Math. Struct. in Comp. Science (2001), *vol.* 11, *pp.* 301–506. Printed in the United Kingdom

Locus Solum:
From the rules of logic to the logic of rules

JEAN-YVES GIRARD

Institut de Mathématiques de Luminy, UPR 9016 – **CNRS**
163, Avenue de Luminy, Case 930, F-13288 Marseille Cedex 09
Email: `girard@iml.univ-mrs.fr`

Received 3 July 2000; revised 5 April 2001

Go back to An-fang, the Peace Square at An-Fang, the Beginning Place at An-Fang, where all things start (. . .) An-Fang was near a city, the only living city with a pre-atomic name (. . .) The headquarters of the People Programmer was at An-Fang, and there the mistake happened: A ruby trembled. Two tourmaline nets failed to rectify the laser beam. A diamond noted the error. Both the error and the correction went into the general computer.
Cordwainer Smith *The Dead Lady of Clown Town,* 1964.

Contents

https://doi.org/10.1017/S096012950100336X

1. Introduction: alternative titles

A purely interactive approach to logic

'Interactive' might suggest yet-one-more-game-semantics. However, the material presented here is neither syntax nor semantics, and, moreover the word *purely* suggests a distance from the mere idea of game: there is no rule – or no referee, if you prefer – as in real life. And *logic*, without an 's', stands for what should be the most natural thing in nature – something too often presented as the most artificial one.

This monograph ends with a dictionary discussing these issues: a sort of final introduction, since one can only be introduced to known material. For instance if you go to DIALECTICS you will learn about the word

Ludics

which is the real alternative title, the very name of this new area.

The novelty of ludics is conveyed by our title

Locus Solum

after the book by Raymond Roussel, *Locus Solus*, that is, 'solitary place'. *Locus Solum* means something like

Only the location matters

since the results presented here establish the pregnancy of location, the *locus*, in logic. As you will see, the irruption of the *locus* in no way weakens or dilutes logical principles: they just become different, more harmonious, and stronger. Moreover, the logic-we-used-to-know-and-love is still present, but it now gets a specific name, *spiritual logic*: ludics created spiritual logic in the same way that Brouwer created classical logic and Luther created catholicism.

This monograph has been conceived as the project of giving reasonable foundations to logic, on the largest possible grounds, but without the notorious reductionist connotation usually attached to the word 'foundations'. *Locus Solum* would like to be the common playground of logic, independent of systems, syntaxes, not to mention ideologies. But wideness of scope is nothing here but the reward of sharpness of concern: I investigate the multiple aspects of a single artifact, the *design*. Designs are not like those syntax-versus-semantics whores that one can reshape according to the humour of the day: one cannot tamper with them, period. But what one can achieve with them, once their main properties – separation, associativity, stability – have been understood, is out of proportion with their seemingly banal definition.

Finally, this monograph has been written during the year 2000, the year of commemorative frenzy. So let me review the last century, from the viewpoint of logical foundations.

1900-1930, the time of illusions: Naive foundational programs, such as Hilbert's, refuted by Gödel's theorem.

1930-1970, the time of codings: Consistency proofs, monstrous ordinal notations, *ad hoc* codings, a sort of voluntary bureaucratic self-punishment.

1970-2000, the time of categories: From the mid sixties the renewal of natural deduction, the Curry–Howard isomorphism, denotational semantics, system $\mathbb{F}$ . . . promoted (with the decisive input of computer science) an approach in which the objects looked natural and reasonably free from foundational anguish.

Proof-theory started as a justification of the rules of logic, as they were given to us, that is, classical logic. The rules became in turn an object of study, inducing their own logic, which is not the original (classical) one. Intuitionistic logic and the later linear logic, not to mention ludics, are part of this logic of rules – hence our subtitle:

From the rules of logic to the logic of rules.

Remerciements

Je voudrais remercier tous ceux qui m'ont aidé dans cette interminable rédaction; il m'est impossible de mentionner tout le monde, sauf à dévaloriser le remerciement et à rendre l'oubli involontaire encore plus injuste. Je me contenterai de remercier dans l'ordre chronologique :

— Michele Abrusci et le groupe de logique de Roma III ; c'est à l'occasion de mon séjour à Rome en Février 2000 que j'ai entrepris cette rédaction. Une rédaction pendant l'Année Sainte, ce pourrait expliquer la prégnance du $\Delta\alpha\acute{\iota}\mu\omega\nu$ et de la *Fides* dans le texte, sans parler du *Mystère de l'Incarnation.*

— Les logiciens de l'Equipe de Logique de la Programmation de Marseille-IML, et tout spécialement : Luca Paolini pour son stage de DEA qui suggérait des améliorations importantes de rédaction ; Claudia Faggian pour ses suggestions quant à la normalisation des desseins et sa contribution à la théorie de la parcimonie.

— Giuseppe Longo qui m'a offert le bon espace d'expression dans MSCS, et ceci dans un temps très court.

— Olivier Danvy, Thomas Ehrhard, Giuseppe Longo, Laurent Regnier, Simonetta Ronchi, Philip Scott qui ont complété le dictionnaire sur des points où je me sentais peu sûr de moi. Ces rares articles supplémentaires sont attribués à leur auteur ; on ne s'étonnera

donc pas qu'ils s'auto-mentionnent, puisque je leur ai demandé d'écrire sur leur propre travail.

— Et, *last but not least*, Pierre-Louis Curien qui a relu plusieurs fois le texte, et dont les commentaires et suggestions ont été des plus précieux.

2. Designs

We first construct the concrete objects that logic is made of. These objects – called *designs* – play the role devoted to proofs, λ-terms, *etc.* in the usual syntax, and to functions, cliques, *etc.* in denotational semantics, and even to classical models to a reasonable extent.

2.1. *Locations*

The material in this section is identical (up to terminology) to similar material in Girard (2000).

2.1.1. *Biases and loci* The basic analytical artifacts (designs) are located 'somewhere'. We shall therefore build a system of *locations*. A design $\mathfrak{D}$ roughly represents a cut-free proof of some formula A, in which all logical information has been erased: only locations are kept. If A has been located at the empty sequence $\langle\rangle$, then any formula occurring in the proof is a subformula of A: it has a precise location in the subformula tree of A. Without loss of generality, we can assume that the number of immediate subformulas of any formula B is at most denumerable, and we can name the various immediate subformulas of B as $B0, B1, B2, \ldots$ The natural numbers $0, 1, 2, \ldots$ that distinguish the immediate subformulas of B are called *biases*; when we translate syntax, their choice is strictly arbitrary – there is no tricky coding at work. For instance, the three immediate[†] subformulas of the formula $A = ((P^\perp \oplus Q^\perp) \otimes R^\perp)$ of Subsection 2.2.1, p. 307, will be distinguished by the biases $3, 4, 7$, but we could equally have chosen $9, 6, 22$. An address (or *locus*) in the tree is therefore a sequence of biases. We will now formalise these definitions.

Definition 1. (Loci) A *bias* is a natural number (notation $i, j, k \ldots$). A *ramification* is a finite set of biases, (notation $I, J, K, \ldots$). A *locus*, or address, is a sequence $\langle i_1, \ldots, i_n\rangle$ of biases (notation $\sigma, \tau, \upsilon, \xi \ldots$). The *parity* of a locus is defined as the parity of its length n.

