2009) investigated the approach in greater detail (cf. also Giaretta 2009).

These authors explained the hierarchization of K-operators with an explicit reference to Russell’s ramified theory of types (RTT), cf. Russell (1908, 1910 with Whitehead) or the respected RTT by Church (1976) which has a clearer formulation. Recall that Russell hierarchized his propositions and
so-called propositional functions. Russell’s hierarchy thus significantly differs from the hierarchy of T-predicates and languages later developed by Tarski (1931/1956).

The typing approach has been subjected to a strong criticism, most notably in a paper by Carrara and Fassio (2011). The main objection is that the typing approach is only designed for a solution of the paradox: it lacks an independent justification, thus it is ad hoc. I have rejected such criticism in Author (2015). I reveal mainly the fact that the real target of the criticism is Tarskian typing knowledge, not Russellian typing knowledge. The latter one has an independent justification in formation rules for propositions and propositional operators such as K. The main ideas of the approach will also be briefly introduced in this paper. But my previous paper could not cover a refutation of a special kind of criticism, which is my present aim.

The probable root of such criticism seems to be found in a criticism of Tarski’s hierarchical solution to the Liar paradox (1936/1956) based on a revenge form of the Liar paradox for this approach. This criticism can be derived from the paper by Kripke (1975, p. 697), see Priest (1987/2006, pp. 19–20) for a detailed elaboration. The criticism evokes the predicate “not true in any language (of a given hierarchy of languages)”, formally “λs∀l¬T(s,l)”, where s is a variable for sentences (or rather the names of sentences, which is not important to distinguish here) and l a variable for languages (as sets of expressions). Using the sentence S, which is given as “∀l¬T(S,l)”, one can easily bring the paradox back. In recent theory of paradoxes and philosophical logic, this particular objection to hierarchical solutions to paradoxes is a standard one.1

As I will discuss below, Williamson (2000) first outlined a criticism which in fact consists in a development of a revenge form of FP for the typing approach. A more detailed elaboration was put forward by Hart (2009).

I am going to show that any such revenge form of FP for Russellian typing knowledge suffers from at least one of several defects. Firstly, the criticism invites us to formalize only a seemingly intuitive notion:2

*not known on any type-level (type, order) of knowledge*

with help of the formula:

“∀t¬Kt′p”.

---

1But realize that Tarski himself could defend his position by pointing out that introducing that predicate in a language leads to a paradox, which should be considered as proof that this predicate cannot be introduced to that language. Such a kind of proof had already been used by Tarski in his seminal study in the case of the ordinary T-predicate.  

2Why the “type/order” ambiguity? As I will explain later, “order” is more appropriate, though the recent literature uses rather “type” or “type-level” in these contexts.
From the viewpoint of a type theorist it is not clear, however, what the formula in question means in the first place. Below, I will suggest two main kinds of readings of this ambivalent formula and, consequently, two main kinds of (Hart’s) revenge form of FP.

In each case, I will show that the basic principles of Russellian typing block the argument – contrary to what the critics intended to demonstrate. On some readings, however, the blocking by typing must be supplemented by a new finding concerning knowledge as explicated in the framework of RTT. Not mentioning that I have to expose proper methods to quantify over types/orders in RTT.

The structure of this paper is as follows. The following Section 2 briefly introduces FP, the main ideas of RTT, the Russellian typing rule, and blocking FP by this method. In Section 3, I first explain the difference between two basic kinds of reading of “∀t ¬K_t p”, which depends on the assumed arity of “K_t”. Then, I will investigate the most important examples of monadic and dyadic readings of Hart’s revenge FP which seems to be anticipated by Williamson. Section 4 provides a brief conclusion.

2 Fitch’s knowability paradox and Russellian typing knowledge

2.1 Fitch’s knowability paradox

Fitch’s knowability paradox brings a surprising discovery for the epistemological optimism known as Verificationism which claims that every truth is knowable:

\[(\text{Ver}) \quad \forall p (p \supset \Diamond Kp) \quad \text{Verificationism (Knowability principle)}\]

For, by adding Ver to the obvious principle that not every truth is known:

\[(\text{NonOmn}) \quad \exists p (p \land \neg Kp) \quad \text{Non-omniscience}\]

we derive that every truth is known:

\[(\text{Omn}) \quad \forall p (p \supset Kp) \quad \text{Omniscience}\]

Which is paradoxical.

The derivation utilizes only unproblematic principles of multimodal logic, namely:

\[^3\text{Using familiar notation of multimodal logic, “K” stands for ‘knowing’ (‘it is known that . . . ’) and “\Diamond” (or “□”) stands for ‘possibly’ (or ‘necessarily’); “\Diamond K” stands for ‘it is possible to know’. I use double quotation marks for quotation of expressions, while single quotation marks are used for indication of propositions and other extralinguistic entities, or, sometimes, for indication of a shift in meaning.}\]
(Fact)  \( Kp \vdash p \)  Factivity of Knowledge

(Dist)  \( K(p \land q) \vdash (Kp \land Kq) \)  Distributivity of \( K \) over \( \land \)

(Nec)  if \( \vdash p \), then \( \vdash \square p \)  Necessitation rule

(ER)  \( \square \neg p \vdash \neg \Diamond p \)  interdefinability of modal operators

Here is the derivation:

1.  \( \exists p(p \land \neg Kp) \)  NonOmn
2.  \( (p \land \neg Kp) \)  an instance of 1.
3.  \( \forall p(p \supset \Diamond Kp) \)  adding Ver
4.  \( (p \land \neg Kp) \supset \Diamond (Kp \land \neg Kp) \)  substituting 2. for \( p \) in 3.
5.  \( \Diamond (Kp \land \neg Kp) \)  MP(4.,2.)
6.  \( K(p \land \neg Kp) \)  assumption \textit{per absurdum}
7.  \( (Kp \land \neg Kp) \)  Dist (6.)
8.  \( (Kp \land \neg Kp) \)  Fact (7.)
9.  \( \neg (Kp \land \neg Kp) \)  \textit{reductio} (8. is a contradiction)
10.  \( \neg \Diamond (Kp \land \neg Kp) \)  Nec (9.)
11.  \( \neg \Diamond (Kp \land \neg Kp) \)  ER (10.)

