Noname manuscript No.
(will be inserted by the editor)

The Placeholder View of Assumptions and the Curry-Howard Correspondence

Ivo Pezlar

Received: date / Accepted: date

Abstract Proofs from assumptions are amongst the most fundamental reasoning techniques. Yet the precise nature of assumptions is still an open topic. One of the most prominent conceptions is the placeholder view of assumptions generally associated with natural deduction for intuitionistic propositional logic. It views assumptions essentially as holes in proofs, either to be filled with closed proofs of the corresponding propositions via substitution or withdrawn as a side effect of some rule, thus in effect making them an auxiliary notion subservient to proper propositions. The Curry-Howard correspondence is typically viewed as a formal counterpart of this conception. I will argue against this position and show that even though the Curry-Howard correspondence typically accommodates the placeholder view of assumptions, it is rather a matter of choice, not a necessity, and that another more assumption-friendly view can be adopted.

Keywords placeholder view of assumptions · assumption withdrawing · Curry-Howard correspondence · natural deduction · intuitionistic propositional logic

1 Introduction

Proofs from assumptions are amongst the most fundamental reasoning techniques. Yet the precise nature of assumptions is still an open topic. One of the most prominent conceptions is the placeholder view of assumptions generally associated with natural deduction for intuitionistic propositional logic. It views assumptions essentially as holes in proofs (either to be filled with closed proofs of the corresponding propositions via substitution or withdrawn as a side effect of some rule), thus in effect making

Work on this paper was supported by grant nr. 19-12420S from the Czech Science Foundation, GA ČR.

Ivo Pezlar
Czech Academy of Sciences, Institute of Philosophy, Jilská 1, Prague 110 00, Czech Republic
Tel.: +420 221 183 348
E-mail: pezlar@flu.cas.cz
ORCID: 0000-0003-1965-2159

them an auxiliary notion subservient to proper propositions. The Curry-Howard correspondence is typically viewed as a formal counterpart of this conception (recently, see, e.g., Schroeder-Heister (2016)). I will argue against this position and show that even though the Curry-Howard correspondence typically accommodates the placeholder view of assumptions, it is rather a matter of choice, not a necessity, and that another more assumption-friendly view can be adopted.

This paper is structured as follows: In the first section (Section 1.1), I will briefly introduce assumption withdrawing rules, specifically the implication introduction rule and examine its Curry-Howard correspondence interpretation. In the second section (Section 1.2) I will present Schroeder-Heister's comments about the placeholder view of assumptions and the Curry-Howard correspondence. In the final section (Section 1.3), I will introduce and formalize an alternative view of assumptions that treats them not as placeholders but rather as domains of functions that capture derivations from assumptions.

1.1 Assumption withdrawing

The rule for implication introduction from natural deduction for intuitionistic propositional logic is arguably the best-known example of the assumption withdrawing rule:[1]

$$\dfrac{\begin{matrix}[A] \\ \vdots \\ B\end{matrix}}{A \supset B}$$

It prescribes the following inference step: if we can derive B from assumption A, then we can derive $A \supset B$ and withdraw the initial assumption A (it is worth noting that other assumptions than A may be used in deriving B and those remain open after discharging A). Note that this rule effectively embodies the deduction theorem from standard axiomatic systems. In other words, the implication introduction rules is internalizing structural information from the proof level ("B is derivable from A") to the propositional level ("A implies B").[2]

The problematic aspect of this and other assumption withdrawing rules stems from the fact that it behaves differently from the non-assumption withdrawing rules. More specifically, with implication introduction rule we are deriving the proposition $A \supset B$ not from other propositions as with other standard rules (e.g., conjunction introduction), but from a hypothetical proof. To put it differently, the inference step

[1] Assumption (hypothesis, supposition) withdrawing rules first appeared in the works of Gentzen (1935) and Jaśkowski (1934). It is important to emphasize that we are not really interested in the implication introduction rule itself only insofar as it is an assumption withdrawing rule. Other assumption withdrawing rules from intuitionistic propositional logic such as, e.g., disjunction elimination rule or negation introduction rule, would be suitable for our analysis as well. I only choose to start with the implication introduction rule for its familiarity. I will return to this issue later in section 2.

[2] Schroeder-Heister (2016) describes this as a two-layer system. Note that, strictly speaking, the assumptions are not really withdrawn, they are rather incorporated into the propositional level in the form of an antecedent.

validated by the implication introduction takes us from a *derivation* starting with a hypothesis to a proposition, not just from propositions to another proposition as do rules without assumptions.[3]

For example, consider the following simple proof of the theorem $A \supset ((A \supset B) \supset B)$ of propositional logic:

$$\dfrac{\dfrac{\dfrac{[A \supset B]^1 \qquad [A]^2}{B}\supset\mathrm{E}}{(A \supset B) \supset B}\supset\mathrm{I}_1}{A \supset ((A \supset B) \supset B)}\supset\mathrm{I}_2$$

We start by making two assumptions $A \supset B$ and A. Applying the implication elimination rule (modus ponens) we derive B. What follows are two consecutive applications of implication introduction rule, first withdrawing the assumption $A \supset B$, the second withdrawing the assumption B. Note that it is the fact that B is derivable from $A \supset B$ together with A that warrants the application of the implication introduction rule and the derivation of the corresponding proposition $(A \supset B) \supset B$, at that moment still depending on the assumption A. Analogously with the second application of the implication introduction rule that withdraws this remaining assumption.