Hence $\langle 3, 3, 8\rangle$ is odd, whereas its immediate sublocus $\langle 3, 3, 8, 0\rangle$ is even. I shall follow the usual conventions for concatenation, in particular $\sigma * i$ instead of $\sigma * \langle i\rangle$; sometimes (especially to save space in figures) I shall even use use σi. $\sigma * \tau$ is called a *sublocus* of σ, and is *strict* when $\tau \neq \langle\rangle$, and *immediate* when $\tau = \langle i\rangle$. If two *loci* are incomparable, they are *disjoint*, that is, they have no common sublocus. Finally, the notation $\xi * I$ is short for $\{\xi * i; i \in I\}$.

Ramifications are needed because of multiplicative rules, which involve two (in fact, through focalisation any finite number of) subformulas at the same time. Since subloci

[†] The notion of immediate subformula is combined with focalisation: here two steps are allowed, up to the change of polarity.

correspond to subformulas, and a proof proceeds from its conclusion[†], a locus occurs 'before' its subloci (time relation). When two *loci* are incomparable, their relation is *spatial*, that is, they are completely independent.

The following expressions will be introduced later: *directory* for a set of ramifications, and *reservoir* for a set of biases, finite or infinite.

2.1.2. *Pitchforks* Since we are representing cut-free proofs, we must consider some sort of *sequents*. Although linear logic systematically used one-sided sequents $\vdash \Gamma$, one can as well use two-sided sequents made of positive formulas (typically, $\vdash M, P, Q$, with M negative, P, Q positive is replaced with $M^\perp \vdash P, Q$, which only contains positive formulas). Focalisation makes it possible to restrict to sequents with at most one formula on the left; these are in fact the familiar intuitionistic sequents, with left and right exchanged since it is more natural to work with positive formulas[‡]. *Pitchforks* correspond to what remains of sequents when we only remember locations.

Definition 2. (Pitchforks) A *pitchfork* is an expression $\Xi \vdash \Lambda$ where:

Incomparability: Ξ and Λ are finite sets of pairwise disjoint *loci*; in particular, a locus in Ξ and a locus in Λ are disjoint.

Handle and tines: Ξ contains at most one *locus*, the *handle*, the *loci* in Λ being the *tines*.

Each pitchfork receives a *polarity*: a handleless pitchfork (a 'comb') is *positive*, a pitchfork with a handle being *negative*; in particular the empty pitchfork is positive. A pitchfork is *atomic* when it contains exactly one address, that is, is of the form $\vdash \xi$ or $\xi \vdash$.

In practice, pitchforks always satisfy the following additional condition:

Paritarism: The *loci* in Λ have the same parity, which is opposite to the parity of the handle (if this makes sense).

Typically, all pitchforks occurring in a design of atomic base are *paritary*: this is because all rules involved in designs preserve paritarism. Paritary pitchforks receive a parity, namely that of the tines and/or the parity opposite to the handle, with only one ambiguous case, the empty pitchfork, which receives both parities. Paritarism plays no special role in the theory, but can be used as a sort of *type-checker*.

From the interactive standpoint, the parities correspond to two (essentially isomorphic) players, `Even` and `Odd`. Rules (see Definition 3, p. 308) correspond to *moves* of the players: for instance a proper rule of even focus is a move of `Even`. In the paritary case, the *focus* of the rule is on the left (that is, of different parity) or on the right (same parity) depending on whether the concluding pitchfork has a handle (negative) or not (positive). Polarity can therefore be seen as *relative parity*: *negative* means 'different parity' and *positive* means 'same parity', or in game-theoretic terms 'you start' and 'I start'. For a more detailed discusion, see Subsection 4.2.3.

[†] Cut-elimination also proceeds from the conclusion. By the way, this is the only possibility, especially in the absence of axioms, which will be the case in ludics.

[‡] Remember that intuitionistic logic is mainly based on negative operations, $\Rightarrow, \wedge, \forall$.

Pitchforks are handled with the usual conventions of sequent calculus: Γ, Δ is short for the (disjoint) union $\Gamma \cup \Delta$, and singletons are replaced with their unique element, so that $\xi \vdash \Gamma, \Delta, \lambda$ is short for $\{\xi\} \vdash \Gamma \cup \Delta \cup \{\lambda\}$, and this implicitly means that the sets $\{\xi\}, \Gamma, \Delta$ and $\{\lambda\}$ are disjoint.

2.2. *Designs as dessins*

2.2.1. *From proofs to designs* Designs are proofs written in sequent calculus, or rather the locative structure of a proof in sequent calculus, a sort of proof in 'pitchfork calculus'. To understand how things work, let us take a concrete example, namely the positive formula $A = ((P^\perp \oplus Q^\perp) \otimes R^\perp)$, where P, Q, R are positive (so that the immediate subformulas of A modulo focalisation are $P^\perp, Q^\perp, R^\perp$). The rules for A are

$$\frac{\vdash \Lambda, P, R \qquad \vdash \Lambda, Q, R}{A \vdash \Lambda}\;{\scriptstyle (A \vdash \{\{P,R\},\{Q,R\}\})}$$

$$\frac{P \vdash \Gamma \qquad R \vdash \Delta}{\vdash \Gamma, \Delta, A}\;{\scriptstyle (\vdash A,\{P,R\})}$$

$$\frac{Q \vdash \Gamma \qquad R \vdash \Delta}{\vdash \Gamma, \Delta, A}\;{\scriptstyle (\vdash A,\{Q,R\})}$$

The right rules are obtained by combining a right Tensor-rule with one of the two possible right Plus-rules and negation, yielding two possibilities distinguished as $(\vdash A, \{P, R\})$ and $(\vdash A, \{Q, R\})$; *the* left rule is obtained by combining *the* Par-rule with *the* With-rule and negation. The rule is written $(A \vdash \{\{P, R\}, \{Q, R\}\})$ in order to stress the existence of two premises: one involving P, R; the other involving Q, R.

The basic idea of designs is to forget everything but locations. So, assume that the *locus* of A is ξ and that P, Q, R, respectively, correspond to the (distinct!) biases $3, 4, 7$. Then we can rewrite our rules as[†]:

$$\frac{\vdash \Lambda, \xi 3, \xi 7 \qquad \vdash \Lambda, \xi 4, \xi 7}{\xi \vdash \Lambda}\;{\scriptstyle (\xi \vdash \{\{3,7\},\{4,7\}\})}$$

$$\frac{\xi 3 \vdash \Gamma \qquad \xi 7 \vdash \Delta}{\vdash \Gamma, \Delta, \xi}\;{\scriptstyle (\vdash \xi,\{3,7\})}$$

$$\frac{\xi 4 \vdash \Gamma \qquad \xi 7 \vdash \Delta}{\vdash \Gamma, \Delta, \xi}\;{\scriptstyle (\vdash \xi,\{4,7\})}$$

The example clearly shows how to translate essential parts of a proof, but says nothing as to the identity axiom. Indeed, the identity axiom $A \vdash A$ is reduced by means of η-expansion to atomic identity axioms $p \vdash p$, but, since the proofs we have in mind are in fact universally quantified with respect to p, the η-expansions never stop. To cut a long

[†] Γ, Δ, Λ are no longer made of formulas, but of *loci*; to save space we have used the notation $\xi 3$ instead of $\xi * 3$ *etc.*

https://doi.org/10.1017/S096012950100336X

story short, no specific rule is needed to construct the *faxes* that correspond to identity axioms.