Formula 11. contradicts formula 5. Hence, adding Ver to NonOmn leads to Omn.

2.2 Russellian typing knowledge: forming intensional entities

Russellian explication of the realm of entities has it that besides extensional entities such as individuals, classes of individuals, or truth-functions (as mappings) there exist also \textit{intensional entities}. These do not obey the Principle of Extensionality, because they can be \textit{equivalent} without being \textit{identical}. Examples of intensional entities include (Russellian) structured \textit{propositions} and all the varied operators operating on such propositions, i.e. \textit{intensional operators} as I will call them. Their structure corresponds to the structure of expressions usually chosen for their record; but the abstract intensional entities must not be identified with their records.\textsuperscript{4,5}

Formation of intensional entities is governed by several principles, the best known being the Vicious Circle Principle (VCP). As Russell noted in the famous work written with Whitehead (1910, p. 41), VCP is entailed

\textsuperscript{4}The semantic scheme employed here is thus: sentence – proposition – truth-value; analogously for predicates expressing intensional operators (they denote classes, whereas they express binary ‘propositional functions’).

\textsuperscript{5}The exposition of type matters in Section 2 is partly idiosyncratic, yet it is intended to cover the mainstream opinions adopted in various RTTs.
by a more fundamental principle I call the *Principle of Specification*: no
entity can be fully specified in terms of itself. To illustrate, to fully specify a
function (as mapping) one must determine all its arguments and values,
which would be impossible if the function itself was among its own arguments or
values.

Intensional entities contain variables or objects which are supposed to be
turned into variables. The very formation of such entities requires that they
cannot be in the range of their own variables. Something similar was referred to by
Russell in his formulation of VCP (1908, pp. 225, 237; 1910 with Whitehead, p. 40). I formulate the *Intensional VCP* as follows:

> No entity involving a variable can be itself in the range of this variable. 
> It is thus of, i.e. belongs to, a higher type.

To illustrate, the compound proposition involving a variable for propositions,
schematically \((\ldots p \ldots)\), could not be itself in the range of its variable \(p\). If it were, the variable \(p\) could not be specified, since a variable is given mainly by its range; consequently, \((\ldots p \ldots)\) could not even be specified. Any proposition covering a totality of propositions in the range of \(p\) has to be member only of a superior totality, i.e. being a member of a type of a higher order than the type collecting the propositions in the range of \(p\).

*Type* is thus a collection (a set) of (all) entities of the same kind, types are thus pairwise disjoint. For instance, there is type of individuals or type of unary truth-functions (as mappings). Types which contain common extensional objects will be called *extensional types*. Types which contain intensional objects will be called *intensional types*.\(^7\)

An implementation of the Intensional VCP leads to the division of each intensional type into particular (sub)types called *orders* of that type. For example, the intensional type of propositions splits into the type of first-order propositions, the type of second-order propositions, \ldots, the type of \(n\)-order propositions. By saying that a proposition \(p\) is of order \(k\) (for \(1 \leq k \leq n\)), we mean that \(p\) belongs to the type of \(k\)-order propositions.\(^8\)

The *range* of, say, the propositional variable \(p^k\) is thus always restricted to a particular type, in this case the type of \(k\)-order propositions. This

---

\(^6\)I adopt a widely used notational ambivalence that both simple proposition and propositional variable are written by the same single sign “\(p\)”. In RTT, however, a propositional variable is of a higher order than any proposition which is in its range, but I will ignore it for the sake of simplicity.

\(^7\)“Intensional type” is only an instrumental term of my exposition because no intensional type is really a type.

\(^8\)Assuming cumulativity (see below), the types which are parts of intensional types are not pairwise disjoint. Another remark: it is well known that Russell’s motivation for a ramification of the intensional type of propositions was an intention to solve the Russell-Myhill paradox. Russell then realized fully the principles governing formation of propositions, e.g. VCP. Solution to paradoxes is thus better construed as a by-product of a careful formation of propositions and other intensional entities.
means that neither the higher-order proposition $p^{k+1}$, nor the compound proposition $(\ldots p^k \ldots)$ (which has the same order as $p^k$, viz. $k+1$), is in its range. It has a consequence for the construction of *epistemic propositions*, i.e. propositions constructed with help of the intensional operator $K$, cf. below.

It would be too restrictive if the range of, say, a propositional variable contained not all propositions of all lower orders. For practical purposes we often require that the range of the variable $p^k$ contains not only propositions such as $p^{k-1}$ (for $(k-1) > 2$), but also $p^2$ or $p^1$. This requirement can be incorporated into the *Principle of Cumulativity* formulated by Church (1976, p. 748):

*Every entity of a certain order $k$ is also of order $k+1$.*

By *Churchian RTTs* I mean those RTTs which implement this principle. Clear-cut examples comprise Church’s (1976) and Tichý’s (1988) RTTs. In the case of Russell’s RTT, the implementation of cumulativity is not unambiguous.

In order to resolve all interpretative issues I will utilize a unique RTT developed by Tichý (1988; see its modified definition in Appendix), because it is the only RTT which includes all the entities, which are needed. Since Tichý’s RTT is a part of a rather complex logical framework which is not in the focus of our paper, I will abstract from it to a large extent, using a simpler notation etc. Cf. Appendix where Tichý’s logical system is sketched and our simplification of it are described.

Now I state only the main differences of Tichý’s RTT from the two best known type theories. Both Russell’s and Church’s RTT has only one extensional type, viz. the type of individuals and a lot of intensional types – the type of propositions, the type of monadic propositional functions, etc. Tichý’s RTT, on the other hand, has many extensional types – the type of individuals, the type of functions from individuals to truth-values (i.e. classes of individuals), etc. –, but only one intensional type. This type contains Tichý’s so-called constructions, which are certain algorithmic procedures producing other, usually extensional, objects. Instead of constructions I will still simply speak about propositions and intensional functions.

