Transcendental syntax IV : logic without systems

Jean-Yves Girard
Directeur de Recherches émérite
jeanygirard@gmail.com

January 24, 2022

For André

Abstract

A derealistic, system-free approach, with an example: arithmetic.

Keywords : logic, arithmetic, derealism.

1 BHK revisited

1.1 A system-free approach

According to a widespread prejudice, logic should depend upon a system limiting the validity of its laws. Typically, the excluded middle should be accepted in the classical chapel but refused in the intuitionistic bunker. A conception that Kreisel refuted in his day: the polemics as to $A \vee \neg A$ does not concern the system, but the connective, i.e., $\vee := \parr$ vs. $\vee := \oplus$.

The first evidences against this "fishbowl" view of logic date back to the early 1930's. Typically, Gentzen's *subformula property* which restricts proofs of A to the constituents of A, thus excluding the wider system in which A may have been proved. But the most spectacular blow against bunkerisation is to be found in BHK (Brouwer-Heyting-Kolmogoroff), which presents a sort of functional definition of proofs (section 3).

This approach, which does not refer to any system, acknowledges the fact that logic deals with pure reason, truths beyond discussion.

1.2 Axiomatic realism

Getting rid of systems means standing up against *axiomatic realism*, the duality between syntax and its realistic counterpart, semantics.

But axiomatics and semantics have little to do with proofs. Being concerned with *falsification*, they are, so to speak, scouting the intellectual wilderness: the consistency of $\neg A$ (or the existence of a model refuting A) shows that we shouldn't waste energy in trying to prove A. By telling us *where not to go*, they are very precious auxiliaries, but too warped to be anything more, since they yield *contingent* truths: A may be valid in system $\mathbb{T}$ and its negation $\neg A$ in system $\mathbb{U}$, both being consistent.

Axiomatics and semantics deal with counterexamples, i.e., impermanence. While our basic interest lies in logic, i.e., permanence.

1.3 The first leakage: emptiness

BHK, although the only approach respecting the meaning of the word "logic", has serious leaks. The most obvious being *emptiness*: what to do in presence of formulas with no proofs, typically the absurdity $\mathbf{0}$? Since $\neg A := A \Rightarrow \mathbf{0}$, a proof of a negation becomes a function with the empty set – an unfriendly fellow – as target; this forces the source to be empty as well, in which case the proof becomes the bleak empty function $\emptyset$.

The emptiness of $\mathbf{0}$ justifies the excluded middle: either A has a proof or it has none, in which case the empty function which maps proofs of A to proofs of $\mathbf{0}$ is a proof of $\neg A$. This is quite embarrassing and various modifications, none of them definite, have thus been proposed, yielding various *realisability* interpretations. Those "semantics of proofs" are only useful tools, not the real thing – just like a scout is not the Army.

The only way to fix the leakage is to allow all formulas to have proofs, a proposal which conflicts with consistency. Not quite indeed: it is enough to distinguish, among proofs, the real ones from those which are here "to fill the holes". A situation akin to what happens with computer folders: those who look empty to the user indeed harbour "invisible" files `.xxx` which contain essential informations, the name of the folder or the list of its visible files.

Every proposition, including the absurdity $\mathbf{0} := !(\text{フ} \parr \text{ヲ}) \otimes \text{ヲ}$ (section 3.4), admits "proofs". A *truth* criterion (section 3) will determine which ones are visible, i.e., "true"; in the case of absurdity, none.

1.4 The second leakage: operationality

The functions at work in definition 2 play an essential role, but their status remains rather vague. Should we understand them as computable (recursive) functions or plain set-theoretic graphs? Each answer leads to a specific *category* of morphisms, i.e., a semantics. Categories presuppose the *form* (whence the word "morphism"): their intrinsic essentialism makes them one of the best semantic artifacts, but surely not a way out the bunker.

It seems that rock bottom was hit with the *constellations* of [4], that I will rename *designs*. A product of the experience of proof-theory and computer science, they embody the lessons of Gentzen (their *stars* are sort of logic-free sequents), Herbrand (they socialise through *unification*), logic programming (they look like deterministic PROLOG programs) and proof-nets.

Under certain circumstances, two designs may merge to form a new one through a process that may diverge: this *normalisation* is akin to the traditional cut-elimination – or the *resolution* of logic programming.

This process, which corresponds to the functional application at work in definition 2, presupposes neither logic nor categories: the merger of two designs can be expressed as a composition… provided we select appropriate sources and targets, but there is no univoque way to do so.

1.5 The third leakage: language

BHK is concerned with those formulas taken from a given language, typically arithmetic. Of course, if we want to free ourselves from systems, we must be ready to consider new formulas and connectives, including eccentric ones, i.e., not limit ourselves to an *a priori* choice: we should be able to consider general propositions, not only those available in a particular fishbowl.

The naive definition of those language-free formulas, called *behaviours*:

A behaviour is any set of designs.

is not technically mature: it must be regulated, typically to exclude the nightmare of emptiness.

The basic example of such a regulation is given by the *correctness criterion* of proof-nets. Which can be expressed in terms of a duality between designs: $\mathcal{P}$, the one under testing vs. $\mathcal{T}$, the test. The test succeeds if the combination $\mathcal{P} + \mathcal{T}$ merges into a design of a certain form, notation

$$\mathcal{P} \perp \mathcal{T}$$

Hence given a set $\mathbf{P}$ of designs, we can define its orthogonal $\sim\mathbf{P}$, i.e., the set of tests it passes. The biorthogonal $\sim\sim\mathbf{P}$ is, so to speak, the regulated version of $\mathbf{P}$, indeed the behaviour generated by $\mathbf{P}$.

DEFINITION 1
A behaviour *is any non trivial set of designs equal to its biorthogonal.*

"Non trivial" means that the behaviour and its orthogonal are non empty. With denumerably many designs, the number of possible behaviours has the power of the continuum. No fishbowl can harbour that many propositions!

1.6 The fourth leakage: usine

This happened to be the only leakage ever observed in the literature. Assuming everything works swell – and it does with our definitions – remains the problem of the distinction between *usine* and *usage* (*factory* and *use*, the use of French emphasising the opposition). L'usage is nothing but the BHK definition, which yields functions, etc. L'usine is the place where we get the certainty that those so-called functions do what they mean to do.

The successful passing of the tests implies cut-elimination and consistency. Therefore incompleteness forbids any form of absolute certainty as to l'usine which usually involves infinitely many tests.

People addressing the issue did not seem to realise that they were up against incompleteness. For instance those asking that, besides the functional proof of definition 2, one should add an auxiliary proof that the function does what it means to do. But how to deal with this "meta-proof"? If we treat it in the BHK style, it will need in turn its own auxiliary proof, etc.: metas all the way down. In [8], Kreisel proposed to make the meta-proof a formal one in a system given in advance – but later claimed (private communication, circa 1979) that this was a practical joke.

We do know that consistency proofs are impossible, that the Hilbert program cannot be fixed. So let us address the issue without any dogmatism. A behaviour $\mathbf{G}$ is the orthogonal of a set of tests, a "preorthogonal". The most elementary behaviours admit finite preorthogonals and will therefore be subject to a completely finite checking. But the preorthogonal is, most of the time, infinite and there is no way to implement infinitely many tests: the fact that $\mathcal{F}$ is a BHK proof cannot be an absolute certainty. It can, however, be justified by the usual tools of mathematics, i.e., within set theory.

See annex, p. 22 for further developments.

2 The architecture of logic

2.1 Logic vs. set theory

We propose to delegate the abstract testing (usine) to set theory: this makes our ultimate – reductionist foundations – depend upon set theory. Just like axiomatic realism, whose justification boils down to some set-theoretic semantics. Both approaches, derealistic and realistic thus rest upon plain mathematics, so let us compare the two approaches in foundational terms.