The examples of pitchfork rules just shown are indeed enough to represent any known logical inference. But this is not enough, we need a Joker, that is, a rule that one can always apply. This rule, called *Daimon*, has the value of an axiom, an arbitrary one. It is restricted to positive pitchforks (simply because its negative version is of no use). Syntactically it can be seen as a mistake of logic (I cannot prove $\vdash \Gamma$, so I admit it!), but this is a 'good' mistake, which is absolutely needed to get enough 'proofs' (enough designs) with good properties.

We are now in position to define designs, but the real definition is rather difficult to grasp. In French, the two words *dessein* (= design, plot *etc.*) and *dessin* (= drawing, picture *etc.*) sound the same, and this is why:

- We first define a slightly incorrect notion, designs-dessins, which are basically the 'proof-trees' constructed with our rules.
- Since these dessins contain irrelevant information, we then give the real definition of designs-desseins.
- In practice, no particular care is needed, that is, results will usually be stated for the correct notion of design-dessein and proved with the help of the friendlier designs-dessins.

2.2.2. *Dessins*

Definition 3. (Designs-dessins) A *design* is a proof-tree made of pitchforks. The last pitchfork of the design is called the conclusion or *base*. Each pitchfork occurring in the design is the conclusion of a unique *rule* among those given below. The possible rules fall into three categories:

Daimon:

$$\frac{}{\vdash \Lambda}\,\maltese \tag{1}$$

Positive rule: I is a ramification, for $i \in I$ the Λ_i are pairwise disjoint and included in Λ: one can apply the rule (finite, one premise for each $i \in I$)

$$\frac{\ldots \xi * i \vdash \Lambda_i \ldots}{\vdash \Lambda, \xi}\,{\scriptstyle (\vdash \xi, I)} \tag{2}$$

Negative rule: $\mathscr{N}$ is a set of ramifications, the *directory of the rule*, for all $I \in \mathscr{N}$, $\Lambda_I \subset \Lambda$: one can apply the rule (perhaps infinite, one premise for each $I \in \mathscr{N}$)

$$\frac{\ldots \vdash \Lambda_I, \xi * I \ldots}{\xi \vdash \Lambda}\,{\scriptstyle (\xi \vdash \mathscr{N})} \tag{3}$$

ξ is called the *focus* of the rules $(\vdash \xi, I)$ and $(\xi \vdash \mathscr{N})$. Since I is a ramification and $\mathscr{N}$ is a set of ramifications, the symbol $\vdash$ in the name of the rule is redundant, except for the case $I = \mathscr{N} = \emptyset$, and we can omit it without creating impossible confusions: we simply write (ξ, I) or $(\xi, \mathscr{N})$. The three rules discovered in Subsection 2.2.1, p. 307, are therefore written $(\xi, \{3, 7\})$, $(\xi, \{4, 7\})$ and $(\xi, \{\{3, 7\}, \{4, 7\}\})$.

Our rules are in fact a combination of the usual linear rules and weakening – strict linearity would force $\Lambda = \bigcup \Lambda_i$ in the positive case, and $\Lambda = \Lambda_I$ in the negative case. If such *mistakes* have been allowed, this must be ascribed to the mysteries of interactivity: typically, the Separation Theorem (Theorem 2, p. 328) and the Projection Theorem (Theorem 19, p. 354) make heavy use of 'weakening'.

Remark 1. No assumption of finiteness, well-foundedness or recursivity is made, so designs can be badly infinite. However, infinite designs can naturally be approximated by means of finite designs: anticipating our designs-as-desseins, the inclusion $\mathfrak{D} \subset \mathfrak{E}$ means that $\mathfrak{E}$ has been obtained by 'adding' extra premises to the negative rules of $\mathfrak{D}$, hence any design $\mathfrak{E}$ is the directed union of finite designs $\mathfrak{E}_i$, obtained by restricting all negative rules of $\mathfrak{E}$ to finite directories, all but a finite number of them being empty.

2.2.3. *Some basic designs*

Example 1. The design

$$\frac{}{\vdash \Lambda}\,✠ \tag{4}$$

is called the *daimon* and written $\mathfrak{Dai}$.

When $\vdash \Sigma, \xi$ is positive, the rules ✠ and $(\vdash \xi, \emptyset)$ have quite the same premises, but they must be distinguished. The rule ✠, which has no focus, is considered as a positive rule, but an *improper* one. We shall use the expression *proper* design to mean a design distinct from the daimon, which, in the case of a positive base, means that the first rule of the design is proper. By the way observe that there is only one design of base $\vdash$, namely the daimon.

What follows is the most basic example of a design, namely the *fax*.

Example 2. If ξ and ξ' are disjoint, one defines $\mathfrak{Fax}_{\xi,\xi'}$,, a design of base $\xi \vdash \xi'$.

$$\dfrac{\ldots \qquad \dfrac{\begin{array}{c}\vdots\ \mathfrak{Fax}_{\xi' * i, \xi * i}\\ \ldots\ \xi' * i \vdash \xi * i \ldots\end{array}}{\vdash \xi', \xi * I}\,{\scriptstyle (\xi', I)} \qquad \ldots}{\xi \vdash \xi'}\,{\scriptstyle (\xi, \wp_f(\mathbb{N}))} \tag{5}$$

The *fax* relates two occurrences of the same formula A, one located in ξ, the other located in ξ'. Since the two A are intended to be subformulas of the implication $A \multimap A$, one occurring positively, the other occurring negatively, one understands that the locations are usually of opposite parities[†]; it is more correct to rename the A to the right as A'. The actual meaning of the fax comes from the fact that it behaves with respect to normalisation as the identity function (see Example 11) of A, better, the isomorphism

[†] However, if we form a *net* of faxes, paritarism cannot be ensured, see Remark 3, p. 318.

between A and A', corresponding to the ‘delocation’ that exchanges $\xi * \sigma$ with $\xi' * \sigma$. In terms of down-to-Earth game semantics, the fax is nothing but the ludic version of the familiar copycat strategy. The expression ‘fax’ emphasises the *delocation.*

We have said that the fax is the identity axiom, and this is true whether A is atomic or not. But for non-atomic A, typically the formula of Subsection 2.2.1, p. 307, we have another possibility, namely η-expansion. If we translate the proof of $A \vdash A'$ ending with η-expansion, we instead obtain the *pseudo-fax*, as in the following example.