A proof that relies on no assumptions is called a closed proof. If a proof depends on some assumptions that are yet to be withdrawn (i.e., open/active assumptions) it is called an open proof. For example, our derivation of $A \supset ((A \supset B) \supset B)$ constitutes a closed proof, since both assumption were withdrawn in the course of the derivation. Assuming we would not carried out the last inference step, we would get an open proof:

$$\dfrac{\dfrac{[A \supset B]^1 \qquad A}{B}\supset\mathrm{E}}{(A \supset B) \supset B}\supset\mathrm{I}_1$$

since the assumption A, upon which the derivation $((A \supset B) \supset B)$ depends, is still active.

Closed proofs are usually preferred to open ones for the simple reason that closed proofs are generally viewed as the fundamental notion in standard proof-theoretic systems. From this perspective, assumptions are just temporary holes in the proof that are preventing us from reaching a closed proof. These open holes can be are either completely discarded via assumption withdrawing rules or filled in with other already closed proofs via substitution. This is the reason why Schroeder-Heister (2016) and others[4] call this the placeholder view of assumptions: active assumptions are just auxiliary artefacts of the employed proof system that behave differently than proper propositions, i.e., propositions that do not appear as assumptions.

3 This non-standard behaviour is also the reason why Prawitz (1965) describes assumption withdrawing rules as improper rules and introduces the distinction between inference rules and deductions rules. For more, see Prawitz (1965), Pezlar (2014).

4 See, e.g., Francez (2015), Oliveira (2019).

Sidenote. Although the adoption of assumption withdrawing rules is the current "industry standard", by no means is it the only option. Most notably, Frege was strongly against the whole idea of reasoning from hypotheses (see, e.g., Frege (1991), p. 335), similarly also Tichý (1988). Schroeder-Heister (2016) (p. 256) calls it the no-assumptions view.[5] Alternatives usually lie in either supplying non-proof-theoretic explanations of implication connective or by adopting sequent-style proof calculus, where assumption withdrawing does not occur (or at least not in the same sense as with natural deduction).

1.1.1 The Curry-Howard correspondence

The placeholder view of assumptions is also supported to a large extent by the Curry-Howard correspondence in its basic form which links typed lambda calculus and implicational fragment of intuitionistic propositional logic.[6] Under this correspondence, natural deduction assumptions correspond to free variables of lambda calculus, which fits well with the interpretation of assumptions as open holes in the proof.

For example, assuming only the implicational fragment of intuitionistic propositional natural deduction, we get the following correspondences between the propositional and functional dimensions of the Curry-Howard correspondence:

NATURAL DEDUCTION	LAMBDA CALCULUS
assumption	free variable
implication introduction	function abstraction
implication elimination	function application

Under this correspondence, the implication introduction rule will then look as follows:

$$\dfrac{\begin{matrix}[x:A]\\ \vdots \\ b(x):B\end{matrix}}{\lambda x.b(x):A\supset B}$$

Note that the act of withdrawing the assumption A corresponds to λ-binding of the free variable x. The whole proof of the theorem $A \supset ((A \supset B) \supset B)$ would then proceed in the following way:

$$\dfrac{\dfrac{\dfrac{[x:A\supset B]^1 \qquad [y:A]^2}{xy:B}\supset\mathrm{E}}{\lambda x.xy:(A\supset B)\supset B}\supset\mathrm{I}_1}{\lambda y.\lambda x.xy:A\supset((A\supset B)\supset B)}\supset\mathrm{I}_2$$

[5] However, it is worth noting that even with Frege the situation is not straighforward, as one of the reviewers pointed out. Recently, Schroeder-Heister (2016) (p. 257, footnote) noted that there is a "hidden two-layer system" of assumptions and assertions in the background of Frege's system as well. See also Schroeder-Heister (2014).

[6] See, e.g., Sørensen and Urzyczyn (2006).

with the concluding proof object (closed term) $\lambda y.\lambda x.xy$ with no free variables representing the final closed proof with no active assumptions. In contrast, the open proof discussed earlier:

$$\dfrac{\dfrac{[x : A \supset B]^1 \qquad y : A}{xy : B}\supset\text{E}}{\lambda x.xy : (A \supset B) \supset B}\supset\text{I}_1$$

concludes with the proof object $\lambda x.xy$ that still contains the free variable y corresponding to the yet to be withdrawn assumption A.

1.2 The placeholder view of assumptions

The Curry-Howard correspondence is generally viewed as incorporating the placeholder view of assumptions. Probably most recently, this point was explicitly made in Schroeder-Heister (2016):

> A formal counterpart of [the placeholder view of assumptions] is the Curry-Howard correspondence, in which open assumptions are represented by free term variables, corresponding to the function of variables to indicate open places. (Schroeder-Heister (2016), p. 255)

and in Schroeder-Heister (2012) he adds:

> The lambda calculus view and the corresponding Curry-Howard-interpretation actually incorporates the placeholder view of assumptions by always using a variable to represent the ground for an assumption, which by means of substitution can be filled with a term standing for a closed proof of it. (Schroeder-Heister (2012), p. 939)

Furthermore, in the same paper Schroeder-Heister advocates for a more general concept of inference that takes us not from propositions to other propositions, but from (inferential) consequence statements $A \models B$ to other consequence statements in order to – amongst other things – equalize the status of assumptions and assertions. The general form of inference rules is the following:[7]

$$\dfrac{A_1 \models B_1 \quad \dots \quad A_n \models B_n}{C \models D}$$

where the antecedents can be empty. As he explains:

> This corresponds to the idea that in natural deduction, derivations can depend on assumptions. Here this dependency is expressed by non-empty antecedents, as is the procedure of the sequent calculus. Our model of inference is the sequent-calculus model... (Schroeder-Heister (2012), p. 938)

[7] As one of the reviewers correctly noted, this general form of inference rules can be achieved in standard natural deduction with sequent-style (also known as logistic) presentation as well. I agree, however, it would still rely on the same Curry-Howard correspondence and thus kept viewing assumptions as placeholders, which I wish to avoid. In other words, I want to move from $x : A \vdash b(x) : B$, where x is a ground for A and $b(x)$ is a ground for B, towards $f : A \vdash B$, where f is a ground for the whole derivation from A to B. For more, see below.