### 2.3 Russellian typing knowledge: the Rule of Typing of Propositions

Before I formulate the typing rule, I advert to the intuitive validity of the idea of typing knowledge. Consider an ordinary proposition such as ‘Fido is a dog’ which unfolds some fact. It is a proposition of order 1, it belongs to the type of first-order propositions. Such propositions can be known, which is also a fact, but of a slightly different kind. The proposition ‘Xenia
knows that Fido is a dog’ is an epistemic proposition which informs us about somebody’s attitude to the first-order proposition in question and thus it is of order 2. Quite analogously, the epistemic proposition ‘Yannis knows that Xenia knows that Fido is a dog’ is of order 3. Etc., until n. In other words, the order of an epistemic proposition is governed chiefly, though not quite exclusively, by the number of embedded intensional operators K, plus one. Except propositions, the type theorists usually type also (intensional) functions and even variables, etc.

A careful formation of propositions is fixed in formulations of a type theory. As mentioned above, the type theory which is in fact used in this paper is a certain extension of Tichý’s RTT but we will abstract from its idiosyncratic, complex details to meet a greater audience.

It seems that the following Rule of Typing Propositions will be sufficient for our purposes. Let O be any (monadic) intensional operator, e.g. K:

i. The lowest possible order of a (simple) proposition not containing any intensional operator is 1.

Now let \( p^k \) be any proposition of order \( k \), for \( k \geq 1 \).

ii. The lowest possible order of an intensionally compound proposition such as \( O^m p^k \) (for \( m \geq k \), where \( ^{am} \) indicates the order of the argument for \( O \)) is \( m + 1 \).

iii. The lowest possible order of an extensionally compound proposition is identical with the lowest possible order of its subproposition which has the highest order in it.

An example of i.: \( p^1 / 1 \), where “/” abbreviates “has the lowest possible order”. Examples of ii.: \( K^1 p^1 / 2; K^2 K^1 p^1 / 3; K^2 p^1 / 3 \) (note that \( K^2 p^1 \) is possible due to the Principle of Cumulativity). Examples of iii.: \( (p^1 \land q^1) / 1; (p^2 \land q^1) / 2 \). In other words, the type of compound propositions is not indicated, one has to compute it. Expressions “\( K^1 p^2 \)”, “\( K^1 K^2 p^1 \)”, “\( K^2 K^2 p^1 \)”, “\( K^1 (p^2 \land q^1) \)” do not express any propositions.

We may occasionally speak about an order of the operator \( K \), although I did not cover it in the above typing rule for propositions. It would be enough to supplement the above rule in the sense that if one takes, in some context, the order of the proposition \( K^m p^k \) being \( m + l \), for \( l \geq 0 \), then the order of

---

9Do not confuse it with Tarskian typing according to which the sentence – not a proposition – “Fido is a dog” is attached by the subscript numeral “\( 0 \)”, the sentence “Xenia knows that Fido is a dog” by the subscript numeral “\( 1 \)”, etc., thus the predicate “\( K \)” splits into “\( K_0 \)”, “\( K_1 \)”, . . . , and “\( K_n \)”.

10Because of cumulativity, the formulation speaks about the lowest possible order, not only about ‘the’ order. A proposition is intensionally compound if it is compound by means of an intensional operator; it is extensionally compound if it is compound by means of an extensional operator.
$K^m$ is also $m + l$. It holds that the least order of $K^m$ is higher, by 1, than the order of $p^k$ at which $K^m$ is applicable. Analogously for, say, variables; a variable for intensional operators has higher order than any operator in its range.

Concluding this section, in Russellian typing approach the typing of intensional entities is justified by the rules (esp. VCP) which govern their non-circular formation. In particular, the specification of the operator $K^m$ could not be successfully finished, if the range of $K^m$’s applicability would contain any proposition involving $K^m$. Another important feature of the careful formation of intensional entities is that quantification over them is always restricted; this is in fact Russell’s famous doctrine ‘all/any’: one cannot quantify over all propositions there are, but only over propositions of a definite, particular order.

2.4 Blocking the reductio and an invalid rule concerning knowledge

In compliance with the type approach, the above untyped form of FP should be amended in conformity with the above Rule of Typing Propositions. In its consequence, not only are there differences among propositions including $K$, but also differences among rules governing $K$ (which will not, however, be indicated below). To illustrate, one particular rule Dist is Dist$^{2(1)}$–$^{2(2)}$: $K^2(p^1 \land q^2) \vdash (K^2p^1 \land K^2q^2)$.

The crucial part of FP then looks as follows:

\begin{align*}
6^L. & \ K^2(p^1 \land \neg K^1p^1) \quad \text{assumption per absurdum} \\
7^L. & \ K^2p^1 \land K^2\neg K^1p^1 \quad \text{Dist (6$^L$.)} \\
8^L. & \ K^2p^1 \land \neg K^1p^1 \quad \text{Fact (7$^L$.)}
\end{align*}

We thus reveal that the proposition $8^L$. has a content distinct from the originally considered one: $8^L.$ is not a contradiction. Hence, the reductio is blocked.

However, the proposition $8^L.$ would be a contradiction only provided the following rule – which might be perhaps called the Rule of Decrementation of Order of Knowledge – is valid:

$$K^2p^1 \vdash K^1p^1.$$  

The reason for the invalidity of this rule has been largely discussed, cf. Williamson (2000), Linsky (2009), Paseau (2008), Carrara and Fassio (2011). Here I can only repeat the most essential idea of my own explanation (2013), which incorporates elements of the JTB theory. An $m$-order knowledge$^m$ of

\footnote{The limits of applicability of $K^m$ is obvious from its so-called $\eta$-expanded form, $\lambda p^k(K^m p^k)$ – the range of $p^k$ cannot contain any proposition involving $K^m$.}
an $n$-order proposition $p^n$, for $m \geq n$, is defined with help of a justification$^m$ of $p^n$; a justification$^m$ is thus a part of knowledge$^m$ (belief$^m$ is another part). A justification$^m$ of a proposition $p^n$ is defined by means of the existence of at least one $m$-order proposition $q^m$ which is a reason$^m$ for that $p^n$ ($q^m$ can be, say, a step in the proof of $p^n$). This yields that an $m$-order justification$^m$ cannot be an $m-1$-order justification$^{m-1}$, thus one cannot derive $K^{m-1}p^n$ from $K^mp^n$.