Set theory is a system, but a well-established one, so flexible and universal that one hardly notices its boundaries: for us, it is mathematics, period. If we insist upon absolute certainty (section 2.3), we must acknowledge the possibility of a failure of this framework. This highly unreasonable occurrence would equally affect both approaches.

Set theory being incomplete, it is likely that it cannot establish that some proof is a proof, i.e., miss the fact that some design $\mathcal{P}$ belongs to some behaviour $\mathbf{G}$. But this limitation of the derealistic approach, based on far-reached unprovable statements, is mainly theoretical. On the other hand, the realistic approach is most effectively limited by the walls of its self-chosen prison. As a consequence, the metastatic proliferation of systems.

Take for instance my system $\mathbb{F}$ of fifty years ago [3]: *les candidats de réductibilité* – which are the prefiguration of behaviours – are handled by means of the comprehension principle. If we still see it as a system, we are bound to build extensions – not necessarily bad, like the *constructions* of Coquand [1] –, but sort of prisons anyway. Or we could dump the idea of any system and directly work on behaviours, with almost unlimited possibilities.

Last but not least, most systems are wrong because the semantic justification leaks. The notion is easily tampered with and "bad witnesses" eliminated: this is what happened to the embarrassing empty model of predicate calculus (section 2.3).

2.2 Systems vs. toolbox

So we don't quite need logical systems: if we are not happy with our formulas, connectives, etc., define new ones by biorthogonality, establish their basic properties and *add them to our data base*. This stock may take the form of an *open* toolbox containing various designs together with the name of the behaviour they belong to. A list of untyped artifacts – delogicalised proofs – together with their types, those types being attributed externally, by arbitrary mathematical methods. The toolbox requires no sophisticated logical structure, e.g., a sequent calculus formulation: we can even use the most archaic logical formulation (axioms and *Modus Ponens*), which allows us to draw consequences from the principles listed in the data base, i.e., combine the tools. No cut-elimination, normalisation, etc. at the level of the toolbox is needed, since it is the task of the tools themselves: when we combine them by *Modus Ponens*, they initiate a converging merging process.

This is a major improvement over the fishbowl approach for which each novelty prompted a change of system, the creation of a schismatic chapel. An approach which culminated with *logical frameworks* [7] where systems $\mathbb{T}, \mathbb{U}, \mathbb{V}, \ldots$ could be put under the same roof with no right to communicate : like hospital patients, each of them quarantined in his room, lest he contaminate the others.

The fact that l'usine has been delegated to current mathematics, i.e., set theory, makes our toolbox absolutely faultless – except the legitimate doubt (section 2.3). The only limit to this approach is our own imagination.

2.3 Certainty

The logical discussions of yesteryear were polluted by the obsession of foundations. We must adopt an adult approach to the question and reflect upon our certainties or, dually, our doubts.

Generally, the testing cannot actually be performed – it is infinite – and is delegated to set theory. It is *legitimate* to doubt as to the reliability of set theory – in the same way we cannot be absolutely confident in the daily return of the Sun. But these doubts are not quite *reasonable*. Some form of certainty thus arises from the set theoretic foundation of logic: I call it *epidictic*. Due to incompleteness, this certainty is only reasonable, not absolute: it leaves some room for limited, but legitimate, doubts.

The old foundational approach did not distinguish between legitimate and reasonable: it was seeking a sort of *apodictic* certainty – the one which leaves not the slightest doubt – and neglected anything irrelevant to this chimeric issue. It promoted a reductionistic viewpoint based on brute force – consistency as rock bottom –, thus excluding any sort of finesse.

Like any kind of religious approach, the developments of the apodictic ideology contradict its goals. The search for final justifications leads to overlook obvious mistakes, for which the doubt is more than legitimate, reasonable:

typically the ludicrous principle $\forall \Rightarrow \exists$. Based on the misuse of variables, it is obviously false; but consistent hence, from the apodictic ideology which deals with "strength", a neglectable drawback. The Al Capone method was applied to the embarrassing witness – the empty model – which refutes the nonsense: it was disposed of on the way to Court, this is why models are supposed to be non empty!

2.4 Constraints and freedom

As we observed with the dubious $\forall \Rightarrow \exists$, each axiomatic system can be justified by means of an *ad hoc* relation to reality. This is precisely why their results are not portable: these systems are prisons, with their own approach to reality, what they call semantics. If we can still use such a prostituted expression, derealism is *the* ultimate semantics.

It is therefore much demanding and does not content itself with a model. For instance, they were serious grounds for the logical constants $\mathbf{1}$ and $\perp$ of linear logic: no need to explain the interest of having neutral elements for the multiplicative connectives. However, altough a considerable amount of energy was devoted to that peculiar task, the theory of proof-nets never worked for those constants. There is only one way out, namely accept the fact that $\mathbf{1}$ and $\perp$ are wrong, i.e., impossible. Forcing them to integrate the bulk of logic would destroy the whole architecture. By the way, if we insist upon something of the like, $\forall X\,(X \multimap X)$ and $\exists X\,(X \otimes {\sim} X)$ will provide reasonable ersatz, but not the real thing which remains a logical fantasy.

The point of good constraints is that they create freedom. Derealism refuses $\mathbf{1}$ and $\perp$ but accepts equality, the most notorious failure of axiomatic realism, based upon the Leibniz definition

Any property of a *is a property of* b.

As observed in [5], individuals a and b can never be equal, since they can be distinguished by their position w.r.t. "and". Axiomatic realism will object by claiming that we are actually speaking of the respective denotations, i.e., semantics, of these objects and that properties should be consistent with denotations. But how do we know that a property only depends upon the semantics? Elementary, my dear Watson: when it is compatible with... equality! This circular riddle is implemented in various systems telling us which properties are legit. Hence, without system, no Leibniz definition, no equality. By the way, the proof-theoretical treatment of equality is admittedly *ad hoc*: it involves generalised identity axioms embodying the cuts one cannot eliminate, e.g., $t = u, v = u, A[t] \vdash A[v]$.

But who told us that there is a special, segregated category of "individuals" proceeding from the Sky; furthermore that they harbour properties in the same way dogs have a tail? Wouldn't it be simpler if those individuals were just plain propositions, equality being equivalence? This obvious solution can indeed be used to define natural numbers and *prove* the third and fourth Peano axioms (section 5). Exit the aporia of the Leibniz equality.

So why did it take so long to integrate the most natural logical primitive? Simply because of the classical prejudice: up to consistency, everything is classical, hence the excluded middle

$$A \equiv B \ \vee \ B \equiv C \ \vee \ C \equiv A \tag{1}$$

which implies the impossibility of three unequivalent propositions. Intuitionism, which does not agree on this, does not disagree either, i.e., proves $\neg\neg\,(1)$. Linear logic – which should not be seen as a system, but a space of freedom –, by restricting the contraction rule to specific cases, makes (1) the exception, by no means the rule. No doubt a useful exception, but which can be a pain in the neck in some cases.

Another issue related to freedom: the paper [6] introduced *light exponentials*, i.e., connectives dedicated to perenniality, with some relation to computational complexity. They were developed in various systems (BLL, LLL, ELL...) whose relative qualities I shall not discuss for the very reason that we move on sort of quicksand, with no real benchmark: the semantics turns out to be more treacherous than ever. This is why it would be of utmost importance to determine whether or not light exponentials are more than a figment of axiomatic realism, in other terms whether they can be explained in terms of behaviours.