Example 3.

$$\dfrac{\dfrac{\overset{\vdots\ \mathfrak{Fax}_{\xi'3,\xi3}}{\xi'3 \vdash \xi3} \qquad \overset{\vdots\ \mathfrak{Fax}_{\xi'7,\xi7}}{\xi'7 \vdash \xi7}}{\vdash \xi', \xi\{3,7\}}\,(\xi',\{3,7\}) \qquad \dfrac{\overset{\vdots\ \mathfrak{Fax}_{\xi'4,\xi4}}{\xi'4 \vdash \xi4} \qquad \overset{\vdots\ \mathfrak{Fax}_{\xi'7,\xi7}}{\xi'7 \vdash \xi7}}{\vdash \xi', \xi\{4,7\}}\,(\xi',\{4,7\})}{\xi \vdash \xi'}\,(\xi,\{\{3,7\},\{4,7\}\})$$

This design differs from the general fax only in the last rule, where the finite directory $\{\{3,7\},\{4,7\}\}$ replaces the *full* $\wp_f(\mathbb{N})$, that is, most premises of the last rule have been severed; in terms of *desseins*, see below, the pseudo-fax is a subset of the fax. In terms of syntax, fax and pseudo-fax both correspond to η-expansions of the identity axioms. The difference is that the pseudo-fax is relative to a specific formula, whereas the fax is generic, that is, works for any formula.

Whether they are the same or they are different, more precisely *in which sense* they are or are not the same belongs to the theory of *subtyping* and *incarnation*, which is to be developed later.

Definition 4. If the pitchfork $\Xi \vdash \Lambda$ occurs in the design $\mathfrak{D}$, the subtree $\mathfrak{E}$ induced by $\mathfrak{D}$ ‘above’ $\Xi \vdash \Lambda$ is a design of base $\Xi \vdash \Lambda$, which we call a *subdesign* of $\mathfrak{D}$.

Typically, the designs $\mathfrak{Fax}_{\xi'*i,\xi*i}$ are subdesigns of $\mathfrak{Fax}_{\xi,\xi'}$. The notion of subdesign has nothing to do with inclusion or *precedence*.

2.3. *Designs as desseins*

2.3.1. *Introduction to desseins* Observe that there is (with dessins) a problem, namely that the name of a positive rule $(\vdash \xi, I)$ does not tell us in which way the context splits. You could say: ‘let us mention it in the name of the rule’, and by the way this was the solution taken in Girard (2000). Unfortunately, there is no way to recognise this splitting interactively, that is, by duality[†].

Let us consider the following example.

[†] Concretely, the Separation Theorem (Theorem 2, p. 328) works for desseins, not for dessins.

Example 4.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\sigma 2 \vdash \xi 30}\,(\sigma 2,\emptyset)}{\vdash \xi 30, \sigma}\,(\sigma,\{2\}) \qquad \dfrac{}{\vdash \xi 3\{0,5\}, \sigma}\,\maltese}{\xi 3 \vdash \sigma}\,(\xi 3,\{\{0\},\{0,5\}\}) \qquad \dfrac{\dfrac{}{\vdash \tau}\,\maltese}{\xi 7 \vdash \tau}\,(\xi 7,\{\emptyset\})}{\vdash \xi, \sigma, \tau}\,(\xi,\{3,7\})$$

The last positive rule ‘gives’ σ to 3 and τ to 7. Can we dispatch the context differently? Of course, since the positive rule $(\vdash \sigma, \{2\})$ is performed on the branch indexed by 3, one could hardly give σ to 7, but what about τ, who has been given to 7, and which is passive? Indeed the dessins in the following examples offer equally valid dispatchings of the context – τ is, so to speak, between 3 and 7, perhaps nowhere.

Example 5.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\sigma 2 \vdash \xi 30, \tau}\,(\sigma 2,\emptyset)}{\vdash \xi 30, \sigma, \tau}\,(\sigma,\{2\}) \qquad \dfrac{}{\vdash \xi 3\{0,5\}, \sigma, \tau}\,\maltese}{\xi 3 \vdash \sigma, \tau}\,(\xi 3,\{\{0\},\{0,5\}\}) \qquad \dfrac{\dfrac{}{\vdash}\,\maltese}{\xi 7 \vdash}\,(\xi 7,\{\emptyset\})}{\vdash \xi, \sigma, \tau}\,(\xi,\{3,7\})$$

and

Example 6.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\sigma 2 \vdash \xi 30}\,(\sigma 2,\emptyset)}{\vdash \xi 30, \sigma}\,(\sigma,\{2\}) \qquad \dfrac{}{\vdash \xi 3\{0,5\}, \sigma}\,\maltese}{\xi 3 \vdash \sigma}\,(\xi 3,\{\{0\},\{0,5\}\}) \qquad \dfrac{\dfrac{}{\vdash}\,\maltese}{\xi 7 \vdash}\,(\xi 7,\{\emptyset\})}{\vdash \xi, \sigma, \tau}\,(\xi,\{3,7\})$$

In short, the splitting of the context is a convenient decorative (dessin = picture) feature, *which only makes sense for those* loci *in the context that are used as foci of positive rules.* What is amusing is that this use destroys the *locus* (it is no longer there), what we call *location post mortem*. So designs should be considered up to this basic ambiguity: this leads to desseins.

Let us consider the dessin of Example 4, p. 311: we replace the tree with another one, roughly of the same shape, as in the following example.

Example 7.

$$\dfrac{\dfrac{(\sigma, \{2\})}{(\xi 3, \{0\})} \quad \dfrac{\maltese}{(\xi 3, \{0, 5\})} \quad \dfrac{\maltese}{(\xi 7, \emptyset)}}{(\xi, \{3, 7\})} \tag{6}$$

The precise idea is to replace in the tree any negative pitchfork $\xi \vdash \Lambda$ that is the conclusion of rule $(\xi, \mathcal{N})$ with several copies, one for each $I \in \mathcal{N}$, renamed as (ξ, I): this essentially makes negative branchings occur one step below. Each positive pitchfork is renamed after the rule with that conclusion.

If we interpret Examples 5, p. 311, and 6, p. 311, we get the same tree, which is the common underlying dessein. As an illustration, let us try with the fax (Example 2, p. 309) as follows.

Example 8.

$$\cdots \quad \dfrac{\dfrac{\cdots \qquad \dfrac{\dfrac{\begin{gathered}\vdots\\ \cdots\ (\xi ij,K)\ \cdots\end{gathered}}{(\xi i,J)}\,{\scriptstyle j\in J,K\in\wp_f(\mathbb{N})}}{(\xi' i,J)} \qquad \cdots}{(\xi',I)}\,{\scriptstyle i\in I,J\in\wp_f(\mathbb{N})}}{(\xi,I)} \quad \cdots_{\ I\in\wp_f(\mathbb{N})} \tag{7}$$

This dessein is not a tree, since it has no root. Observe that the repetitive pattern is even more conspicuous than in the dessin. Let us now look at the pseudo-fax of Example 3, p. 310.

Example 9.

$$\dfrac{\dfrac{\cdots \quad \dfrac{\dfrac{\begin{gathered}\vdots\\ \cdots\ (\xi 3j,K)\ \cdots\end{gathered}}{(\xi 3,J)}}{(\xi' 3,J)} \quad \dfrac{\dfrac{\begin{gathered}\vdots\\ \cdots\ (\xi 7j,K)\ \cdots\end{gathered}}{(\xi 7,J)}}{(\xi' 7,J)} \quad \cdots}{(\xi',\{3,7\})}}{(\xi,\{3,7\})} \qquad \dfrac{\dfrac{\cdots \quad \dfrac{\dfrac{\begin{gathered}\vdots\\ \cdots\ (\xi 4j,K)\ \cdots\end{gathered}}{(\xi 4,J)}}{(\xi' 4,J)} \quad \dfrac{\dfrac{\begin{gathered}\vdots\\ \cdots\ (\xi 7j,K)\ \cdots\end{gathered}}{(\xi 7,J)}}{(\xi' 7,J)} \quad \cdots}{(\xi',\{4,7\})}}{(\xi,\{4,7\})}$$

This dessein is just the subset of the previous one restricted to the initial values $I = \{3,7\}, I = \{4,7\}$.