To illustrate, Xenia’s knowing$^2$ the first-order proposition ‘Fido is a dog’ can be justified$^2$ by the reason$^2$ that the proposition has been told to Xenia by Yannis who is a reliable informant. Xenia’s knowledge$^2$ of that proposition is thus not comparable with the knowledge$^1$ of that proposition for the reason$^1$ that (say) Fido is a four-legged friend of humans, which is the same, let us admit, as being a dog.

3 A revenge form of Fitch’s knowability paradox

3.1 Williamson’s and Hart’s revenge forms of Fitch’s paradox

As mentioned above, the first version of the revenge FP was previously suggested by Williamson (2000, p. 281). According to him, we are capable to grasp the idea of totally unknown proposition $p$, which amounts to that $p$ is unknown$^i$ for each level $i$ (whereas $p$ may be true); then, one can adapt FP by considering the proposition that $p$ is a totally unknown truth, since that proposition cannot be known$^i$ for any level $i$.

The consideration revolves around a proposition written by Carrara with Fassio by means of the formula "$\forall t \neg K^t p$" (using "$t$" instead of Williamson’s "$i$"; Hart follows Williamson in not using a formal notation). The formula involves quantification over type-levels.

As mentioned in the introduction, I view "$\forall t \neg K^t p$" as an ambivalent expression. The root of this interpretative complication lies in that the proposal conflicts with the fact that the well-known operator $K^m$ is monadic and "$m$" is an integral part of the symbol "$K^m$" – it is no variable at all.$^{12}$

If a variable should occur in the body of a formula as a genuine variable for type-levels, it has to be syntactically clear. After an appropriate adaptation of the notation we get

$$\forall t \neg K^m_2(p^k,t) \text{ (for } m \geq k).$$

\[\text{12} \text{Do not confuse it with my use of a metalanguage when writing, e.g., "} K^k, \text{ for } k=1\text{" in which "} k\text{" serves as a variable; on that occasion, I intend to refer to } K^1, \text{ which is that knowledge operator which applies to a proposition of order 1 (} k=1\text{). In metalanguage, "} K^{k+1}, \text{ or simply "} K\text{", represents the variable } K^{k+1} \text{ the range of which is sufficient for a given consideration.}\]
In this formula, one employs a dyadic (binary) operator which is written
"$K_m^2$" to avoid confusion with the monadic symbol "$K^m$".

In a consequence of such disambiguation, I find several readings of the
symbol "$K_t$". These can be sorted into two groups. On the readings I
call monadic readings, "$K_t$" is yet understood as somehow representing
a monadic operator. Monadic readings will help us to better understand the
failure of the revenge FP on the readings I call ‘dyadic’. On dyadic readings,
"$K_t$" is understood in the manner suggested in the preceding paragraph.
The difference between possible dyadic readings consists mainly in the na-
ture of entities in the range of $t$. In this paper, I will concentrate on the
most probable (dyadic) readings. 13

Since there should be types or orders in the range of $t$, the whole idea of a
revenge FP for typing approach slightly resembles Gödel’s (1944) criticism
of Russell’s RTT. Gödel objected that some formulations of the general
principles of RTT, e.g. ‘No proposition can quantify over all propositions
they are’, violate the very principles. Such criticism seems to suggest that
either Russell’s theory should be type-free, or capable of speaking about
its own principles without violating them. 14 The latter possibility seems to
imply, inter alia, the quantification over types within RTT.

Instead of the modified FP suggested by Williamson, I will examine
the crucial part of the modified FP presented in details by Hart (2009, p.
322). Small differences between the two arguments can be ignored here (the
formula $1^H.$ corresponds to our formula 2. and the formula $2^H.$ corresponds
to our formula 5.; Hart’s Tarskian typing is adapted here to the Russellian
one; “UI” abbreviates “Universal Instantiation”).

\[
\begin{align*}
1^H. & \quad (p \land \forall t \neg K_t^1 p) & \text{assumption of order } t+1 \\
2^H. & \quad \diamond (K_t^{t+1} p \land \forall t \neg K_t^t p) & \text{Ver (} 1^H) \\
3^H. & \quad \diamond (K_t^{t+1} p \land K_t^{t+1} \forall t \neg K_t^t p) & \text{Dist (} 2^H) \\
4^H. & \quad \diamond (K_t^{t+1} p \land \forall t \neg K_t^t p) & \text{Fact (} 3^H) \\
5^H. & \quad \diamond (K_t^{t+1} p \land \neg K_t^{t+1} p) & \text{UI (} 4^H; t+1 \text{ for } t) \\
\end{align*}
\]

The contradiction $(K_t^{t+1} p \land \neg K_t^{t+1} p)$ cannot hold, thus the whole formula
$5^H.$ is a contradiction. Hence, it is necessary, Hart argues, that every truth
is known at some type-level. 15

---

13 I also omit a deeper examination of the revenge FP by Carrara and Fassio (2011, pp.
187–8). They assumed that the ignorance mentioned in $(p \land \neg Kp)$ is to be intended in a
quite unrestricted sense – $p$ is not known in any knowledge state. Then, the order of $K$
in both occurrences in $(Kp \land \neg Kp)$ is the same, from which a contradiction follows and the
paradox is thus not blocked. However, their assumption that $K$ is quite type-theoretically
unrestricted contradicts the rules of RTT: the outer $K$-operator must have a higher order
than the embedded one.

14 An analogous criticism was suggested by Fitch (1946, p. 71, passim).

15 By the Necessitation Rule, one derives $\square \neg(p \land \forall t \neg K_t^t p)$ from the already derived
$\neg(p \land \forall t \neg K_t^t p)$. By applying classical transformation rules one gets $\square (\neg p \lor \forall t \neg K_t^t p)$ and then $\square p \supset \exists t K_t^t p$, which can be generalized as $\square \forall(p \supset \exists t K_t^t p)$.
One can notice that UI applied here is a tool for a certain unwelcome shift. The feeling that something illegitimate takes place here will be confirmed by my analyses.