To show that this rule is correct, we have demonstrate that given the grounds for the premises (denoted as $g : A \models B$) we can construct grounds for the conclusion. In other words, the grounds of the conclusion have to contain some operation f transforming the grounds for the premises to the grounds for the conclusion. Schematically:

$$\frac{g_1 : A_1 \models B_1 \quad \ldots \quad g_n : A_n \models B_n}{f(g_1, \ldots g_n) : C \models D}$$

Schroeder-Heister comments on this rule as follows:

> . . . [H]andling of grounds in the sense described is different from that of terms in the typed lambda calculus. When generating grounds from grounds according to [the rule immediately above], we consider grounds for whole sequents, whereas in the typed lambda calculus terms representing such grounds are handled within sequents. So the notation $g : A \models B$ we used above, which is understood as $g : (A \models B)$, differs from the lambda calculus notation $x : A \vdash t : B$, where t represents a proof of B from A and the declaration $x : A$ on the left side represents the assumption A. (Schroeder-Heister (2012), p. 939)

However, it should be mentioned that he left it "open how to formalize grounds and their handling." (ibid., p. 938)

I will argue that even though lambda calculus with the Curry-Howard interpretation can be seen as embodying the placeholder view of assumptions in the intuitionistic propositional logic, within the family of Curry-Howard correspondence based systems we can consider a generalized approach that is free of this view. In other words, I will argue that the Curry-Howard correspondence is not necessarily the formal counterpart of the placeholder view of assumptions, which is generally associated with natural deduction for intuitionistic propositional logic. Furthermore, I will also argue that it can be used for formalizing and handling grounds for the whole consequence statements as discussed in the quote above.

1.3 Function-based approach to assumptions

Let us return to the implication introduction rule:

$$\frac{\begin{matrix}[x : A] \\ \vdots \\ b(x) : B\end{matrix}}{\lambda x.b(x) : A \supset B}$$

Adopting the sequent-style notation for natural deduction,[8] we can rewrite this rule as follows:

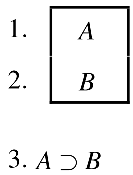
$$\frac{x : A \vdash b(x) : B}{\vdash \lambda x.b(x) : A \supset B}$$

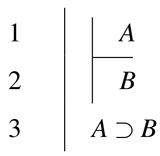
[8] See, e.g., Gentzen's system NLK, discussed in von Plato (2012).

where the symbol $\vdash$ is used to separate assumptions from (derived) propositions.

First, note that the premise $x : A \vdash b(x) : B$ for this rule is a hypothetical judgment (or consequence statement to use Schroeder-Heister's terminology) claiming that we can derive B under the assumption A. In other words, the premise is a derivability statement, which immediately addresses the problematic aspect of implication introduction rule discussed at the beginning: no longer are we deriving a proposition from a derivation but rather one derivability statement from another.

Furthermore, note that the assumption withdrawing is still present, yet it would be misleading to keep viewing assumptions as placeholders or holes, since now they are constitutive parts of the derivability statement. In other words, assumptions are now best regarded not as placeholders, but rather as a context in which the asserted propositions were made. Yet, even though assumptions are no longer just placeholders, they are still not on equal terms with proper asserted propositions positioned on the right hand side of the symbol $\vdash$: assumptions are still represented via variables and our main goal is to eliminate them from the proof, i.e., to empty the context of the asserted proposition. In other words, even though the sequent-style presentation of natural deduction rules is free of some of the criticism from earlier, it still relies on the same Curry-Howard correspondence with assumptions represented via free variables. Thus. the placeholder view of assumptions is still present, it is just less apparent. For the same reason I also omit discussion of linear notations such as Jaśkowski's box-style notation (see Jaśkowski (1934)) which inspired the later Fitch-style notation (see Fitch (1952)). For example, if we compare the structure of their implication introduction rules, we can see that they still operate in the same general way by taking a hypothetical derivation of B from A as a premise:[9]





Now, let us return to the sequent-style notation and take a closer look at the conclusion of the rule:

$$\frac{x : A \vdash b(x) : B}{\vdash \lambda x.b(x) : A \supset B}$$

Specifically, notice that the derivation of B from A is coded with an abstraction term from lambda calculus, which means it captures some sort of function. Reasoning backwards, this should mean that between the assumption (context) and the conclusion (asserted proposition) has to be a relationship that can be understood functionally, otherwise, we would have nothing to code via lambda terms. To put it differently, there has to be some more fundamental notion of function at play that we are coding through the concrete abstraction term.