Remark 2. The irrelevance of τ is related to *weakening*, that is, to the rule

$$\frac{\vdash \Gamma}{\vdash \Gamma, A} \tag{8}$$

in which the context A (the locus of τ) is destroyed (if we look at the proofs from conclusion to premises, or in terms of normalisation). The reader may ask why weakening is not forbidden from the very start.

1 The forbidding of weakening is only necessary in the absence of polarisation, for want of linearity, see the critical pair of Lafont, which does not make sense in a polarised universe.

2 Weakening is essential in the proof of internal completeness of the tensor product.

https://doi.org/10.1017/S096012950100336X

The question whether or not weakening is now a fully respectable rule is beyond the scope of this monograph.

However, the irrelevance of τ cannot be fully ascribed to weakening. Typically in the logical proof

$$\dfrac{\dfrac{}{\vdash \top, A} \qquad \dfrac{}{\vdash \top}}{\vdash \top \otimes \top, A} \tag{9}$$

which does not use weakening, A could as well be given to the right premise:

$$\dfrac{\dfrac{}{\vdash \top} \qquad \dfrac{}{\vdash \top, A}}{\vdash \top \otimes \top, A} \tag{10}$$

2.3.2. *Desseins* In what follows we fix a pitchfork $\Upsilon \vdash \Lambda$, which we shall call the *base*.

Definition 5. (Chronicles) A *proper action* is a triple (ϵ, ξ, I) consisting of a *polarity* $\epsilon = \pm 1$, a *focus* ξ (that is, a locus) and a ramification I. There is also an *improper* action, namely the *Daimon* $(+1, ✠)$, positive.

In what follows 'proper action' means an action κ whose focus ξ is a sublocus of a (necessarily unique) *locus* of the base $\sigma \in \Lambda$ (respectively, $\sigma \in \Upsilon$), and polarity is $+1$ if the parities of ξ, σ are the same (respectively, opposite), -1 if the parities of ξ, σ are opposite (respectively, the same)[†]. In practice we never indicate the polarity of an action and simply write (ξ, I)[‡] or ✠.

A *chronicle of base* $\Upsilon \vdash \Lambda$ is a non-empty sequence of actions $\langle \kappa_0, \ldots, \kappa_n \rangle$ such that:

Alternation: The polarity of κ_p is equal to the polarity of the base for p even, and different for p odd.

Daimon: For $p < n$ κ_p is proper, that is, $\kappa_p = (\xi_p, I_p)$.

Negative actions: A negative focus ξ_p must be chosen either in Υ (then $p = 0$ and the base is negative), or in $\xi_{p-1} * I_{p-1}$.

Positive actions: A positive focus ξ_p must be chosen either in Λ, or in one of the $\xi_q * I_q$, where (ξ_q, I_q) is one of the previous negative actions, that is, $q < p$ and $p - q$ odd.

Destruction of foci: Focuses are pairwise distinct, that is, cannot be reused.

A chronicle $\langle \kappa_0, \ldots, \kappa_n \rangle$ is said to be *proper* or *improper* depending on κ_n.

We use the Gothic letters $\mathfrak{c}, \mathfrak{d}, \mathfrak{e} \ldots$ to denote chronicles. We use the expression *subchronicle* to denote the restriction of a chronicle.

Definition 6. (Coherence) The chronicles $\mathfrak{c}, \mathfrak{c}'$ are *coherent* when:

Comparability: Either one extends the other, or they first differ on negative actions, that is, $\mathfrak{c} = \mathfrak{d} * \kappa * \mathfrak{e}$, $\mathfrak{c}' = \mathfrak{d} * \kappa' * \mathfrak{e}'$, with $\kappa \neq \kappa'$ negative.

Propagation: In case $\mathfrak{c}, \mathfrak{c}'$ first differ on κ, κ' with *distinct* foci, then all ulterior foci (that is, belonging to $\mathfrak{e}, \mathfrak{e}'$) are distinct.

[†] When the base is paritary, the polarity of the action is therefore the relative parity of ξ and the base.

[‡] This notation is ambiguous in the case of *cut-nets*, since the same action may occur twice: once positively and once negatively.

Definition 7. (Designs-desseins) A *design* of base (or *conclusion*) $\Upsilon \vdash \Lambda$ is a set $\mathfrak{D}$ of chronicles of base $\Upsilon \vdash \Lambda$ such that:

Arborescence: $\mathfrak{D}$ is closed under restriction, in other words, it is a forest.
Coherence: The chronicles of $\mathfrak{D}$ are pairwise coherent.
Positivity: If $\mathfrak{c} \in \mathfrak{D}$ has no extension in $\mathfrak{D}$, its last action is positive.
Totality: If the base is positive, $\mathfrak{D}$ is non-empty.

A design is *positive* or *negative* according to its base.

Let us explain the definitions:

— The notion of chronicle exactly corresponds to a branch in the forest associated with a dessin, as in Examples 7, p. 311, 8, p. 312, 9, p. 312. The conditions eliminate sequences that would not come from an actual dessin.

— Comparability (in the definition of coherence relation) says that whenever $\mathfrak{c} * \kappa, \mathfrak{c} * \kappa'$ belong to the same design, with κ positive, then $\kappa = \kappa'$ (typically, a negative action is followed by a unique positive action). In particular, a positive design (which is non-empty by totality) has a well-defined first action, that is, is a tree.

— Propagation is the most subtle property: returning to our examples, for instance Example 7, p. 311 – after the first action $(\xi, \{3,7\})$, a ternary branching occurs, with a choice between three negative actions, $(\xi * 3, \{0\})$, $(\xi * 3, \{0,5\})$, $(\xi * 7, \emptyset)$, say κ_i for $i = 1,2,3$. κ_1 and κ_2, which have the same negative focus, come from the same negative rule, κ_3 comes from a distinct negative rule. κ_1 and κ_3 have distinct foci because they are performed above two different premises of a positive rule, *on which the context splits*: returning to the original dessin 4, p. 311, we see that an action on σ has been performed at the extreme left that is, above κ_1; should we allow an action on σ above κ_3, we would be unable to split the context between 3 and 7.

— The condition of positivity can be understood as follows: take a chronicle $\mathfrak{c}$ ending with a negative action, then it corresponds to one of the premises of a negative rule, and the conclusion of a positive rule, which corresponds to the ‘next’ action. The argument does not work when $\mathfrak{c}$ ends with a positive action, which may be a daimon, or the premise of a negative rule of the form $(\xi, \emptyset)$.

— Totality states the existence of a first action when the base is positive; it is therefore a technical variant of positivity, made necessary by the fact that we do not recognise the empty sequence as a chronicle – see also the discussion in Subsection 2.4, p. 315.