3.2 Monadic readings

Before I examine the two most relevant dyadic readings, I am going to demonstrate type restrictions on the refutation of a revenge FP which employs the notion

\( \text{not knowing } p \text{ in any way (by any kind of knowing),} \)

The notion captures idea of the quantification over knowing in a natural way: only K-operators, not types/orders, are means of knowledge. For our formalization of the notion thus let \( m \) be a variable ranging (thanks cumulativity) over all K-operators from 1 until the order \( m \):

\[
\forall K^m \neg K^m p^k, \text{ for } m \geq k.
\]

The respective monadic reading of Hart’s argument is thus:

\[
1^V. (p^k \land \forall K^m \neg K^m p^k) \quad \text{assumption of order } m + 1
\]

\[
2^V. \Diamond K^{m+1} (p^k \land \forall K^m \neg K^m p^k) \quad \text{Ver (1}\,^V\,.)
\]

\[
3^V. \Diamond (K^{m+1} p^k \land K^{m+1} \forall K^m \neg K^m p^k) \quad \text{Dist (2}\,^V\,.)
\]

\[
4^V. \Diamond (K^{m+1} p^k \land \forall K^m \neg K^m p^k) \quad \text{Fact (3}\,^V\,.)
\]

\[
5^V. \Diamond (K^{m+1} p^k \land \neg K^{m-i} p^k) \quad \text{UI (4}\,^V\,.; \text{ if } K^m \text{ for } K^m; \text{ for } i \geq 0 \text{ and } (m-i) \geq k)
\]

Obviously, the version of UI which is applied here is not unlimited. One cannot substitute, e.g., \( K^{m+j} \) (for \( j > 0 \)) for \( K^m \), because \( K^{m+j} \) cannot be an admissible value for \( K^m \). The proposition \( 5^V. \) is not a contradiction. The reductio is thus blocked; there is no respective revenge.

The very same conclusion is reached when utilizing Russell’s idea of reducibility (Russell 1908, p. 243; Whitehead, Russell 1910, pp. 59, 74) which says that instead of higher-order operators we can quantify over their lower-order correlates. But the admissible values of the variable \( K^m \) are still operators of an order lower than or equal to \( m \), not higher.

3.3 Dyadic readings

As mentioned above, if one really intends to quantify over the range of \( t \), then “\( K^t p \)” should be read as speaking about a binary relation \( K_2 \) between propositions and objects from the range of \( t \). This substantial difference from monadic readings means that we are urged to consider an entirely
different operator instead of the original familiar monadic operator $K$. The criticism of the typing approach to FP thus changed the subject matter.

One reason why this is inconvenient lies in that the dyadic operator $K_2$ has no natural intuitive correlate. To properly understand the notion $K_2$, we have to define it. Every definition will have approximately the following form:\footnote{I view definitions not as (Russellian) linguistic abbreviations but as deduction rules which link some equivalent intensional functions.}

$$K_2^m(p^k, t) = \exists y \exists K^{m+1}(\sharp K^{m+1}p^k \land (t \ast (K^{m+1}p^k))), \text{ for } m \geq k.$$  

The definition explains a dyadic notion of knowledge in terms of a monadic one. It roughly says that $K_2^m$ applies to the couple $\langle p^k, t \rangle$ if and only if there exists an $m+1$-order $K$-operator such that $K^m p^k$ holds – i.e. $p^k$ is known\footnote{Recapitulation of orders: $p^k / m$ (because $p^k$ is an $m$-order argument for $K^m$; recall that, to simplify type considerations, we do not discriminate between the order of a proposition and a propositional variable having such propositions as values); $K^m / m+1$ (because $K^m$’s order is higher, by 1, than the order of $p^k$), $K^{m+1} / m + 2$ (because $K^{m+1}$’s order is higher, by 1, than the order of its possible value $K^m$). $K_2^m$ is an $m+1$-order object defined with help of an $m+2$-order definiens.} -- whereas $K^m p^k$ is of type/order $t$.

The range of the variable $K^{m+1}$ contains $m+1$-order knowledge operators such as $K^m$ which are applicable on $m$-order propositions such as $p^k$; $K^{m+1} p^k$ thus produces, e.g., $K^m p^k$.\footnote{\sharp is Tichý’s double execution (cf. Appendix), which is a special kind of construction. \sharp increases order, which is suppressed here (instead \sharp we could use another method which does not increase order but which is complicated to be introduced here).} The intensional operator $\sharp$ takes a proposition such as $K^m p^k$ (which is produced by $K^{m+1} p^k$) to the truth-value yielded by the proposition.\footnote{The method suggested below differs from the method used in typed $\lambda$-calculi described in computer science; the differences will not be discussed here.}

We would like to get just that $m$ as the type/order on which the proposition $p^k$ is known. The only problem is that $m$ which we see in the record of the definition is not a variable (cf. 3.1). Fortunately, there are some ways how to attune $m$ and $t$ to transfer, so to speak, the content of $m$ to the genuine variable $t$. This coordination of $m$ and $t$ will be embodied in the selected relation $\ast$ between $K^m p^k$ and $t$; this relation mostly depends on the value of $t$ which will be an explication of type or order.

### 3.4 Types as objects of meta-RTT

RTT can explicitly speak about its own types.\footnote{The method suggested below differs from the method used in typed $\lambda$-calculi described in computer science; the differences will not be discussed here.} At the first sight, types can be best described as universal classes of objects of a given kind. For instance, the type of first-order propositions would be represented by the class $UC^{p^1}=\{p^1_1, p^1_2, \ldots, p^1_n\}$, where $p^1_i$ is a certain first-order proposition.
To say that $p_1^i$ belongs to the type of first-order propositions amounts to saying that it belongs to $UCp_1$.

Since such type-representing universal classes belong to different types, they all cannot be in the range of one and the same variable. No quantification over types would then be possible. This is the reason why I am suggesting another explication of the notion of type. Beside a statement that a certain object is of a particular type, we will be also capable of expressing that a given object is of some type – which amounts to quantification over types.

It is important to point out that, when speaking about certain properties of a particular instance of RTT, our statements belong to that instance of RTT which comments on it, i.e. to a meta-RTT. Note, however, that no instance of RTT can speak about its own types: any meaningful discussion about types presupposes its specification, thus an instance of RTT which speaks about types cannot be defined over these types.