We can try to capture this observation via the following rule:

[9] The other reason is that formulating the Curry-Howard correspondence for these systems is more complicated and much less explored (see, e.g., Geuvers and Nederpelt (2004)).

$$\frac{x : A \vdash b(x) : B}{f : A \Rightarrow B}$$

where f is to be understood as exemplifying the more fundamental notion of a function that takes us from A to B.[10]

Note that this rule can be roughly understood as the opposite of the implication introduction rule that goes in the other direction: while the implication introduction rule makes the derivation captured via the derivability statement in the premise more concrete in the form of implication proposition and the corresponding lambda term, this rule makes the derivation more general in the sense that it is now considered as a function from A to B. Also notice that assumptions are no longer placeholders or contexts, but types of arguments for the function f capturing the corresponding derivation.[11] In other words, assumptions now stand equal to proper propositions, they are not just an auxiliary notion captured via free variables.

Furthermore, capturing derivations in this way allows us to consider grounds for the whole consequence statements as Schroeder-Heister required, not just grounds for the conclusions under some assumptions. More specifically, treating consequence statement $A \models B$ as a function type $A \Rightarrow B$ (in accord with the Curry-Howard correspondence) and a ground g as an object f of this type, we can reformulate the general rule as follows:

$$\frac{g_1 : A_1 \Rightarrow B_1 \quad \ldots \quad g_n : A_n \Rightarrow B_n}{f(g_1, \ldots g_n) : C \Rightarrow D}$$

As a more concrete example, consider, e.g., the rule for derivation composition.[12] In Schroeder-Heister's notation we get the rule:

$$\frac{g_1 : A \models B \qquad g_2 : B \models C}{f(g_1, g_2) : A \models C}$$

stating that if we have grounds for derivation of B from A and for C from B, then we can apply some function/operation f to these grounds and obtain grounds for the derivation of C from A.

With our approach based on functions we can properly explicate this process by defining what operation has to be used to transform (in a constructive manner) grounds for premises into the grounds for the conclusion. The key observation is that

---

[10] I will talk more about the differences between $\lambda x.b(x) : A \supset B$ and $f : A \Rightarrow B$ later in section 1.3.1. For now, it suffices to say that $\lambda x.b(x)$ is a function in the sense that it is an element of some cartesian product type, while f is a function in a more primitive sense as an unsaturated entity awaiting arguments (see, e.g., Nordström et al (1990), p. 49). Furthermore, the move from $A \vdash B$ towards $A \Rightarrow B$ should not be conflated with the move from a n-level turnstile $A \vdash^n B$ to a $n+1$-level turnstile $\vdash^{n+1}$ $(A \vdash^n B)$ considered, e.g., by Došen (1980) (I thank an anonymous reviewer for bringing this to my attention.) Conceptually, the main difference is that Došen's iterating turnstiles grow infinitely "upwards", while the idea behind $A \Rightarrow B$ is rather that we are going "downwards" to the more fundamental notion of function behind $A \vdash B$.

[11] To borrow terminology from category theory, assumptions are now source objects of the whole derivation/function, with proper propositions being the target objects.

[12] As one of the reviewers correctly pointed out, derivation composition is, strictly speaking, not a basic rule, but a property of natural deduction derivations. Thus the above rule should be rather viewed as an optional rule justified by the corresponding theorem proving the desired property (see, e.g., Theorem 8.1.4 in Negri et al (2001), p. 171.)

since derivations are understood as functions, their composition becomes essentially just the composition of functions. The rule we obtain will be as follows:

$$\frac{g_1 : A \Rightarrow B \qquad g_2 : B \Rightarrow C}{(g_2 \circ g_1) : A \Rightarrow C}$$

where $(g_2 \circ g_1)(x) : C$ is defined as $g_2(g_1(x)) : C$ within the context $x : A$.

Finally, it is worth reiterating that antecedents of the consequence statements can be empty. In practice this corresponds to premises that are not hypothetical judgments. For example, while $g : A \Rightarrow B$ denotes a premise where B depends on some assumption A and f is a function, $a : \ \Rightarrow A$ signifies a premise where A does not depend on any assumption and a is not a function.[13] If we write $a : \ \Rightarrow A$ simply as $a : A$, non-assumption withdrawing rules will closely coincide with the standard rules. For example, the conjunction introduction rule could be captured as $\frac{a : A \qquad b : B}{f(a,b) : A \wedge B}$ where f is a pairing function.

1.3.1 Formalization

So far, I have treated $f : A \Rightarrow B$ quite informally to mean "f is a function from A to B". In this section, I will provide more specific explication of this kind of statement utilizing Martin-Löf's constructive type theory, in both its lower- (see Martin-Löf (1984)) and higher-order presentations (see Nordström et al (1990), Nordström et al (2001)).[14]

Let us start by retracing our steps that led us to the introduction of $f : A \Rightarrow B$. We began with implication introduction rule. In constructive type theory, implication is defined using the dependent function type called Π type, which is cartesian product of a family of sets.[15] It has the following introduction rule:[16]

$$\frac{x : A \vdash b(x) : B(x)}{\lambda x.b(x) : (\Pi x : A)B(x)}$$

Now let us assume a scenario where x is not free in B, in other words where B does not depend on A. Thus we end up with an introduction rule for a non-dependent function type:

$$\frac{x : A \vdash b(x) : B}{\lambda x.b(x) : (\Pi x : A)B}$$

[13] We are interested in getting rid of the placeholder view of assumptions by capturing assumptions as arguments of functions which represent the corresponding derivations from these assumptions. That does not mean that every derivation has to be explainable via functions (of course, it can be done if we assume non-emptiness of all antecedents).