3 Truth

3.1 The tarskian pleonasm

It suffices to compare BHK

DEFINITION 2
A proof of $A \Rightarrow B$ is a function $\mathcal{F}$ mapping any proof $\mathcal{P}$ of A to a proof $\mathcal{F}(\mathcal{P})$ of B.

to Tarski's "definition" of truth, e.g.,

DEFINITION 3
$A \Rightarrow B$ is true when the truth of A implies *the truth of B.*

(and its declinations for $\wedge, \vee, \neg, \ldots$ in terms of and, or, not, ...) to see the difference between an inspired approach and a pleonasm which boils down to "A is true when A". But the truism is the ultimate form of snobbery: you think the Emperor is naked, mistake, you just don't see his new clothes.

Indeed, the famous *vérité de La Palice*, a theory of truth due to a French precursor of analytic philosophy, e.g.,

Un quart d'heure avant sa mort, il était encore en vie.

foreshadows definition 3.

The current opinion among non believers is that tarskian truth is, unfortunately, correct. But even this correctness is dubious, since truth does not apply to formulas but to proofs! Section 4.3 wil provide us with examples *contradicting* the tarskian definition, which is thus not even a pleonasm.

3.2 Generalities about visibility

Remember that we definitely dumped fishbowls, hence no longer deal with the formulas of a language, but with general behaviours (definition 1). Our definition of truth takes the form:

DEFINITION 4
G *is true when it harbours* a visible *design.*

The visible designs are the true ones, the actual *proofs* so to speak. Visibility, yet to be defined, should enjoy certain implicit requirements:

- It should be closed under cut: hence, if $\mathcal{P}$ and $\mathcal{F}$ are proofs of **G** and **G** $\Rightarrow$ **H**, then the design $\mathcal{F}(\mathcal{P})$ of **H** must be visible, i.e., a proof of **H**.
- Some behaviour, typically the absurdity **0**, must be without visible element, i.e., not true.

If these requirements are satisfied, then truth is consistent: $\mathbf{G}$ and its classical negation $\mathbf{G} \Rightarrow \mathbf{0}$ cannot both have visible designs, i.e., both be true. An exclusion that does not extend to linear negation: the self-dual behaviours $\text{フ} = \sim\text{フ}$ and $\text{ヲ} = \sim\text{ヲ}$ are true.

Since truth deals with proofs and not with mere provability, the truth of a compound behaviour cannot be reduced to the truth of its constituents. Therefore it cannot follow any kind of truth table. In particular, a conjunction may be true while one of the conjuncts is not. So tarskian truth is worse than a useless and snobbish ready-made, it is a plain mistake!

3.3 Multiplicative case

We shall first explain the solution in the case of the multiplicative proof-nets of linear logic; we consider formulas built from literals $p, \sim p, q, \sim q, r, \sim r, \ldots$ by means of $\otimes$ and $⅋$. Besides the usual $\otimes$ and $⅋$-links, we allow arbitrary links $\overbrace{p_1, \ldots, p_k}$, with $k > 0$, which resemble axioms in the sense that they are without premise. The usual correctness criterion is applied to the structures built from those, $\overbrace{p_1, \ldots, p_k}$ being seen as a vertex with edges $p_1, \ldots, p_k$: this generalises the usual case based on the sole $\overbrace{p, \sim p}$, see section 3.5 below.

A proof structure with literals $q_1, \ldots, q_N$ (with possible repetitions, this is the familiar nonsense about "occurrences") can be seen as a partition $\mathcal{P}$ of $\{1, \ldots, N\}$ the classes of which are precisely the "axioms" $\overbrace{q_{i_1}, \ldots, q_{i_k}}$ used. A switching of the proof-net yields another partition $\mathcal{T}$ of the same $\{1, \ldots, N\}$. Both partitions can be put together to form a bipartite graphs: the classes being its vertices, an edge between $X \in \mathcal{P}$ and $Y \in \mathcal{T}$ is an element of $\mathcal{P} \cap \mathcal{T}$. The Danos-Regnier criterion [2] requires that, for any $\mathcal{T}$ arising from a switching, the induced graph is connected and acyclic. In particular $X \in \mathcal{P}$ and $Y \in \mathcal{T}$ intersect in at most one point.

Let n and m be the respective numbers of partitions in $\mathcal{P}$ and $\mathcal{T}$. If the proof is correct, then $n + m - N = 1$ (Euler-Poincaré), what can be written $(2n - N) + (2m - N) = 2$. The *weight* $|\mathcal{P}| := 2n - N$, which does not depend upon $\mathcal{T}$, can be written as the sum of the weights of its "axiom links" defined by $|\overbrace{p_1, \ldots, p_k}| = 2 - k$. Our visibility definition writes as:

Definition 5
$\mathcal{P}$ is visible when $|\mathcal{P}| \geq 0$.

Observe that the weight of the familiar $\overbrace{p, \sim p}$ is $2-2=0$, hence a proof-net using the familiar identity axioms is of total weight 0, hence visible.

Visibility satisfies the requirements of the previous section. First, it is deductively closed: normalising a cut amounts at replacing $\overbrace{p_1, \ldots, p_k, p}$ and $\overbrace{\sim p, q_1, \ldots, q_\ell}$ (total weight $2-(k+1)+2-(\ell+1) = 2-(k+\ell)$) with $\overbrace{p_1, \ldots, p_k, q_1, \ldots, q_\ell}$ (weight $2-(k+\ell)$, i.e., the same). Moreover, not everything is true: typically, $p \parr q \parr r$, whose only correct proof-net, which uses the axiom $\overbrace{p, q, r}$ of weight -1, is invisible.

Incidentally, we gave the fatal blow to tarskian truth: $(p \parr q \parr r) \otimes s$ is true while $p \parr q \parr r$ is not.

3.4 The constants are dead, long live the constants!

Our multiplicative example has been oversimplified for pedagogic purposes. Atoms indeed split into two classes, *objective* and *subjective*, each one being closed under negation. This modification makes it possible to handle the absurdity $\mathbf{0}$ and is the key to second order (section 5.2). It only affects the weighing of "axioms", written $\overbrace{p_1, \ldots, p_k, q_1, \ldots, q_\ell}$, the p_i being objective, the q_j subjective.

- If $\ell = 0$, i.e., if the axiom is objective, then $|\overbrace{p_1, \ldots, p_k}| = 2-k$.
- Otherwise, $|\overbrace{p_1, \ldots, p_k, q_1, \ldots, q_\ell}| = -k$

Subjective atoms, whatever their number, count for two objective ones.

Keeping definition 5 of visibility, it remains to show the deductive closure of truth. $|\overbrace{p_1, \ldots, p_k, q_1, \ldots, q_\ell, p}|$ takes the value $-k$ if p is subjective; if p is objective, it takes one of the values $2-(k+1)$ (if $\ell = 0$) and $-(k+1)$ (if $\ell \neq 0$). Ditto with $|\overbrace{\sim p, r_1, \ldots, r_{k'}, s_1, \ldots, s_{\ell'}}|$: possible weights $-k'$, $2-(k'+1)$ and $-(k'+1)$. Both of them weight $a-(k+k')$ where a takes one of the values $2, 0, -1, -2$: $a = 1$ is excluded since this would require, say, p to be objective and $\sim p$ subjective. On the other hand, $\overbrace{p_1, \ldots, p_k, q_1, \ldots, q_\ell, r_1, \ldots, r_{k'}, s_1, \ldots, s_{\ell'}}$ weights $2-(k+k')$ or $-(k+k')$. The weight can decrease during normalisation only if $a = 2$, in case $\ell = \ell' = 0$ but the normalised "axiom" would weight $2-(k+k')$.