Let $\tau$ be an atomic type of a particular meta-RTT whereas $\tau$ contains logically primitive objects $T_1, T_2, \ldots, T_n$. These objects serve as explicata (or ‘surrogates’) of types of a particular object-RTT.

All such types $T_1, T_2, \ldots, T_n$ without exception are in the range of the variable $t$. The appropriate version of UI thus enables substitution of any $T_i$ for the variable $t_i$. I will write “$t_m$”, “$t_{m+1}$”, etc., instead of “$T_1$”, “$T_2$”, etc., whereas the type $t_{m+1}$ is a type lexicographically following $t$ and represents a type of a higher order (by 1) than $t$.

Here is the definition:

$$K_m^m(p^k, t_m) = \exists \exists K_{m+1}^{m+1} (\sharp K_{m+1}^{m+1} p^k$$

\begin{align*}
&\land (\text{TheTypeOfImmediatelyLowerOrderThan}(t_m) \\
&\in (\text{TheClassOfTypesOfSomethingInSomeContext}^{m+1}(K_{m+1}^{m+1} p^k))))),
\end{align*}

for $m \geq k$.

In its second conjunct, we put the condition that $t_i$ is one of the types the proposition such as $K^m p^k$ has. It is so because cumulativity makes the proposition belonging, dependently on this or that context (which is limited to the order $m + 1$), to more types, whereas these types form a class. The function ‘the-class-of-types-of-something-in-a-context$^{m+1}$’ maps a given proposition to a class of corresponding types. The function ‘the-type-of-immediately-lower-order-than’ is used in the definition because the type of e.g. $K^m p^k$ is in that class of types but we want to derive from this the type at which $p^k$ is known, which is the immediately lower-order type.

Note that the values of the variable $t_i$, i.e. particular types belonging to some object-RTT, have only an indirect correlation with the type of the operator such as K which belongs to meta-RTT. The types of this meta-RTT cannot be something the meta-RTT can speak about. Thus the very possibility of the correlation depends on a link between K of the object-RTT.
and $K_2$ of the meta-RTT, which has to be established in a metameta-RTT.

If we examine the appropriately précised Hart’s revenge FP, we find that $5^T$. is not a contradiction, thus the reductio is blocked:

\[ 1^T. \ (p^k \land \forall t_m \neg K^n_2(p^k, t_m)) \quad \text{assumption of order } m+1 \]
\[ 2^T. \ (p^k \land \forall t_m \neg K^n_2(p^k, t_m), t_{m+1}) \quad \text{Ver (}1^T.\text{)} \]
\[ 3^T. \ (K^{m+1}_2(p^k, t_{m+1}) \land K^{m+1}_2(\forall t_m \neg K^n_2(p^k, t_m), t_{m+1})) \quad \text{Dist (}2^T.\text{)} \]
\[ 4^T. \ (K^{m+1}_2(p^k, t_{m+1}) \land \forall t_m \neg K^n_2(p^k, t_m)) \quad \text{Fact (}3^T.\text{)} \]
\[ 5^T. \ (K^{m+1}_2(p^k, t_{m+1}) \land \neg K^n_2(p^k, t_{m+1})) \quad \text{UI (}4^T.; \ t_{m+1} \text{ for } t_m; \text{ for } l \geq 0) \]

Although this particular version of UI allows us to change $t_m$ to $t_{m+l}$, we are not in any sense forced, or entitled, to the claim amounting to that the proposition $p^k$ which is unknown$^m$ is also unknown$^{m+1}$; cf. also the penultimate section of this paper (3.6).

### 3.5 Orders as numbers

Explicit speaking about types has enabled us to release an unrestricted UI. Now we will add the possibility that the values of $t$ can be incremented, which requires that the values of $t$ are numbers whereas orders can fittingly be identified with (positive) natural numbers. Such understanding of "dt" in “$K^t$" is closest to the ideas of the critics.$^{20}$

Let $\nu$ be an atomic type of RTT which contain natural numbers. Let $\nu$ be the range of the variable $t$. In the following definition, we check whether $t$ is the lowest possible order of a given epistemic proposition and subtract the number 1 in order to get the order at which the embedded proposition is known.$^{21}$

\[
K_2^n(p^k, t) =_{df} \exists K^{m+1}(\sharp K^{m+1}p^k \\
(\sharp (t-1) = \text{TheLowestPossibleOrderOf}^{m+1}(K^{m+1}p^k)), \text{for } m \geq k. \]

Note that the correlation between $ts$ and $ms$ is not accidental: the number which is assigned to the proposition such as $K^mp^k$ is the order of that proposition in that instance of RTT which involves the proposition as well as the number.

Now let us look at the appropriately disambiguated form of the revenge FP:

\[ 1^O. \ (p^k \land \forall t \neg K^n_2(p^k, t)) \quad \text{assumption of order } m+1 \]

---

$^{20}$Cf., e.g., Hart (2009, p. 322).
$^{21}$The function ‘the-lowest-possible-order-of’ is properly definable only in meta-RTT. In its definiens, we would map propositions to their types and these types to their orders, i.e. numbers.
2\textsuperscript{O}. \(\Diamond K_{2}^{m+1}(p^{k} \land \forall t \neg K_{2}^{m}(p^{k}, t), t + 1)\)  \hspace{1cm} \text{Ver (1\textsuperscript{O}.)}

3\textsuperscript{O}. \(\Diamond (K_{2}^{m+1}(p^{k}, t + 1) \land K_{2}^{m+1}(\forall t \neg K_{2}^{m}(p^{k}, t), t + 1))\)  \hspace{1cm} \text{Dist (2\textsuperscript{O}.)}

4\textsuperscript{O}. \(\Diamond (K_{2}^{m+1}(p^{k}, t + 1) \land \forall t \neg K_{2}^{m}(p^{k}, t))\)  \hspace{1cm} \text{Fact (3\textsuperscript{O}.)}

5\textsuperscript{O}. \(\Diamond (K_{2}^{m+1}(p^{k}, t + 1) \land \neg K_{2}^{m}(p^{k}, t + 1))\)  \hspace{1cm} \text{UI (4\textsuperscript{O}.; t + 1 for t)}

(Since \(t\) need not be lower or equal to \(m\), the second conjunct of 5\textsuperscript{O}. is admissible. When \(t + 1\) is greater than the number \(m\), which is the order of the epistemic proposition \(K_{2}^{m}p^{k}\), \(K_{2}^{m}(p^{k}, t + 1)\) is false.)