[14] There are, of course, other possible ways to formalize it. If we weren't limited to the Curry-Howard correspondence based systems, the most straightforward choice would probably be category theory, or more precisely categorial proof theory (see Došen (2016), Došen and Petrić (2004)), where $f : A \Rightarrow B$ would be interpreted simply as a morphism f from A to B.

[15] Sets are to be understood in Martin-Löf's constructive sense.

[16] To simplify presentation, I omit the corresponding introduction rule for equal elements of this type.

where $(\Pi x : A)B$ becomes the standard function space between A and B, more commonly written as $A \to B$. Since constructive type theory is built upon the Curry-Howard correspondence, we can also view A and B as propositions. Now applying the Brouwer-Heyting-Kolmogorov proof interpretation of logical constants, specifically of implication as a function that takes a proof of A and transforms it into a proof of B, we can see that $A \to B$ can be also interpreted as the implication $A \supset B$.

Now, once we have implication defined, let us have a look at the premise for the corresponding introduction rule:

$$x : A \vdash b(x) : B$$

This is an example of a hypothetical judgment of constructive type theory and it is easy to see that it mirrors derivability statements discussed earlier. It tells us that we know $b(a)$ to be a proof of the proposition B assuming we know a to be a proof of the proposition A and furthermore that $b(a)$ and $b(c)$ are equal proofs of B whenever a and c are equal proofs of A. To put it differently, the hypothetical judgment $x : A \vdash b(x) : B$ can be seen as stating that $b(x)$ is a function with domain A and range B.[17]

This fact, however, cannot be stated directly in the lower-order presentation of constructive type theory we have been using so far. Thus we move towards the higher-order presentation, which can be understood as a generalization of the lower-order presentation using a more primitive notion of type. Amongst other things, higher-order variant allows us to, e.g., capture whole rules of lower-order presentation via judgments, explicate meanings of constants such as Π or λ, express the Curry-Howard correspondence as a definition in the system itself, and most importantly for our present purpose here, form a higher-order notion of function which can be used to capture the function hidden behind the hypothetical judgment $x : A \vdash b(x) : B$ discussed above.

Higher-order (dependent) function types are formed by the following rule:

$$\frac{\alpha : type \qquad x : \alpha \vdash \beta : type}{(x : \alpha)\beta : type}$$

where $\alpha : type$ is a higher-order judgment declaring that α is a type. If β does not depend on α, we will write $(x : \alpha)\beta$ as $(\alpha)\beta$. For example, propositional negation $\neg$ can understood as a function of type $(prop)prop$, where $prop$ is the type of propositions.

Objects of type $(x : \alpha)\beta$ are functions that can be introduced via the rule of abstraction:

$$\frac{x : \alpha \vdash b : \beta}{(x)b : (x : \alpha)\beta}$$

where the prefix brackets ( ) indicate the abstraction: all free occurrences of x in b become bound in $(x)b$. Note that this rules coincides with our preliminary rule for introducing functions presented earlier.

Now assume we have a derivation of proposition B from proposition A, i.e., $x : A \vdash b : B$. Using the above rule, we can now derive the following higher-order judgment:

$$(x)b : (A)B$$

[17] See Martin-Löf (1984).

which captures the corresponding derivation. It is easy to see that $(x)b : (A)B$ can be used to interpret our statement $f : A \Rightarrow B$, as was required. In other words, $(x)b : (A)B$ can be understood as a higher-order judgment declaring that we have (potentially open) derivation of B from A coded via the function $(x)b$.

For example, the rule for derivation composition can be now formalized as:

$$\frac{f : (A)B \qquad g : (B)C}{(g \circ f) : (A)C}$$

where the composition operator $\circ$ is to be understood as a function of type $((A)B)((B)C)(A)C$ assuming $A : prop$, $B : prop$, and $C : prop$.

Analogously, the functional variant of implication introduction rule would be:

$$\frac{f : (A)B}{\lambda(f) : A \supset B}$$

where λ is now a higher-order function (i.e., a function taking as arguments other functions) applied to f, not just a variable binding symbol.

It is important to emphasize that function type $(A)B$ cannot be conflated with function type/set $A \supset B$, the most basic reason is that they are inhabited by different objects: the former by functions, the latter by elements specified by Π-introduction rule, i.e., objects of the form $\lambda x.b(x)$ (or $\lambda(f)$ in its higher-order presentation) that are used to code functions, similarly as are sets of ordered pairs used to code functions in set theory. More generally, the notion of function behind the type $A \supset B$ is parasitic on a more fundamental notion of function behind the type $(A)B$.[18] From the logical point of view, the main reason we should avoid merging $(A)B$ and $A \supset B$ is that A in $(A)B$ is an assumption of derivation, while A in $A \supset B$ is an antecedent of implication, hence they are objects of different levels. This is perhaps best illustrated by the fact that assuming some function f of type $(A)B$ essentially corresponds to assuming the rule $\frac{\mathrm{A}}{\mathrm{B}}$ in Schroeder-Heister's natural deduction with higher-level rules (Schroeder-Heister (1984)), where even rules can act as assumptions to be discharged.[19]

Another way to view the difference between these two notions of functions is to consider their associated notions of function application. Typically, abstraction terms $\lambda x.b(x)$ are applied to their argument terms via some application function **ap** (which is often left implicit), thus we obtain $\mathbf{ap}(\lambda x.b(x), a)$. However, it is easy to see that $\mathbf{ap}(\lambda x.b(x), a)$ itself relies on a more primitive notion of application of **ap** to $\lambda x.b(x)$ and a. And it is this more fundamental notion of application that is associated with functions of type $(A)B$:[20]

$$\frac{f : (A)B \qquad a : A}{f(a) : B}$$

[18] I thank Ansten Klev for making me aware of this issue. See also Klev (2019a), Klev (2019b).