Indeed, up to linear equivalence, there are only two atoms, the objective フ ("fu") and the subjective ヲ ("wo"). Both are provable, self-dual and inequivalent. They can be used to define the absurdity by $\mathbf{0} := !(\text{フ} \parr \text{ヲ}) \otimes \text{ヲ}$. Indeed, section 4.2 of [5], proves, without using the notations (フ and ヲ were still in limbo) the rule

$$\frac{\vdash \Gamma, A}{\vdash \Gamma, \top}$$

which is an alternative formulation of the famous *ex nihilo quod libet* $\mathbf{0} \multimap A$. Incidentally, the notion of *épure* (= working drawing) of paper [5] is different: either $k = 2, \ell = 0$ or $k = 0$. This ensures that $|\overbrace{p_1, \ldots, p_k, q_1, \ldots, q_\ell}| = 0$, hence épures are visible.

The constants ($\mathbf{1}$ and $\perp$) are dead, long live the constants (フ and ヲ). Whose multiplicative combinations yield natural numbers (section 4 below).

3.5 Variables

According to a dubious tradition, propositional calculus should be built from unspecified constants $P, Q, R, \ldots$. Weird constants indeed, for which anything can be substituted: this is what one usually calls variables! But such variables should then be styled *second order*, a part of logic against which a fatwa was declared. Let us call a spade a spade and use the notation $X, Y, Z, \ldots$ to emphasise the fact that we are dealing with variables.

Those variables were part of proof-nets original, since we needed some sorts of atoms. Those proof-nets made use of binary identity links $\overbrace{X, \sim X}$. They are compatible with our truth definition, since they are binary; their weight is always zero, since X and $\sim X$ are simultaneously objective or subjective.

The restriction to the links $\overbrace{X, \sim X}$ has nothing to do with a sort of systemic ukase, it can be derived from closure under instanciation: the net should remain correct when we replace its variables with independent propositions. This can take the form of a switching (already presented in [4], but without the notation フ), involving the selection of a "value" for each variable X with three cases :

$$\begin{array}{llll} X & := \text{フ} & \sim X := \text{フ} \\ X & := \text{フ} \otimes \text{フ} & \sim X := \text{フ} \parr \text{フ} \\ X & := \text{フ} \parr \text{フ} & \sim X := \text{フ} \otimes \text{フ} \end{array}$$

This excludes all possible practical jokes, e.g., $\overbrace{\ X\ }, \overbrace{X, Y}, \overbrace{X, X}, \overbrace{X, X, \sim X}$.

3.6 General case

We are not quite dealing with proof-nets, but with the designs of a behaviour. The main difference with the multiplicative case is that duplications and erasings may occur during normalisation. Our numerical criterion is obviously sensitive to these operations, hence we must be cautious.

The truth of $\mathcal{P} \in \mathbf{G}$ is related to the testing process. So let $\mathcal{T}$ be a test in $\sim\mathbf{G}$, hence $\mathcal{P} \perp \mathcal{T}$. The actual performance of the test, a normalisation in the sense of [4], involves the building of a connected-acyclic graph whose vertices are made of two designs, $\mathcal{P}_{\mathcal{T}}$ and $\mathcal{T}'$, each ray of $\mathcal{P}_{\mathcal{T}}$ being a ray of $\mathcal{T}'$; the edges are those common rays. $\mathcal{P}_{\mathcal{T}}$ and $\mathcal{T}'$ are obtained through a unification (matching) procedure which replaces any star σ of those designs with various substitutions $\sigma\theta_i$.

The visibility of $\mathcal{P}$ is obtained by means of a weighing of the stars of $\mathcal{P}_{\mathcal{T}}$. Remembering that rays are divided into objective and subjective ones, let $\llbracket t_1, \ldots, t_k, u_1, \ldots, u_\ell \rrbracket \in \mathcal{P}_{\mathcal{T}}$, then:

- If $\ell = 0$, then $|\llbracket t_1, \ldots, t_k \rrbracket| = 2 - k$.
- Otherwise, $|\llbracket t_1, \ldots, t_k, u_1, \ldots, u_\ell \rrbracket| = -k$.

The closure of visibility under cut is the consequence of the fact that the matching between t and u of complementary colours is impossible if one is objective and the other subjective. Generally, the testing should anticipate general normalisation; in terms of truth, it should make sure that the $\mathcal{P}_{\mathcal{T}}$ are representative of the $\mathcal{P}_{\mathcal{Q}}$ occurring during the actual normalisation of a cut between $\mathcal{P}$ and a design $\mathcal{Q}$, visible or not, in some $\vdash \sim\mathbf{G}, \mathbf{\Gamma}$.

4 Natural numbers

We now restrict our attention to the multiplicative combinations of the self-dual constants ꓶ and ꓱ. We shall classify them up to linear equivalence (i.e., logical equality) $A \equiv B := (A \multimap B) \,\&\, (B \multimap A)$.

4.1 First series

Definition 6

The weight *of the multiplicative A built from the sole* フ *is defined as the common weight of all designs of A:*

$$\begin{array}{rcl} |\text{フ}| & = & 1 \\ |A \otimes B| & = & |A| + |B| \\ |A \parr B| & = & |A| + |B| - 2 \end{array}$$

In particular, $|\sim A| = 2 - |A|$ and $|A \multimap B| = |B| - |A|$.

For $n > 0$ define $\text{フ}_n := \text{フ} \otimes \text{フ} \otimes \ldots \otimes \text{フ}$ (a n-ary tensor) and for $n < 2$ $\text{フ}_n := \text{フ} \parr \text{フ} \parr \ldots \parr \text{フ}$ (a $2-n$-ary par), both cases agreeing on $\text{フ}_1 := \text{フ}$. Observe that $\sim\text{フ}_n \equiv \text{フ}_{2-n}$.

Theorem 1

$A \equiv \text{フ}_{|A|}$

Proof : by recurrence on the size of A, the basic case $A = \text{フ}$ being trivial. If A is a tensor $B \otimes C$ and $B \equiv \text{フ}_{|B|}, C \equiv \text{フ}_{|C|}$, then $A \equiv \text{フ}_{|B|} \otimes \text{フ}_{|C|} \equiv \text{フ}_{|A|}$. If A is a "par" $B \parr C$, the previous case shows that $\sim A$ is equivalent to $\text{フ}_{|\sim A|}$, hence $A \equiv \sim\text{フ}_{2-|A|} \equiv \text{フ}_{|A|}$. $\square$

$\text{フ}_0 = \text{フ} \parr \text{フ}$ is a sort of corrected version of the late neutral $\mathbf{1}$, ditto for $\text{フ}_2 = \text{フ} \otimes \text{フ}$ w.r.t. $\perp$. フ_0 and フ_2 are, so to speak, the logical part of the multiplicative units. They basically work because フ and ヲ no longer follow any semantic paradigm!

All フ_n, for $n \geq 0$, are provable. As a particular case, フ_0, フ_1 and フ_2 are provable together with their linear negations フ_2, フ_1 and フ_0. For $n < 0$, the フ_n are not provable; they are indeed *refutable* (section 4.3).

4.2 Second series

For $n \in \mathbb{Z}$, we define the ヲ_n : $\text{ヲ}_0 := \text{ヲ}$, $\text{ヲ}_n := \text{フ}_n \otimes \text{ヲ}$ when $n \neq 0$.