We can see that although the currently applied version of UI increases \(t\) to \(t + 1\), nothing pushes us to derive that a proposition which is unknown\(^{m}\) is also unknown\(^{m+1}\) (in the sense of replacement of \(K_{2}^{m}(p^{k}, t + 1)\) in favour of \(K_{2}^{m+1}(p^{k}, t + 1)\)). The proposition 5\textsuperscript{O}. is consequently not a contradiction, thus the \textit{reductio} is blocked.

3.6 The invalid Rule of Incrementation of Order of Knowledge

However, is the rule \(\neg K_{2}^{m}(p^{k}, t) \vdash \neg K_{2}^{m+1}(p^{k}, t)\) really invalid as we assume in dyadic readings? (If it were valid, 5\textsuperscript{T}. and 5\textsuperscript{O}. would be contradictions and the \textit{reductio} would not be blocked.)

This question can be reduced to a simpler question concerning the invalidity of the following rule which I call the \textit{Rule of Incrementation of Order of Knowledge}:

\[
K_{m}^{m}p^{k} \vdash K_{m+1}^{m+1}p^{k}.
\]

It can hardly be unnoticed that it is a swapped form of the invalid Rule of Decrementation of Order of Knowledge \(K_{m+1}^{m+1}p^{k} \vdash K_{m}^{m}p^{k}\) which has been discussed at the end of Section 2.

At the first sight, the rule is valid because the Principle of Cumulativity makes the \(m\)th-order proposition \(q^{m}\) being also an \(m+1\)st-order proposition, thus the reason\(^{m}q^{m}\) for knowing\(^{m}p^{k}\) can be the reason\(^{m+1}q^{m+1}\) for knowing\(^{m+1}p^{k}\).

However, the justification\(^{m}\) of \(p^{k}\) is only one component part of its knowing\(^{m}\). A substantial part of knowledge is also a belief in a given proposition. It is thus more than evident why we cannot derive \(K_{m+1}^{m+1}p^{k}\) from \(K_{m}^{m}p^{k}\):

though \(p^{k}\) is known\(^{m}\) and we thus believe\(^{m}\) that \(p^{k}\), there can be nobody believing\(^{m+1}\) that \(p^{k}\); hence, \(p^{k}\) cannot be known\(^{m+1}\) in this case.

For the case of ignorance which is relevant here: ignoring\(^{m}\) \(p^{k}\) amounts to that \(p^{k}\) is true\(^{m}\) and justified\(^{m}\) by a certain reason\(^{m}\), but it is not
believed\(^m\). It follows that \(p^k\) is true\(^{m+1}\) and justified\(^{m+1}\) by a certain \(m+1\)-order reason\(^m\), yet it does not follow that \(p^k\) is unbelieved\(^{m+1}\). Hence, the ignorance of \(p^k\) on the level \(m\) does not entail the ignorance of \(p^k\) on the level \(m+1\).

4 Conclusion

The above investigation leads to the conclusion that the revenge form of FP utilized by Williamson and Hart to challenge (Russellian) typing knowledge and its solution to FP are defective.

The notation evoked in the criticism obscured that the argumentation harbours a serious violation of rules governing Russellian typing knowledge. This ascertainment cannot be surprising: the background idea of the attack – namely that some epistemic propositions of RTT can speak about knowledge on the same type level as they belong to – is quite antagonistic to the core ideas of RTT.

Acknowledgment

[self-identifying text]

Appendix

In early 1970s, Tichý extended Church’s (1940) simple theory of types by allowing also partial polyadic functions. In (1988), he ramified this type theory, which also means that the simple theory is embedded in his RTT (in its definition below see the part (t.1) and (t.n+1)). Except familiar extensional entities (e.g. individuals, various functions as mappings, . . .), Tichý adopted also certain intensional entities he called constructions. These structured abstract procedures can be likened to not necessarily effective algorithmic computations. Each object \(O\) (even a construction) is constructed by infinitely many equivalent, but non-identical constructions. For example, the number seven is constructed by adding three to four or, equivalently, by the square root of forty nine, etc. Each construction \(C\) is specified by i. the object \(O\) it constructs, ii. the way \(C\) constructs \(O\) (by means of which subconstructions).

Constructions are conveniently written by \(\lambda\)-terms. Their semantics has thus two levels: the constructions are direct semantic values of the \(\lambda\)-terms, whereas the objects constructed by the constructions are ‘extensional semantic values’ of the \(\lambda\)-terms. The behaviour of five kinds of constructions can briefly be described as follows.\(^{22}\) Let \(v\) be a valuation, i.e. a field consisting

\(^{22}\)For more details see (Tichý 1988), where the notion of construction is carefully exposed
of sequences which each contains objects of one particular type:

i. where $X$ is a non-construction or construction, the trivialization $^0X$ $v$-constructs the entity $X$ directly, without any change;

ii. the variable $x_k v$-constructs the $k$-th object in the sequence (a part of $v$) of objects of the type the variable ranges over;

iii. where $C, C_1, \ldots, C_m$ are any constructions, the composition $[C C_1 \ldots C_m]$ $v$-constructs the value (if any) of the $m$-ary function (if any) $v$-constructed by $C$ at the $(m$-ary$)$ argument (if any) $v$-constructed by $C_1, \ldots, C_m$;

iv. the closure $\lambda x_1 \ldots x_m C$ $v$-constructs the $m$-ary function from (strings of) values of $x_1, \ldots, x_m$ to the corresponding results of the construction $C$ (a rather simplified formulation);

v. the double execution $^2C v$-constructs the object (if any) $v$-constructed by the construction (if any) $v$-constructed by the construction $C$.

The constructions of kind iii. and v. can be $v$-improper, i.e. $v$-constructing nothing at all.