[19] I would like to thank one of the reviewers for this remark.

[20] See, e.g., Nordström et al (1990), p. 143, Klev (2019a), pp. 286-287

Furthermore, note that on the functional approach the priority of categorical vs. hypothetical[21] is reversed: the notion of closed proof becomes conceptually secondary to the notion of hypothetical proof (derivation from assumption(s)), because the notion of function behind objects inhabiting $A \supset B$ is derived from the more fundamental notion of the function behind the objects inhabiting $(A)B$.

2 Beyond implication

So far, we have been discussing the alternative to the placeholder view of assumption only in the implicational fragment of intuitionistic propositional logic. But what has been said about the implication introduction rule – which is arguably naturally related to functions via the Curry-Howard correspondence – I believe applies to other assumption withdrawing rules as well. Recall that we were not really interested in the rule itself, but in the nature of its hypothetical premise which seems to be shared across other assumption discharging rules not only from intuitionistic propositional logic but classical propositional logic as well.

To show this I provide two more analyses of assumption withdrawing rules, namely the disjunction elimination rule and reductio ad absurdum rule. The disjunction elimination rule is chosen because it also a rule from intuitionistic propositional logic, but it is an elimination rule and when its assumptions are discharged, they "disappear" from the conclusion. This is in contrast to implication introduction rule where the withdrawn assumption leaves a trace in the form of the antecedent of the inferred conditional. The classical reductio ad absurdum rule also does not leave any trace of the withdrawn assumptions, but more importantly it represents a non-intuitionistic rule. More specifically, if we add it to the standard rules for intuitionistic propositional logic, we can obtain a natural deduction system for classical propositional logic.[22]

Let us start with the disjunction elimination rule:

$$\dfrac{A \vee B \qquad \begin{array}{c}[A]\\ \vdots \\ C\end{array} \qquad \begin{array}{c}[B]\\ \vdots \\ C\end{array}}{C}$$

which says that we can derive C from $A \vee B$ if we can derive C from A and also from B. It is essentially proof by cases: if we can derive C from both A and B separately, then we can derive it from $A \vee B$ as well.

With the Curry-Howard correspondence, we get:

[21] The so-called first dogma of standard semantics, see Schroeder-Heister (2008), Schroeder-Heister (2012).

[22] I would like to thank one of the reviewers for drawing my attention to this issue. The reviewer also suggested analysing the rule of bivalence also called the rule of dilemma or the rule of excluded middle (see, e.g., Tennant (1978), p. 48 or Negri et al (2001), p. 12; recently also adopted by D'Agostino et al (2020) as the only assumption discharging rule) as an example of a classical rule. However, I chose the reductio ad absurdum rule because it is generally a more familiar rule and it lends itself more naturally to the functional interpretation in the style of the Curry-Howard correspondence.

$$\dfrac{c : A \vee B \qquad \begin{matrix}[x : A] \\ \vdots \\ d : C\end{matrix} \qquad \begin{matrix}[y : B] \\ \vdots \\ e : C\end{matrix}}{\mathbf{case}(c, x.d, y.e) : C}$$

where **case** is a constant capturing the behaviour of the disjunction elimination rule. It takes three arguments, specifically, a proof object c of $A \vee B$ and two further proof objects $x.d$ and $y.e$ (where x, y is bound in d and e) that capture the two derivations of C from disjuncts of $A \vee B$. These proof objects are then used to determine the specific proof object for C, i.e., $d(a)$ if $\mathbf{inl}(a) : A \vee B$ or $e(b)$ if $\mathbf{inr}(b) : A \vee B$, where **inl** and **inr** are injective functions that inform us from which proposition was the disjunction derived.

As we can see, this rule has three premises, two of them are hypothetical judgments, thus in its functional variant two functions should appear as premises. Applying the same principles as above, we can obtain the following rule:

$$\dfrac{A \vee B \qquad A \Rightarrow C \qquad B \Rightarrow C}{C}$$

which can be then formalized in CTT as:

$$\dfrac{c : A \vee B \qquad f : (A)C \qquad g : (B)C}{d(c, f, g) : C}$$

Note that the higher-order function d takes three arguments, two of them are functions. Compare this with the original variant of this rule where the derivations are coded via the terms $x.d$ and $y.e$ and instead of a higher-order function d we have a constant **case**.[23]

Next, we examine the reductio ad absurdum rule:[24]

$$\dfrac{\begin{matrix}[\neg A] \\ \vdots \\ \bot\end{matrix}}{A}$$

given the standard definition of negation $\neg A = A \supset \bot$, where $\bot$ denotes falsity, we can unpack this rule as follows:

$$\dfrac{\begin{matrix}[A \supset \bot] \\ \vdots \\ \bot\end{matrix}}{A}$$

[23] Note that from the viewpoint of the lower-order presentation of constructive type theory, **case** is essentially just an auxiliary symbol and not a proper object of the type theory (the same can be said about λ, Π, etc.). This is not the case with higher-order presentation where **case** can be regarded as a higher-order function d of type $(A : prop)(B : prop)(C : (A \vee B)prop)(c : A \vee B)((x : A)C(i(A,B,x)))((y : B)C(j(A,B,y)))C(c)$ where $i : (A : prop)(B : prop)(A)(A \vee B)$ corresponds to **inl** (analogously for j) and $\vee : (prop)(prop)prop$.