Proposition 1

$\text{ヲ} \equiv \text{フ}_0 \otimes \text{ヲ}$

Proof : from $\vdash \text{ヲ}, \text{ヲ}$ and $\vdash \text{フ}_0$, we get $\vdash \text{ヲ}, \text{フ}_0 \otimes \text{ヲ}$, hence the implication $\text{ヲ} \multimap \text{フ}_0 \otimes \text{ヲ}$. Conversely, $\vdash \text{フ}$ and $\vdash \text{フ}, \text{ヲ}, \text{ヲ}$ admit designs of respective weights 1 and -1 which combine into a design of weight 0 of $\vdash \text{フ} \otimes \text{フ}, \text{ヲ}, \text{ヲ}$ which yields a proof of $\text{フ}_0 \otimes \text{ヲ} \multimap \text{ヲ}$. $\square$

Hence $\text{ヲ}_n \equiv \text{フ}_n \otimes \text{ヲ}$ for all $n \in \mathbb{Z}$. More generally:

Proposition 2
$\text{フ}_m \otimes \text{ヲ}_n \equiv \text{ヲ}_{m+n}$

(obvious) and

Proposition 3
$\text{ヲ}_m \otimes \text{ヲ}_n \equiv \text{ヲ}_{m+n}$

Proof : boils down to $\text{ヲ} \otimes \text{ヲ} \equiv \text{ヲ}$. From $\vdash \text{ヲ}, \text{ヲ}, \text{ヲ}$, we get $\text{ヲ} \otimes \text{ヲ} \multimap \text{ヲ}$; conversely, $\vdash \text{ヲ}, \text{ヲ}$ and $\vdash \text{ヲ}$ yield $\vdash \text{ヲ}, \text{ヲ} \otimes \text{ヲ}$, hence $\text{ヲ} \multimap \text{ヲ} \otimes \text{ヲ}$. $\square$

Proposition 4
$\text{フ}_{n+2} \parr \text{ヲ} \equiv \text{ヲ}_n$

Proof : from $\vdash \text{ヲ}, \text{ヲ}$ and designs in $\sim \text{フ}_{n+2}$ and フ_n of respective weights $-n$ and n, one gets a proof of $\vdash (\sim \text{フ}_{n+2} \otimes \text{ヲ}), (\text{フ}_n \otimes \text{ヲ})$, hence $\text{フ}_{n+2} \parr \text{ヲ} \multimap \text{ヲ}_n$. Conversely, from $\vdash \text{ヲ}, \text{フ}_2, \text{ヲ}$, we get $\vdash \sim \text{フ}_n \parr \text{ヲ}, (\text{フ}_2 \otimes \text{フ}_n) \parr \text{ヲ}$, hence $\text{ヲ}_n \multimap \text{フ}_{n+2} \parr \text{ヲ}$. $\square$

Theorem 2
Any multiplicative combination A of フ and at least one ヲ is provably equivalent to some ヲ_n.

Proof : by recurrence on the size of A, the basic case $A = \text{ヲ}$ being trivial. If A is a tensor $B \otimes C$, at least one of B and C uses a ヲ and we are left with the cases $\text{ヲ}_m \otimes \text{ヲ}_n$, $\text{フ}_m \otimes \text{ヲ}_n$ and $\text{ヲ}_m \otimes \text{フ}_n$ which by propositions 2 and 3 are equivalent to ヲ_{m+n}. If A is a "par" $B \parr C$, the previous case shows that $\sim A$ is equivalent to some ヲ_n, hence $A \equiv \text{フ}_n \parr \text{ヲ}$; using proposition 4, we get $A \equiv \text{ヲ}_{n-2}$. $\square$

Definition 6 can be extended to multiplicative combinations of フ and ヲ:

Definition 7

$$\begin{array}{rcl}
|\text{フ}| & = & 1 \\
|\text{ヲ}| & = & 0 \\
|A \otimes B| & = & |A| + |B| \\
|A \parr B| & = & |A| + |B| - 2 \quad \text{if one of} A, B \text{ is} \text{ヲ}-\text{free} \\
|A \parr B| & = & |A| + |B| \quad \text{otherwise}
\end{array}$$

By 1 and 2, A is equivalent to either $\text{フ}_{|A|}$ or $\text{ヲ}_{|A|}$. In general:

1. $\text{フ}_m \otimes \text{フ}_n \equiv \text{フ}_{m+n}$ and $\text{フ}_m \parr \text{フ}_n \equiv \text{フ}_{m+n-2}$
2. $\sim\text{フ}_n \equiv \text{フ}_{2-n}$ and $\text{フ}_m \multimap \text{フ}_n \equiv \text{フ}_{n-m}$
3. $\text{ヲ}_m \otimes \text{ヲ}_n \equiv \text{ヲ}_m \parr \text{ヲ}_n \equiv \text{ヲ}_{m+n}$
4. $\sim\text{ヲ}_n \equiv \text{ヲ}_{-n}$ and $\text{ヲ}_m \multimap \text{ヲ}_n \equiv \text{ヲ}_{n-m}$
5. $\text{ヲ}_m \otimes \text{フ}_n \equiv \text{ヲ}_{m+n}$ and $\text{ヲ}_m \parr \text{フ}_n \equiv \text{ヲ}_{m+n-2}$
6. $\text{ヲ}_m \multimap \text{フ}_n \equiv \text{ヲ}_{n-m-2}$ and $\text{フ}_m \multimap \text{ヲ}_n \equiv \text{ヲ}_{n-m}$

4.3 Truth and falsity

THEOREM 3
The フ_n and ヲ_n are refutable for $n < 0$.

Proof : $\text{フ}_n \multimap \text{ヲ}_n$ being equivalent to ヲ_0 ($= \text{ヲ}$), is provable; $\neg\text{ヲ}_n \multimap \neg\text{フ}_n$ is thus provable, which reduces the theorem to the case of ヲ_n. Now $\text{ヲ}_n \multimap \text{ヲ}_{-1}$ being equivalent to the provable ヲ_{-1-n}, we are reduced to proving $\neg\text{ヲ}_{-1}$: from $\vdash \sim\text{ヲ}_{-1}, \text{ヲ}_{-1}$ and $\vdash \text{フ}$, we get $\text{ヲ}_{-1} \Rightarrow\ !\text{ヲ}_{-1} \otimes \text{フ}$, i.e., $\text{ヲ}_{-1} \Rightarrow \mathbf{0}$ that is the negation $\neg\text{ヲ}_{-1}$. $\square$

Let us sum up the basic facts about truth and falsity (i.e., refutability):

1. The フ_n and ヲ_n are true for $n \geq 0$, false for $n < 0$.
2. The implications $\text{フ}_m \multimap \text{フ}_n$, $\text{フ}_m \multimap \text{ヲ}_n$ and $\text{ヲ}_m \multimap \text{ヲ}_n$ are true for $m \leq n$, false when $m > n$.
3. The implication $\text{ヲ}_m \multimap \text{フ}_n$ is true when $n \geq m+2$, false otherwise.

The two series are thus distinct, the sole relation between them being the double implication

$$\text{フ}_n \multimap \text{ヲ}_n \multimap \text{フ}_{n+2}$$

We definitely *contradict* the excluded middle (1) which forbids the existence of three provably unequivalent popositions! This implies necessary divergences from classical truth which are made possible by the fact that our truth applies to proofs and not to propositions. In particular the novelty cannot be tamed by a change of truth tables, say replacing **t,f** with $\mathbb{Z}$. Typically, A of weight n can be equivalent to フ_n or ヲ_n.

The following table is a list of possible deviations (with $\mathbf{t}$ = true, $\mathbf{f}$= false) w.r.t. classical truth. The first line, with $A = B = \text{フ}_0$, yields $A \parr B = \text{フ}_{-2}$ and $\sim A = \text{フ}_2$. The second line, with $A = \text{フ}_{-1}, B = \text{フ}_1$, yields $A \otimes B = \text{フ}_0$ and $A \parr B = \text{フ}_{-2}$. No deviation when both A and B are false. "$\parr$" is more deviant than "$\otimes$": this is because negation does not exchange **t** and **f**.