We still need to know how we should associate a dessin with a dessein. The simplest way is to do it in two steps:

— First we reconstruct a fake dessin, fake in the sense that we systematically recopy the contexts (in the positive rules, $\Lambda_i = \Lambda$, in the negative rules $\Lambda_I = \Lambda$). This offers no difficulty.

— Then we remove from any pitchfork $\Upsilon \vdash \Lambda$ all the *loci* that are not used as foci ‘above’ the pitchfork. Observe that we no longer get $\Lambda_I = \Lambda$, and, moreover, the condition of propagation implies that, in case of a positive rule, the Λ_i are pairwise disjoint.

Observe that the splitting of the context is not decidable, since it depends on the eventual behaviour of the design. However, we have established the existence of a *minimal*[†] dessin associated to a design. It is impossible to identify a dessein with its minimal dessin: the assignment does not commute to normalisation, to incarnation, *etc*., hence we can hardly call this assignment canonical.

2.4. *Partial designs*

If[‡] we drop the totality condition, that is, accept the empty set as a design of a given positive base, we can speak of a *partial* design, the usual designs being therefore styled *total*. The unique quite partial design is denoted by the symbol $\mathfrak{Fid}$ and is called the *pseudo-design*; its nickname is *Faith*[§], and it plays the role of the familiar unsolvable $\Delta\Delta$ of λ-calculus. It corresponds to the idea of a positive pitchfork (the conclusion) with no rule above, which we can write:

$$\frac{}{\vdash \Lambda}\,\Omega \tag{11}$$

A natural generalisation would be to allow more partial elements simply by removing the condition of positivity, that is, an arbitrary positive pitchfork – not only the conclusion – could be the conclusion of no rule at all. But this generalisation is fake, for if it is not a conclusion, such a pitchfork is the premise of index I of some negative rule $(\xi, \mathcal{N})$, and the same effect would be achieved by severing the premise, that is, by replacing $\mathcal{N}$ with $\mathcal{N} - \{I\}$.

We can also use the generalisation the other way around: let us formally introduce another improper positive rule, Ω, such that $\mathfrak{Fid}$ is the positive partial design ending with the 'rule' Ω . We can now decide that all negative branchings are full (that is, with directory $\mathcal{N} = \wp_f(\mathbb{N})$): we need to complete the branchings, but any missing branch can be justified by the new rule. In this presentation, the status of the negative rules is simplified (it becomes quite invertible, since what is above $\xi \vdash \Lambda$ is always the rule $(\xi, \wp_f(\mathbb{N}))$). The totality condition just means that Ω cannot be the last rule of a design. This variant, which basically complicates the description of designs-dessins, has no practical interest; but it is theoretically important, since it suggests a certain symmetry between the two 'improper' rules. Indeed the ordering of designs, see Section 4.1.2, p. 328, can be synthesised by the formula (20):

$$\Omega \preceq (\xi, I) \preceq \maltese$$

However, there is an essential difference: as we shall see, Ω naturally occurs as an infinite loop in normalisation, whereas $\maltese$ corresponds to immediate termination. Imagine that we want to find the first rule of a positive design through normalisation (see below), and let

[†] In the sense of the most parsimonious assignment of contexts.

[‡] In Chapter 8 we shall revisit the expression 'partial design' so as to make it relative to a given bihaviour. Then Ω appears as the unique 'design' that is quite partial, irrespective of the bihaviour.

[§] The name is intended to convey its deep operational meaning: as we shall see, the partial design is the formal result of a diverging normalisation, which is morally infinite. If the result never shows up, all that remains is the faith that we had at each moment that we should eventually get something.

us admit that we are only interested in a proper rule. The answers Ω and ✠ are therefore 'bugs', but of different natures:

Too late: ✠ says that you will not get your proper rule, but at least you know this.

Please wait: Ω occurs in the case that your answer never comes, but how can you know it? Not only you will not get your proper rule, but you may endlessly expect it.

By the way, we see that the proposal of writing Ω as a rule is non-effective in the data because of the undecidability of the halting problem. But this is no different from the situation in λ-calculus. One can either see Ω as the reification of the absence of information (this is a plain set-theoretic interpretation), or we can see it dynamically, that is, imagine that a design is 'growing', *streamlike*: typically, normalisation is a process in which we try to get positive rules (ξ, I) or ✠ above some pitchfork, and as long as this information is not obtained, we can simply write Ω.

On the other hand, nothing like ✠ exists in the logical literature. The delicate point is that most interpretations confuse them[†] (standard denotational semantics identifies both designs $\mathfrak{Fid}$ and $\mathfrak{Dai}$ with $\emptyset$); since everything lies in between $\mathfrak{Fid}$ and $\mathfrak{Dai}$, we have just stepped on the major flaw of traditional denotational semantics.

3. Normalisation

Designs can be combined together so as to form *cut-nets*, for which a deterministic *normalisation* procedure will be defined below. Similar procedures are essential in any reasonable[‡] sequent calculus (including variants such as natural deduction, proof-nets, lambda-calculi, *etc.*). The deep meaning of normalisation is *composition* of morphisms (if we build a category out of proofs) or of strategies (if we take a game-theoretic viewpoint).

The normalisation of designs is deterministic but not necessarily converging, just as in pure λ-calculus. There are two ways to present it, which are of equal interest:

Dessins: This is an analytical, step-by-step description, easy to grasp if one keeps in mind that a design is basically a sequent calculus proof.

Desseins: This is a global, synthetic, description, which is essential in the proof of the main theorems of Section 4.1.

3.1. *Normalisation of dessins*

3.1.1. *Motivations* Designs have been constructed by imitation of cut-free proofs. We should now introduce designs with cuts – but wait, what is cut? Indeed there is no cut-rule (and no identity axiom, that is symmetry, is it not?). A cut is just a coincidence handle/tine between the bases of two designs.

[†] However, some denotational models of programming languages (see, for example Cartwright *et al.* (1994)) include explicit error values. The relation of daimon to errors could be something like: not only give up, but also "confess your sins".

[‡] Unfortunately, modern *Tartuffes* have been lately producing sequent calculi with no cut-elimination – much like cars without engines.

To understand what is going on, let us start with our example of Subsection 2.2, p. 307: a cut between $\vdash \Gamma, \Delta, A$ and $A \vdash \Lambda$ is easily reduced – when both proofs end with rules for A – into two cuts on P and R when $(\vdash A, \{P, R\})$ has been used, or two cuts on Q and R when $(\vdash A, \{Q, R\})$ has been used. The same thing happens for designs, but for the fact that P, Q, R are now called $\xi 3, \xi 4, \xi 7 \ldots$ Is this just a bureaucratic transformation? Not at all, since in the former case, the formula A was forcing a symmetry between left and right rules[†]: the negative rule uses $\mathscr{N} = \{\{3,7\},\{4,7\}\}$, which quite matches the two positive rules, which use either $\{3,7\}$ or $\{4,7\}$. But now that we are only left with locations, this matching is no longer obvious! In logical terms, when we are cutting, the formulas are no longer forced to be the same (even if they share the same location, the location induces no restriction as to the possible rules). Concretely, it may happen that the last rules of the two designs are $(\xi, \mathscr{N})$ and (ξ, I), but that $I \notin \mathscr{N}$: in that case normalisation fails.