[24] It probably goes without saying, but we are now leaving the strictly intuitionistic principles behind the original Curry-Howard correspondence.

And, following Gabbay and de Queiroz (1992), we take it as a special case of their inferential counterpart of Peirce's law:

$$\dfrac{\begin{matrix}[A \supset B] \\ \vdots \\ B\end{matrix}}{A}$$

where B is replaced by $\bot$. The corresponding proof objects are as follows:

$$\dfrac{\begin{matrix}[x : A \supset B] \\ \vdots \\ b(x,\ldots,x) : B\end{matrix}}{\lambda x.b(x,\ldots,x) : A}$$

with the condition that $A \supset B$ is used as both a minor and a major premise.[25]

Like the implication introduction rule, this rule has one premise that is a hypothetical judgment. Thus, its functional variant will be:

$$\dfrac{A \supset B \Rightarrow B}{A}$$

which can be formalized as:

$$\dfrac{f : (A \supset B)B}{\lambda(f) : A}$$

with the reductio ad absurdum special case as $\dfrac{f : (A \supset \bot)\bot}{\lambda(f) : A}$.

Sidenote. Assumption discharging rules of natural deduction systems generally allow one to withdraw assumptions that have not occurred in a derivation (i.e., vacuous discharge) and/or discharge multiple occurrences of an assumption simultaneously (i.e., multiple discharge). How do these discharge policies relate to the functional approach?[26] Let us start with vacuous discharge. Consider, e.g., the theorem $A \supset (B \supset A)$, which can be proved as follows:

$$\dfrac{\dfrac{[A]}{B \supset A}}{A \supset (B \supset A)}$$

The application of implication introduction rule with vacuous discharge in the first derivation step, which can be more explicitly written as:[27]

[25] Fore more, see Gabbay and de Queiroz (1992), p. 1345.

[26] A question raised by one of reviewers.

[27] The first derivation of the proof above is a one-step derivation the conclusion of which is the same as its only assumption. This is considered as a legitimated derivation (see, e.g., Hindley and Seldin (1986), p. 261).

$$\dfrac{\begin{matrix}[A] \\ \vdots \\ A\end{matrix}}{B \supset A}$$

behaves in the same way as the application of implication introduction with non-vacuous discharge. The only difference is that we are withdrawing an assumption, B in this case, that has not been made, thus the original set of premises does not change. More formally (following Prawitz (1965), p. 23), we can represent the general form of an instance of an implication introduction rule as a couple $\langle\langle \Gamma, B\rangle, \langle \Delta, A \supset B\rangle\rangle$ where Δ and Γ are sets of propositions, $\Delta = \Gamma - \{A\}$ and $\langle \Gamma, A\rangle$ represents a deduction of A from Γ. The instance with vacuous discharge will be the same with the difference that A will not be amongst the premises Γ. In our example above, the first derivation step will then amount to the case $\langle\langle A, A\rangle, \langle A, B \supset A\rangle\rangle$, since $A - B = A$ (omitting curly brackets for singleton sets).

This translates straightforwardly into the functional approach by capturing the pairs $\langle \Gamma, B\rangle$ and $\langle \Delta, A \supset B\rangle$ as (potentially multi-argument) functions of types $(\Gamma)B$ and $(\Delta)A \supset B$. Thus, the derivation via the implication introduction rule with vacuous discharge in the example above:

$$\dfrac{[A]}{B \supset A}$$

can be captured as a function of type $(A)B \supset A$ (or as $((A)A)B \supset A$, if we want to use the more explicit form). This coincides with the typical use of vacuous discharge in natural deduction: to derive $B \supset A$ we just need A and no B is required, which is mirrored by the function of type $(A)B \supset A$ that requires only one argument of type A and returns an object of type $B \supset A$. Thus, the functional approach accommodates vacuous discharge.[28]

Next, let us consider multiple discharge. Suppose we have the following derivation:

$$\dfrac{\dfrac{A \supset (A \supset B) \qquad A}{A \supset B} \qquad A}{B}$$

assuming the next derivation step is via implication introduction rule, we can either discharge all occurrences of the assumption A simultaneously or not. If we are interested solely in deducibility, it does not matter which discharge strategy we choose (the set of provable theorems remains the same), but since the Curry-Howard correspondence applies only to systems without the multiple discharge policy (see, e.g., Troelstra and Schwichtenberg (2000), pp. 43–44, Thompson (1999), p. 191), in this paper I naturally presume that not all open assumptions of the same form are always withdrawn simultaneously. Hence, the functional approach does not make use of multiple discharge.

[28] Although I have discussed vacuous discharge only in connection with implication introduction rule, these observations can be analogously applied to other assumption withdrawing rules as well. For example, the general form for disjunction elimination would be $\langle\langle \Gamma_1, A \vee B\rangle, \langle \Gamma_2, C\rangle, \langle \Gamma_3, C\rangle, \langle \Delta, C\rangle\rangle$ where $\Gamma_1 \cup (\Gamma_2 - A) \cup (\Gamma_3 - B)$ (see Prawitz (1965)), which can be captured as $((\Gamma_1)A \vee B)((\Gamma_2)C)((\Gamma_3)C)(\Delta)C$.