A	B	$A \otimes B$	$A \parr B$	$\sim A$
t	**t**		**f**	**t**
f	**t**	**t**	**f**	

A definite jailbreak from tarskism... and any sort of semantics.

5 Arithmetic

We shall now reconstruct arithmetic, not as a system, but as part of our open logic. We basically need two sorts of quantifiers, first and second order.

5.1 First order quantification

First order quantification is about relative numbers, identified with the series フ_n ($n \in \mathbb{Z}$). The following can serve as a definition of *individuals*:

1. The variables $\mathtt{x}, \mathtt{y}, \mathtt{z}, \ldots$ are individuals.

2. $\overline{\mathtt{1}} := \text{フ}$ is an individual.

3. If $\mathtt{t}, \mathtt{u}$ are individuals, so are $t + u := t \otimes u$ and $t - u := u \multimap t$.

Since logic is open, we don't even require that (1)–(3) be the only way to build individuals.

The usual rules of quantification do apply, provided we *declare* our variables. Incidentally, due to the presence of the closed individual フ, the principle $\forall \multimap \exists$ holds : $\forall \mathtt{x} A[x] \multimap A[\text{フ}]$ and $A[\text{フ}] \multimap \exists \mathtt{x} A[x]$.

Variables indeed stand for arbitrary behaviours, analogous to the so-called propositional "constants", indeed variables, of logic. The basic parametric proposition (i.e., predicate) is inequality:

$$\mathtt{t} \leq \mathtt{u} \quad := \quad \mathtt{t} \multimap \mathtt{u}$$

From which we can define equality:

$$\mathtt{t} = \mathtt{u} \quad := \quad (\mathtt{t} \multimap \mathtt{u}) \,\&\, (\mathtt{u} \multimap \mathtt{t})$$

The standard principles of linear logic allow us to establish certain principles which are usally handled via axiomatics. Typically:

$$\begin{aligned} \mathtt{x} \quad &\leq \quad \mathtt{x} && (2)\\ \mathtt{x} + (\mathtt{y} + \mathtt{z}) \quad &= \quad (\mathtt{x} + \mathtt{y}) + \mathtt{z} && (3)\\ \mathtt{x} + (\mathtt{y} - \mathtt{z}) \quad &\leq \quad (\mathtt{x} + \mathtt{y}) - \mathtt{z} && (4) \end{aligned}$$

Let $\overline{\mathtt{0}} := \text{フ} - \text{フ} (= \text{フ} \parr \text{フ})$. Since individuals deal with relative numbers, the third Peano axiom takes the form:

$$\overline{\mathtt{0}} \leq \mathtt{x} \quad \multimap \quad (\mathtt{x} + \text{フ}) \neq \overline{\mathtt{0}} \tag{5}$$

Which can be proved as follows: from $\overline{\mathtt{0}} \leq \mathtt{x}$ we get $\overline{\mathtt{0}} + \text{フ} \leq \mathtt{x} + \text{フ}$, which, combined with $(\mathtt{x} + \text{フ}) = \overline{\mathtt{0}}$, yields the refutable $\overline{\mathtt{0}} + \text{フ} \leq \overline{\mathtt{0}}$.

As to the fourth Peano axiom, the best we can get is the following:

$$(\mathtt{x} + \text{フ}) \leq (\mathtt{y} + \text{フ}) \quad \multimap \quad \mathtt{x} \leq ((\mathtt{y} + \text{フ}) - \text{フ}) \tag{6}$$

which makes use of $\mathtt{x} \multimap ((\mathtt{x} + \text{フ}) - \text{フ})$. The implication $((\mathtt{x} + \text{フ}) - \text{フ}) \multimap \mathtt{x}$ is missing; it is however provable when $\mathtt{x}$ is a "successor":

Theorem 4

$$((\mathtt{x} + \text{フ} + \text{フ}) - \text{フ}) \leq \mathtt{x} + \text{フ}$$

Proof : $\vdash \sim\mathtt{x}, \mathtt{x}$ and $\vdash \text{フ}, \text{フ}, \text{フ}$ (weight -1) yield $\vdash \sim\mathtt{x} \parr \text{フ} \parr \text{フ}, \mathtt{x} \otimes \text{フ}$ (weight -1) hence, with $\vdash \text{フ}$ (weight 1), $\vdash (\sim\mathtt{x} \parr \text{フ} \parr \text{フ}) \otimes \text{フ}, \mathtt{x} \otimes \text{フ}$. □

Summing up, we conclude that the fourth Peano axiom holds for successors, so to speak $\mathtt{SSx} = \mathtt{SSy} \multimap \mathtt{Sx} = \mathtt{Sy}$.

In terms of proof-nets, universal quantification is handled, as in [5], by means of a switching choosing independent values for the variables: the choices $\mathtt{x} = \text{フ}, \mathtt{x} = \text{フ} \parr \text{フ}, \mathtt{x} = \text{フ} \otimes \text{フ}$ are enough (section 3.5 *supra*).

Existential quantification is handled as in [5], with a major simplification: the existential witnesses $\mathcal{G} + \tilde{\mathcal{G}}$ were defined as linear combinations of all elements in the *finite* $\mathcal{G}$ and $\tilde{\mathcal{G}}$. We simplify our construction by using, instead of the full $\mathtt{t}$ and $\sim\mathtt{t}$ a specific test in each of them. With two consequences:

- We no longer use linear combinations (good riddance!).
- The same simplification can be used in the second order case where behaviours are infinite.

First-order is basically weaker than the usual first order of Peano, who could use axiomatics to decide which primitive is legal or not or which principle is true. Since we are concerned with logic, we have no longer access to ukases and are unable to establish the full fourth Peano axiom or define the product $\mathtt{t} \cdot \mathtt{u}$. The missing "axiom" is trivially proved under the form $\mathtt{x} \in \mathbb{N} \multimap ((\mathtt{x} + \text{フ}) - \text{フ}) \multimap \mathtt{x}$ by a recurrence (section 5.3), a second order principle[1], just as the missing product is second order definable (section 5.4).

By the way, one of the blind spots of BHK was the handling of purely universal statements of arithmetic. Basically a proof of $\forall \mathtt{x} A[\mathtt{x}]$ is treated pointwise as a function mapping $n \in \mathbb{N}$ to a proof of $A[\overline{\mathtt{n}}]$, which, being a plain computation, can be described in advance, hence the "proof" reduces to the "meta-proof" of section 1.6, which in turn reduces to meta-meta-proofs all the way down. Observe that $((\overline{\mathtt{n}} + \text{フ}) - \text{フ}) \multimap \overline{\mathtt{n}}$ holds pointwise but that the proofs do not proceed from a common design; there is indeed one for $n > 0$ which does not merge with the case $n = 0$.

5.2 Second order propositional case

Although there is no use for it, let us start with second order propositional quantification, i.e., system $\mathbb{F}$. This was the stumbling block of [5], due to the fact that behaviours are usually infinite: we cannot encapsulate an infinite set inside a design. By the way, should we attempt such a nonsense, we would enter into a wild goose chase as to the cardinality of behaviours.

The original treatment of $\mathbb{F}$ [3] involved *candidats de réductibilité*, which suggests the following definition.

[1]Which also proves $\mathtt{x} \in \mathbb{N} \multimap \overline{\mathtt{0}} \leq \mathtt{x}$, hence $\mathtt{x} \in \mathbb{N} \multimap (\mathtt{x} + \text{フ}) \neq \overline{\mathtt{0}}$.