There is one more positive rule, namely the daimon; since a $\Delta\alpha\acute{\iota}\mu\omega\nu$ is almighty, the normalisation succeeds, but the normalised proof ends with… the daimon.

As we know, cut-elimination involves many commutations of rules. They are, as usual, reliable and boring, but with two novelties:

— The discipline of polarities destroys all possible conflicts, that is, normalisation is strictly deterministic.

— In the important case of a closed net, typically a cut between $\vdash \xi$ and $\xi \vdash$, no commutation is at work.

3.1.2. *Cut-nets*

Definition 8. (Cut-nets) A *cut-net* is a non-empty finite set $\mathfrak{R} = \{\mathfrak{D}_0, \ldots, \mathfrak{D}_n\}$ of designs of respective bases $\Xi_p \vdash \Lambda_p$ such that:

Disjunction: The *loci* occurring in the bases are pairwise disjoint or equal.

Cuts: Every *locus* occurs in at most two bases. In such a case, one occurrence is a handle and the other is a tine. Such a shared *locus* is called a *cut*.

Connected/acyclic: The graph whose vertices are the $\Xi_p \vdash \Lambda_p$ and whose edges are the cuts is connected and acyclic.

A design is a particular case of cut-net, just let $n = 0$.

Since $\sharp(\mathit{components}) - \sharp(\mathit{cycles}) = \sharp(\mathit{vertices}) - \sharp(\mathit{edges})$, the connected-acyclic condition may be restated as 'connected and n cuts', or 'acyclic and n cuts'. Since n handles are consumed in cuts, there is at most one handle that is not a cut, and we can form a pitchfork with the uncut *loci*, the *conclusion* or *base* of the cut-net; a cut-net whose base is the empty pitchfork is said to be *closed*. The unique design $\mathfrak{D}_i$ whose base is positive or is negative with as handle the uncut handle of $\mathfrak{R}$ is the *main design* of the cut-net, its base is the *main pitchfork* of the cut-net and its last rule is the *main rule* of the net.

[†] The discovery of the introduction/elimination symmetry in the framework of natural deduction is the essential achievement of Prawitz (see Prawitz (1965) – if you can find it).

https://doi.org/10.1017/S096012950100336X

Remark 3. $\mathfrak{R}$ is *paritary* when it is made of paritary designs and, moreover, its base is paritary. For instance, two paritary designs of bases $\xi \vdash \sigma$ and $\sigma \vdash \tau$ yield a non-paritary net of base $\xi \vdash \tau$. By the way, observe that such a non-paritary net cannot be made paritary by means of an *ad hoc* delocation. Since paritarism is useful as a type-checker, one is advised to stick, as much as possible, to paritary nets.

Observe that, since the conditions only mention the bases of the designs $\mathfrak{D}_0, \ldots, \mathfrak{D}_n$, certain perversions of the definition are possible:

- — Replacing the design $\mathfrak{D}_0$ with a cut-net $\mathfrak{R}_0 = \{\mathfrak{E}_0, \ldots, \mathfrak{E}_m\}$ with the same base. Is this not very close to the cut-net $\{\mathfrak{E}_0, \ldots, \mathfrak{E}_m, \mathfrak{D}_1, \ldots, \mathfrak{D}_n\}$?
- — Allowing the designs of $\mathfrak{R}$ to be partial; concretely, this means the possibility of using $\mathfrak{Fid}$, which is – since it is positive – the main 'design' of such a *partial* cut-net.

These two possibilities are very important, since they can be combined to formulate the *associativity of normalisation*, which is one of the major analytical theorems.

3.1.3. *Normalisation: closed case* The cut-elimination procedure, called *normalisation*, is a strictly deterministic procedure, which replaces a cut-net $\mathfrak{R}$ with a design of the same base, its *normal form* $[\![\mathfrak{R}]\!]$; the process may diverge, that is, yield no result, or equivalently the partial design $\mathfrak{Fid}$. This (possibly) infinite – but *locally finite* – process proceeds from the conclusion of the net. We define it using designs-dessins, but this is just to be friendly. Since the case of a closed cut-net is by far the most interesting, we start with this case. The general case follows.

Definition 9. (Closed normalisation) Let $\mathfrak{R}$ be a closed cut-net. Then the main design, say $\mathfrak{D}$, is positive, with main rule κ, and three cases occur:

Daimon: κ is the daimon $\maltese$[†]. Then the net normalises into the unique design with an empty base, the daimon: $[\![\mathfrak{R}]\!] = \mathfrak{Dai}$. This case is the only case of termination for a closed net.

Immediate failure: κ is (ξ, I). Hence ξ is a cut, and it occurs as the handle of another design $\mathfrak{E}$, the *adjoint* design of the net, whose last rule is necessarily of the form $(\xi, \mathcal{N})$. If $I \notin \mathcal{N}$, then normalisation fails.

Conversion: This is as above, but $I \in \mathcal{N}$. For $i \in I$, let $\mathfrak{D}_i$ be the subdesign of $\mathfrak{D}$ whose conclusion is the premise of index i ($\xi * i \vdash \ldots$) of (ξ, I), and let $\mathfrak{E}'$ be the subdesign of $\mathfrak{E}$ induced by the premise of index I ($\vdash \xi * I, \ldots$) of the rule $(\xi, \mathcal{N})$. Define $\mathfrak{S}$ by replacing $\mathfrak{D}, \mathfrak{E}$ by the $\mathfrak{D}_i, \mathfrak{E}'$; since $\mathfrak{S}$ is not necessarily connected, let $\mathfrak{S}'$ be the connected component of $\mathfrak{E}'$ in $\mathfrak{S}$. Then $[\![\mathfrak{R}]\!] = [\![\mathfrak{S}']\!]$.

In conversion, the replacement of $\mathfrak{S}$ by $\mathfrak{S}'$ is due to the fact that our rules may involve some 'weakenings': some *loci* occurring in the conclusions of the main and adjoint designs of the net may disappear. When $\mathfrak{S}$ is not connected, we keep, in fact, the connected component of $\mathfrak{E}'$: as usual weakening induces erasings. This (small) problem disappears with *desseins*.

[†] If $n = 0$, the base is empty and the net is already a daimon.

https://doi.org/10.1017/S096012950100336X

The normal form, when it exists, is necessarily the daimon $\mathfrak{Dai}$. However, the normalisation may diverge, either by immediate failure, or because of an infinite series of conversions. We use the notation $[\![\mathfrak{R}]\!] = \mathfrak{Fid}$ to denote the result of a diverging normalisation: this convention becomes very useful if we extend normalisation to partial cut-nets. This induces another case:

Faith: If the main design $\mathfrak{D}$ is $\mathfrak{Fid}$, the normal form is $[\![\mathfrak{R}]\!] = \mathfrak{Fid}$.

Although this is a convenient convention, we should never forget that there is no effective way to determine whether or not a normal form is total, remember that $\mathfrak{Fid}$ behaves like the Ω of λ-calculus and the symbol u of recursion theory[†].