3 Conclusion

In this paper I have argued that the Curry-Howard correspondence is not necessarily connected with the placeholder view of assumptions generally associated with natural deduction systems for intuitionistic propositional logic. Although in its basic form, assumptions, which correspond to free variables, can indeed be thought of as just holes to be filled, we can consider also a functional approach where derivations from assumptions are regarded as functions. On this account, assumptions are no longer just placeholders but domains of the corresponding functions. From the logical point of view, this move corresponds to the shift from reasoning with propositions to reasoning with sequents. Furthermore, I showed that this approach can be captured using constructive type theory, which also seems to be a good candidate for formalizing Schroeder-Heister's notion of ground for a whole sequent.

Acknowledgements I would like to thank the two anonymous referees for providing insightful comments which helped to improve this paper.

References

Došen K (1980) Logical constants: an essay in proof theory. PhD thesis, University of Oxford

Došen K (2016) On the paths of categories. In: Piecha T, Schroeder-Heister P (eds) Advances in proof-theoretic semantics, Springer International Publishing, Cham, pp 65–77, DOI https://doi.org/10.1007/978-3-319-22686-6_4

Došen K, Petrić Z (2004) Proof-theoretical coherence. Studies in Logic (Logic & Cognitive Systems), College Publications

D'Agostino M, Gabbay D, Modgil S (2020) Normality, non-contamination and logical depth in classical natural deduction. Studia Logica 108(2):291–357, DOI https://doi.org/10.1007/s11225-019-09847-4

Fitch FB (1952) Symbolic logic: an introduction. Ronald Press, New York

Francez N (2015) Proof-theoretic semantics. College Publications

Frege G (1991) Collected papers on mathematics, logic, and philosophy. John Wiley & Sons

Gabbay DM, de Queiroz RJGB (1992) Extending the Curry-Howard interpretation to linear, relevant and other resource logics. Journal of Symbolic Logic 57(4):1319–1365, DOI https://doi.org/10.2307/2275370

Gentzen G (1935) Untersuchungen über das logische Schließen. I. Mathematische Zeitschrift 39(1):176–210, DOI https://doi.org/10.1007/BF01201353

Geuvers H, Nederpelt R (2004) Rewriting for Fitch style natural deductions. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 3091:134–154, DOI https://doi.org/10.1007/978-3-540-25979-4_10

Hindley JR, Seldin JP (1986) Introduction to combinators and (lambda) calculus. Cambridge monographs on mathematical physics, Cambridge University Press

Jaśkowski S (1934) On the rules of suppositions in formal logic. Studia Logica 1

Klev A (2019a) A comparison of type theory with set theory. In: Centrone S, Kant D, Sarikaya D (eds) Reflections on the foundations of mathematics, Springer, Cham, DOI https://doi.org/10.1007/978-3-030-15655-8_12

Klev A (2019b) Name of the sinus function. In: Arazim P (ed) The Logica Yearbook 2018, College Publications

Martin-Löf P (1984) Intuitionistic type theory. Studies in proof theory, Bibliopolis

Negri S, von Plato J, Ranta A (2001) Structural proof theory. Cambridge University Press

Nordström B, Petersson K, Smith JM (1990) Programming in Martin-Löf's type theory: an introduction. International series of monographs on computer science, Clarendon Press

Nordström B, Petersson K, Smith JM (2001) Martin-Löf's type theory, Handbook of logic in computer science: Volume 5: Logic and algebraic methods. Oxford University Press, Oxford

Oliveira H (2019) On Dummett's pragmatist justification procedure. Erkenntnis pp 1–27, DOI 10.1007/s10670-019-00112-7

Pezlar I (2014) Towards a more general concept of inference. Logica Universalis 8(1):61–81, DOI https://doi.org/10.1007/s11787-014-0095-3

von Plato J (2012) Gentzen's proof systems: byproducts in a work of genius. Bull Symbolic Logic 18(3):313–367, DOI https://doi.org/10.2178/bsl/1344861886

Prawitz D (1965) Natural deduction: a proof-theoretical study. Dover Books on Mathematics Series, Dover Publications, Incorporated

Schroeder-Heister P (1984) A natural extension of natural deduction. Journal of Symbolic Logic 49(4):1284–1300, DOI https://doi.org/10.2307/2274279

Schroeder-Heister P (2008) Proof-theoretic versus model-theoretic consequence. In: Peliš M (ed) The Logica Yearbook 2007, Filosofia, pp 187–200

Schroeder-Heister P (2012) The categorical and the hypothetical: a critique of some fundamental assumptions of standard semantics. Synthese 187(3):925–942, DOI https://doi.org/10.1007/s11229-011-9910-z

Schroeder-Heister P (2014) Frege's sequent calculus. In: Indrzejczak A, Kaczmarek J, Zawidzki M (eds) Trends in Logic XIII : Gentzen's and Jaskowski's heritage 80 years of natural deduction and sequent calculi, Lodz University Press, Lodz, pp 233–245

Schroeder-Heister P (2016) Open problems in proof-theoretic semantics. Springer, Cham, pp 253–283, DOI https://doi.org/10.1007/978-3-319-22686-6_16

Sørensen MH, Urzyczyn P (2006) Lectures on the Curry-Howard Isomorphism, Volume 149 (Studies in Logic and the Foundations of Mathematics). Elsevier Science Inc., New York, NY, USA

Tennant N (1978) Natural logic. Edinburgh: Edinburgh University Press

Thompson S (1999) Type theory and functional programming. International computer science series, Addison-Wesley

Tichý P (1988) The foundations of Frege's logic. Foundations of Communication, Berlin: de Gruyter

Troelstra AS, Schwichtenberg H (2000) Basic proof theory. Cambridge University Press, DOI https://doi.org/10.1017/cbo9781139168717