Definition 8
A candidate *of* base $\mathcal{T}+\mathcal{U}$*, where* $\mathcal{T},\mathcal{U}$ *are orthogonal tests, is any behaviour* $\mathbf{G}$ *such that* $\mathcal{T}\in\mathbf{G}$ *and* $\mathcal{U}\in\sim\mathbf{G}$*.*

Existential quantification is handled as follows:

Analytically: the *proof* of $\exists\mathbf{X}\mathbf{A}[\mathbf{X}]$ obtained from $\mathbf{A}[\mathbf{T}]$ makes use of a *witness* $\mathcal{T}+\mathcal{U}$, namely the base of the behaviour $\mathbf{T}$, seen as a candidate.

Synthetically: the *behaviour* $\exists\mathbf{X}\mathbf{A}[\mathbf{X}]$ is defined by:

$$\exists\mathbf{X}\mathbf{A}[\mathbf{X}] := \sim\sim(\bigcup_{\mathbf{T}}\mathbf{A}[\mathbf{T}]) \tag{7}$$

Our choice of witness is basically a simplification of what we proposed in [5]: since there is no hope of packing together the full $\mathbf{T},\sim\mathbf{T}$, we cannot avoid partiality (section 6.1 of [5]). Singling out elements $\mathcal{T}\in\mathbf{T}$ and $\mathcal{U}\in\sim\mathbf{T}$ makes it even more partial, but this partiality matters no more in the context of infinite behaviours. Incidentally observe that the existential case actually defines a behaviour: the practical joke of an empty orthogonal is avoided, since it contains the switching $[\![\,\overline{p_\alpha(\mathtt{mag}(x\mathtt{d}y)), p_{\tilde{\alpha}}(\mathtt{mag}(x\mathtt{d}y))}\,]\!]$ ([5], section 6.1)

which checks the orthogonality of the pillars $\mathcal{T},\mathcal{U}$ of the base.

Universal quantification is handled by a plain intersection:

$$\forall\mathbf{X}\mathbf{A}[\mathbf{X}] := \bigcap_{\mathbf{T}}\mathbf{A}[\mathbf{T}] \tag{8}$$

Definitions (7) and (8) follow the original pattern used for system $\mathbb{F}$ ([3]) which now yields a justification of second order principles.

5.3 Recurrence

The principle of recurrence, a.k.a. mathematical induction is usually written:

$$\forall\mathtt{y}\ (A[\mathtt{y}] \multimap A[\mathtt{y}+\overline{1}]) \Rightarrow (A[\overline{0}] \multimap A[\mathtt{x}]) \tag{9}$$

with two defects, one being that it is an axiom, i.e., an ukase proceeding from the Sky, the other being that it is a schema, i.e., a sort of meta-axiom introduced in order to circumvent the fatwa against second order. Replacing the schema with the obvious second order definition makes it possible to *define* natural numbers, Dedekind style, as the smallest set containing zero and closed under successor:

$$\mathtt{x} \in \mathbb{N} \quad := \quad \forall X\,(\forall \mathtt{y}\,(X(\mathtt{y}) \multimap X(\mathtt{y} + \text{フ})) \Rightarrow (X(\overline{\mathtt{0}}) \multimap X(\mathtt{x}))) \qquad (10)$$

From which the implication $x \in \mathbb{N} \multimap (9)$ follows. A useful variant is obtained by applying (9) to $!A[x] \otimes (A[x] \multimap A[x])$, which yields:

$$\mathtt{x} \in \mathbb{N} \multimap \forall \mathtt{y}\,(A[\mathtt{y}] \Rightarrow A[\mathtt{y} + \text{フ}]) \Rightarrow (A[\overline{\mathtt{0}}] \Rightarrow A[\mathtt{x}]) \qquad (11)$$

The handling of quantification over predicates, here unary, is inspired from the propositional case. We should introduce a notion of parametric candidate. First by separating positive from negative occurrences. Typically, $\mathtt{x} \multimap \mathtt{x}$ should be written as $\mathtt{x}^- \multimap \mathtt{x}^+$ and later subject to the constraint $\mathtt{x}^- = \mathtt{x}^+$. In terms of parametric candidates, this means that we should consider doubly indexed families $\mathcal{G}_{m,n}$ $(m, n \in \mathbb{Z})$ of candidates enjoying:

$$m' \leq m, n \leq n' \Rightarrow \mathcal{G}_{m,n} \subset \mathcal{G}_{m',n'} \qquad (12)$$

i.e., covariant in n, contravariant in m; the negation will thus be covariant in m, contravariant in n. They should also be provided with a base $\mathcal{T} + \mathcal{U}$ such that, for all $m, n \in \mathbb{Z}$, $\mathcal{T}(\sim\overline{\mathtt{m}}, \overline{\mathtt{n}}) + \mathcal{U}(\overline{\mathtt{m}}, \sim\overline{\mathtt{n}})$ is a base of $\mathcal{G}_{m,n}$. Typically, if $\mathcal{G}_{m,n} := \overline{\mathtt{m}} \multimap \overline{\mathtt{n}}$, $\mathcal{T}$ and $\mathcal{U}$ stand for switchings of $\parr$ and its dual $\otimes$, so that $\mathcal{T}(\sim\overline{\mathtt{m}}, \overline{\mathtt{n}})$ and $\mathcal{U}(\overline{\mathtt{m}}, \sim\overline{\mathtt{n}})$ are switchings of $\overline{\mathtt{m}} \multimap \overline{\mathtt{n}}$ and $\overline{\mathtt{m}} \otimes \sim\overline{\mathtt{n}}$.

5.4 Product

The product $(\mathtt{t} \cdot \mathtt{x}) \simeq \mathtt{y}$ is defined by a quantification over binary predicates:

$$\forall X\,(\forall \mathtt{x}\,\forall \mathtt{y}\,(X(\mathtt{x}, \mathtt{y}) \multimap X(\mathtt{x} + \text{フ}, \mathtt{y} + \mathtt{t})) \Rightarrow (X(\overline{\mathtt{0}}, \overline{\mathtt{0}}) \multimap X(\mathtt{x}, \mathtt{y}))) \qquad (13)$$

We can then prove the existence of the product by recurrence on $\mathtt{x}$:

$$\mathtt{x} \in \mathbb{N} \;\Rightarrow\; \exists \mathtt{y}\,(!(\mathtt{y} \in \mathbb{N}) \otimes (\mathtt{t} \cdot \mathtt{x}) \simeq \mathtt{y}) \qquad (14)$$

The predicate $(\mathtt{t} \cdot \mathtt{x}) \simeq \mathtt{y}$ is handled by means of a sort of graph recurrence, which amounts at replacing the variable X of definition (13) with a specific binary predicate $A[\mathtt{x}, \mathtt{y}]$. For instance, with $A[\mathtt{x}, \mathtt{y}] := \mathtt{x} \in \mathbb{N}$, we get:

$$(\mathtt{t} \cdot \mathtt{x}) \simeq \mathtt{y} \;\multimap\; x \in \mathbb{N} \qquad (15)$$

Consider $A[\mathtt{x}, \mathtt{y}, \mathtt{x}', \mathtt{y}'] := \mathtt{x} = \mathtt{x}' \multimap \mathtt{y} = \mathtt{y}'$; the following are provable:

$$
\begin{aligned}
& A[\overline{\mathtt{0}}, \overline{\mathtt{0}}, \overline{\mathtt{0}}, \overline{\mathtt{0}}] && (16)\\
& A[\overline{\mathtt{0}}, \overline{\mathtt{0}}, \mathtt{x}' + \text{フ}, \mathtt{y}' + \mathtt{t}] && (17)\\
& A[\mathtt{x} + \text{フ}, \mathtt{y} + \mathtt{t}, \overline{\mathtt{0}}, \overline{\mathtt{0}}] && (18)\\
A[\mathtt{x}, \mathtt{y}, \mathtt{x}', \mathtt{y}'] \multimap\ & A[\mathtt{x} + \text{フ}, \mathtt{y} + \mathtt{t}, \mathtt{x}' + \text{フ}, \mathtt{y}' + \mathtt{t}] && (19)
\end{aligned}
$$