Here is a bit modified definition of Tichý’s RTT (cf. 1988, p. 66). Let $B$ (base) be a non-empty class of pairwise disjoint collections of basic objects:

(t.1) (First-order types, i.e. types collecting first-order objects)

a) Any member of $B$ is a first-order type over $B$.

b) If $\alpha_1, \ldots, \alpha_m, \beta$ are any first-order types over $B$, then $(\beta \alpha_1 \ldots \alpha_m)$ – the collection of all total and partial $m$-ary functions from $\alpha_1, \ldots, \alpha_m$ to $\beta$ – is a first-order type over $B$.

c) Nothing is a first-order type over $B$ unless it so follows from (t.1) a)–b).

(c.n) (n-order constructions, i.e. constructions of $n$-order objects)

a)–b) Any variable $x v$-constructing an $n$-order object, the trivialization of any $n$-order object $X$, i.e. $^0X$, is an $n$-order construction over $B$.

c)–e) If $x_1, \ldots, x_m, C, C_1, \ldots, C_m$ are any $n$-order constructions over $B$, then $^2C_{(i)}$, $^2C_{(i)} [CC_1 \ldots C_m]$, and $\lambda x_1 \ldots x_m C_{(i)}$ are $n$-order constructions over $B$.

and defended. I omitted here the constructions of the kind of single execution and modified the definition of double execution.

23Following the simplification stated in the above text, the order of $^2C$ is assumed to be identical (not greater by 1) with the order of $^2C$.
f) Nothing is an \( n \)-order construction over \( B \) unless it so follows from (c.n.a)—f).

Now let \( *_n \) be a type of \( n \)-order constructions.

\[ (t.n+1) \quad \text{\((n+1\)-order types\)} \]

a) \( *_n \) and any \( n \)-order type over \( B \) is an \( n+1 \)-order type over \( B \).

b) If \( \alpha_1, \ldots, \alpha_m, \beta \) are \( n \)-order types over \( B \), then \((\beta \alpha_1 \ldots \alpha_m)\) – the collection of all total and partial \( m \)-ary functions from \( \alpha_1, \ldots, \alpha_m \) to \( \beta \) – is an \( n+1 \)-order type over \( B \).

c) Nothing is an \( n+1 \)-order type over \( B \) unless it so follows from \((t.n+1).a)\—b)\).

Tichý utilized the notion of construction as an explication of the notion of (structured) meaning. The denotata of expressions are explicated as the objects constructed by the constructions which are meanings of the expressions (note the two-level semantics). If the reference of an expression varies across the logical space and time-scale, its denotation is not an object \( O \) but a PWS-intension (“PWS” stands for “possible world semantics”), i.e. a function from possible worlds (\( W_s \)) and moments of times (\( T_s \)) to objects of the same type as \( O \). For example, the denotation of an ordinary sentence such as “Fido is a dog” is not simply a truth-value but a PWS-proposition, i.e. a PWS-intension having truth-values as its values in \( W_s \) and \( T_s \).

The type basis for such approach to meaning comprises individuals (collected in the type \( \iota \)), (two) truth-values (the type \( o \)), possible worlds (the type \( \omega \)) and real numbers (the type \( \tau \)) representing moments of time. (Above in this paper, I modify the basis by utilizing \( \nu \), collecting natural numbers, instead of \( \tau \) and by adding \( \tau \) collecting explicata of types of an object-RTT.) The meaning of a common sentence such as “Fido is a dog” is the construction \( \lambda w \lambda t [\overline{0} \text{Dog}_w t 0 \text{Fido}] \), where \( 0 \text{Fido}/\iota; \ 0 \text{Dog}/(((o\iota)\tau)\omega) \) (a property of individuals); \( w/\omega; \ t/\tau; \ C_{ut} \) abbreviates \([C w t]\).

Tichý’s (1988) framework offers an elegant solution to the puzzles of hyperintensionality. Propositional attitudes, for example, are explicated as attitudes towards constructions of PWS-propositions, not towards mere PWS-propositions. Because of this and identity criteria for constructions, we are not erroneously allowed – as in intensional logic using PWS-semantics – to infer from an agent’s believing that \( 1 + 1 = 2 \) that the agent believes any other mathematical truth, e.g. Fermat’s Last Theorem. Proper modelling of such attitudes is enabled in Tichý’s system by adopting constructions of higher orders. The sentence “Xenia knows that Fido is a dog” expresses the construction \( \lambda w \lambda t [\overline{0} \text{Knows}^k_{ut} 0 \text{Xenia} 0[\lambda w \lambda t [\overline{0} \text{Dog}_w t 0 \text{Fido}]]) \), where \( 0 \text{Xenia}/\iota; \ 0 \text{Knows}^k/(((o\iota)\tau)\omega) \) (a relation between individuals and \( k \)-order constructions; we do not know whether \( k=1 \)). Here \( 0[\lambda w \lambda t [\overline{0} \text{Dog}_w t \]
Fido] constructs directly the $k$-order construction $\lambda w \lambda t [^0 \text{Dog}_{^0 \text{Fido}}]$ which is the object of Xenia’s knowledge.

In this paper, we abstract from many details of this logical framework. Instead of constructions we speak about intensional operators and propositions. Formal representation of PWS-intensionality is entirely omitted, our particular semantic schemas being thus: sentence – proposition – truth-value, $m$-ary relational predicate – $m$-ary intensional operator – class of $m$-tuples, etc. Signs of trivialization, “$\ominus m$”, are omitted; the sign of double execution, “$\ominus^2 m$”, is replaced by “$\ominus^2$”. Prefix notation and square brackets are replaced by more familiar notational conventions and devices. As for as writing quantifiers (which are in fact subclasses), “$\exists \alpha[x ... x ...]$” abbreviates “$\exists^\alpha \lambda x[x ... x ...]$” (where $\alpha$ is the type ranged over by $x$), “$\diamond [...]” abbreviates “$\exists^\alpha \lambda w \exists^\beta \lambda t[w ... w ... t ... ...]$.” To simplify considerations concerning orders, the order of variables for $k$-order constructions (e.g. $p^k$) and the order of double executions of $k$-order constructions (e.g. $\ominus p^k$) is considered to be $k$, though it is in fact $k+1$.

References