A "graph recurrence" w.r.t. $\mathtt{x}', \mathtt{y}'$, using (16) and (17) yields

$$(\mathtt{t} \cdot \mathtt{x}') \simeq \mathtt{y}' \multimap A[\overline{\mathtt{0}}, \overline{\mathtt{0}}, \mathtt{x}', \mathtt{y}'] \tag{20}$$

Another "graph recurrence" w.r.t. $\mathtt{x}', \mathtt{y}'$, using (18) and (19) yields

$$(\mathtt{t} \cdot \mathtt{x}') \simeq \mathtt{y}' \multimap (A[\mathtt{x}, \mathtt{y}, \mathtt{x}', \mathtt{y}'] \multimap A[\mathtt{x} + \mathtt{t}, \mathtt{y} + \mathtt{t}, \mathtt{x}', \mathtt{y}']) \tag{21}$$

And a graph recurrence w.r.t. $\mathtt{x}, \mathtt{y}$, using (20) and (21) yields:

$$(\mathtt{t} \cdot \mathtt{x}) \simeq \mathtt{y} \multimap ((\mathtt{t} \cdot \mathtt{x}') \simeq \mathtt{y}' \multimap A[\mathtt{x}, \mathtt{y}, \mathtt{x}', \mathtt{y}']) \tag{22}$$

in other terms, the unicity of the product.

Incidentally, the fact that the product is only second order definable may be related to the typical second order feature known as the incompleteness of arithmetic, which relies on an encoding making a heavy use of the product.

A L'usine, again

Usine vs. usage, it's Church vs Curry. The existentialist approach of Curry is quite respected by the notion of *behaviour*. On the other hand, the essentialism inherent to the typing à la Church leads to systems and must be deeply modified. I propose the following :

A type (Church style) is a (finite) battery of tests.

This is compatible with polymorphism : several batteries may be used to "type" the same design. However, there is a problem, the definition seeming not to apply in full generality, because of the absence of finite preorthogonals.

I propose the following solution : instead of a preorthognal of behaviour **P**, a preorthogonal of a *sub-behaviour* of **P**. Orthogonality to such a preorthogonal need not be necessary, it is only *sufficient*. On the other hand, it may remain finite, hence the possibility of a battery of tests. Let us give two examples.

A.1 Identity

The principle $A \vdash A$, the identity "axiom", poses a problem of finiteness. It is tested through simultaneous tests, for $\sim A$ and A, which is possible in certain cases, but doesn't work in general.

Let us suppose that A correspond to general behaviour $\mathbf{A}$, with not finiteness restriction. I still know how to justify $\vdash \sim A, A$ because it is *sufficient* to test it against generic pairs, that of a test for $\sim \mathbf{A}$ and for $\mathbf{A}$ with no reference to $\mathbf{A}$ which therefore takes the moral value of a variable X. We know that the cases :

$$A = \text{フ} \qquad \sim A = \text{フ} \tag{23}$$

$$A = \text{フ} \otimes \text{フ} \qquad \sim A = \text{フ} \parr \text{フ} \tag{24}$$

$$A = \text{フ} \parr \text{フ} \qquad \sim A = \text{フ} \otimes \text{フ} \tag{25}$$

do suffice. They force the presence of a star $[\![\sim A(x), A(x)]\!]$, if I abusively denote the respective locations of $\sim A$ and A by $\sim A(x), A(x)$. This implies in turn that the said star does belong to the behaviour $\vdash \sim A, A$.

These tests are not necessary : if $A = B \otimes C$ and $\vdash \sim A, A$ has been obtained by "η-expansion" from $\vdash \sim B, B$ and $\vdash \sim C, C$, they fail.

We just witnessed the native *sufficient* testing. Remark that its two parts are not independent : if A is tested as $\text{フ} \otimes \text{フ}$, $\sim A$ must be tested as $\text{フ} \parr \text{フ}$.

A.2 Existence

Existence can be informally reduced to a very peculiar case, that of the implication $\forall X\, A \vdash A[T/X]$, in other terms $\vdash \exists X \sim A, A[T/X]$. We must test $(\mathcal{T}, \mathcal{T}', \mathcal{P})$ where $\mathcal{P}$ is the identity $\vdash \sim A[T/X], A[T/X]$. We just observed that this identity possesses a sufficient battery of tests. We conclude that $\mathcal{P}$ belongs to the behaviour associated with $\vdash \sim A[T/X], A[T/X]$.

In order to show that $(\mathcal{T}, \mathcal{T}', \mathcal{P})$ is in the behaviour (7) corresponding to $\vdash \exists X \sim A, A[T/X]$, we imitate the argument given for system $\mathbb{F}$: the comprehension principle shows that the behaviour associated with T is a set, and we use the "substitution lemma" of [3].

A.3 Finitism

The finitistic pattern advocated by Hilbert is correct provided we throw in some necessary distinctions. Three layers are needed :

Usine : typing à la Church, but system-free. It enables us to *predict* what proofs will do.

Usage : typing à la Curry, naturally system-free. A *behavourial* approach, what proofs are actually doing.

Adequation : cut-elimination, so to speak. It shows the accuracy of the prediction of l'usine.

The first two layers are the opposite sides of finitism, of a completely different nature. The first person who (vaguely) understood the distinction was Lewis Carroll (1893), who mistook l'usine for the "meta" of l'usage and built a ludicrous wild goose chase which he dared compare with Zeno's paradox. Indeed, by replacing a cut on A with a a cut on $A \Rightarrow A$, next a cut on $(A \Rightarrow A) \Rightarrow (A \Rightarrow A)$, etc. Carroll's Achilles is constantly fleeing away from the Tortoise… no wonder it never reaches him.

The third layer, adequation, does not belong to finitism : it is where an infinitary, eventually set-theoretic, arguement must be thrown in… with no possible way of bypassing it.

References

[1] T. Coquand and G. Huet. **The calculus of constructions**. *Information and Computation*, 76:95 – 120, 1988.

[2] V. Danos and L. Regnier. **The structure of multiplicatives**. *Archive for Mathematical Logic*, 28:181 – 203, 1989.

[3] J.-Y. Girard. **Une extension de l'interprétation fonctionnelle de Gödel à l'analyse et son application à l'élimination des coupures dans l'analyse et la théorie des types**. In Fenstad, editor, *Proceedings of the* 2^{nd} *Scandinavian Logic Symposium*, pages 63 – 92, Amsterdam, 1971. North-Holland.

[4] J.-Y. Girard. **Transcendental syntax 1: deterministic case**. *Mathematical Structures in Computer Science*, pages 1–23, 2015. *Computing with lambda-terms. A special issue dedicated to Corrado Böhm for his 90th birthday.*

[5] J.-Y. Girard. **Transcendental syntax 3: equality**. *Logical Methods in Computer Science*, 2016. *Special issue dedicated to Pierre-Louis Curien for his 60th birthday.*

[6] J-Y Girard, A. Scedrov, and P. Scott. **Bounded linear logic : a modular approach to polynomial time computability**. *Theoretical Computer Science*, 97:1 – 66, 1992.

[7] R. Harper, F. Honsell, and G. Plotkin. **A framework for defining logics**. *LFCS report series, Edinburgh*, 162, 1991.

[8] G. Kreisel. **Mathematical logic**. In T. L. Saaty, editor, *Lectures in modern mathematics, vol III*, pages 99 – 105. Wiley *&* Sons, New York, 1965.