3.1.4. *Normalisation: open case* Let us now consider the general case, where the base is not supposed to be closed. There are now, besides the three extant cases, two new possibilities:

Positive commutation: The net is positive, with main rule (ξ, I), but ξ is not a cut. Let $\mathfrak{D}_i$ be as in the case of conversion above, and define $\mathfrak{R}'$ by replacing $\mathfrak{D}$ with the $\mathfrak{D}_i$. $\mathfrak{R}'$ splits into several connected components, and each $\mathfrak{D}_i$ lies in a component $\mathfrak{R}_i$, which is a net, and the $\mathfrak{R}_i$ are pairwise distinct. Let the $\mathfrak{E}_i$ be the respective normal forms of the $\mathfrak{R}_i$ (these normal forms do exist, since the $\mathfrak{R}_i$ are negative, see below). The normal form of $\mathfrak{R}$ is the design whose last rule is (ξ, I) and which proceeds with $\mathfrak{E}_i$ above the premise of index i, that is,

$$[\![\mathfrak{R}]\!] = \frac{\cdots \quad [\![\mathfrak{R}_i]\!] \quad \cdots}{\vdash \Lambda, \xi}\,{\scriptstyle(\xi, I)} \tag{12}$$

Negative commutation: The net is negative, with main rule $(\xi, \mathcal{N})$ for its main design $\mathfrak{D}$. For $I \in \mathcal{N}$, let $\mathfrak{D}_I$ be the subdesign of $\mathfrak{D}$ above the premise of index I of the last rule, and let us replace $\mathfrak{D}$ with $\mathfrak{D}_I$ in $\mathfrak{R}$; and let $\mathfrak{R}_I$ be the connected component of $\mathfrak{D}_I$ (again we do not directly get a net, because of weakening). Let $\mathcal{N}'$ be the subset of $\mathcal{N}$ made of those I for which $\mathfrak{R}_I$ has a normal form $\mathfrak{E}_I$. The normal form of $\mathfrak{R}$ is defined as the net ending with $(\xi, \mathcal{N}')$ and which proceeds with $\mathfrak{E}_I$ above the premise of index I, that is,

$$[\![\mathfrak{R}]\!] = \frac{\cdots \quad [\![\mathfrak{R}_I]\!] \quad \cdots}{\xi \vdash \Lambda}\,{\scriptstyle(\xi, \mathcal{N}')} \tag{13}$$

In other words, the positive commutation recopies the last rule and then proceeds separately above each premise. The negative commutation does the same, but some premises may disappear. Observe that, since negative commutation is the only possibility for a negative cut-net, all negative nets have a normal form: the worse that may happen is that $\mathcal{N}' = \emptyset$.

The fact that $\mathcal{N}$ is replaced by $\mathcal{N}'$ in the negative commutation should be understood dynamically: $\mathcal{N}'$ is growing (as soon as we get the last rule of $\mathfrak{D}_I$, we know that $I \in \mathcal{N}'$). But after all, did we know $\mathcal{N}$ that well? One may very well imagine (especially if we

[†] In recursion theory, the symbol u stands for ‘undefined’, and equation $f(a) \simeq b$ means that if one side is defined, then the other side is defined and equal.

https://doi.org/10.1017/S096012950100336X

think of associativity of normalisation, see below) that $\mathcal{N}$ is built through a normalisation process, and so is growing. The missing premises of a negative rule are just those premises that get stalled forever. If we stick to this intuition, we must admit that what we called finite failure is definitely not finite: we are waiting for a premise of index I, but it will arrive tomorrow, or the day after, *etc*. This is to say that the symbol Ω is really for infinite loops.

By the way, we can define normalisation of partial nets, exactly as above. With the convention of full negative branchings, the negative case becomes slightly simpler, since $\mathfrak{E}_I$ is defined for all $I \in \wp_f(\mathbb{N})$, but of course many of the $\mathfrak{E}_I$ may be equal to $\mathfrak{Fid}$; the case 'immediate failure' is replaced with 'faith'.

Let us experiment with normalisation. First we define an interesting guy.

Definition 10. The *negative daimon* $\mathfrak{Dai}^-$ of base $\xi \vdash \Lambda$ is the design

$$\dfrac{\ldots \quad \dfrac{}{\vdash \xi * I, \Lambda}\,\maltese \quad \ldots}{\xi \vdash \Lambda}\,{\scriptstyle (\xi,\wp_f(\mathbb{N}))} \tag{14}$$

The usual daimon is sometimes called the positive daimon, and written $\mathfrak{Dai}^+$.

Example 10. Any cut between a positive daimon and a design normalises into a positive daimon: this is obvious from the definitions. Every cut between a negative daimon of base $\xi \vdash \Lambda$ and a design $\mathfrak{D}$ of base $\Upsilon \vdash \Sigma$ normalises:

- If $\Upsilon = \upsilon$ and $\upsilon \in \Lambda$, the normal form is a negative daimon.
- If $\xi \in \Sigma$ and $\Upsilon = \emptyset$, the normal form is a positive daimon when $\mathfrak{D}$ is a daimon or $\mathfrak{D}$ ends with a rule focusing on ξ. If $\mathfrak{D}$ ends with (σ, I) with $\sigma \neq \xi$, then the normal form is a design ending with (σ, I) with negative daimons above each premise of the rule:

$$\dfrac{\ldots \quad \dfrac{\ldots \quad \dfrac{}{\vdash \sigma * i * I, \Lambda, \Sigma - \xi}\,\maltese \quad \ldots}{\sigma * i \vdash \Lambda, \Sigma - \xi}\,{\scriptstyle (\sigma * i,\wp_f(\mathbb{N}))} \quad \ldots}{\vdash \Lambda, \Sigma - \xi}\,{\scriptstyle (\sigma, I)} \tag{15}$$

- If $\xi \in \Sigma$ and $\Upsilon = \upsilon$, let $(\upsilon, \mathcal{N})$ be the last rule of $\mathfrak{D}$; then the normal form is a restricted negative daimon

$$\dfrac{\ldots \quad \dfrac{}{\vdash \upsilon * I, \Lambda, \Sigma - \xi}\,\maltese \quad \ldots}{\upsilon \vdash \Lambda, \Sigma - \xi}\,{\scriptstyle (\upsilon,\mathcal{N})} \tag{16}$$

In other words, a daimon is able to cope with all situations!

Example 11. Now consider the fax $\mathfrak{Fax}$ of base $\xi \vdash \xi'$, and observe that:

- A cut with $\mathfrak{D}$ of base $\vdash \xi$ ending with rule (ξ, I) will normalise into a design of base $\vdash \xi'$ ending with rule (ξ', I).
- A cut with $\mathfrak{E}$ of base $\xi' \vdash$ ending with rule $(\xi', \mathcal{N})$ will normalise into a design of base $\xi \vdash$ ending with rule $(\xi, \mathcal{N})$.

From this it is easy to show that, in the first case, the normal form is in fact the delocation $\rho(\mathfrak{D})$ of $\mathfrak{D}$, that is, the design obtained by systematically replacing ξ with ξ'. In the second case, the normal form is the delocation $\rho^{-1}(\mathfrak{E})$ of $\mathfrak{E}$. More generally, a cut with the fax on ξ is normalised by replacing ξ with ξ', and a cut with the fax on ξ' is normalised by replacing ξ' with ξ. In particular, the cut between $\mathfrak{Fax}_{\xi,\xi'}$ and $\mathfrak{Fax}_{\xi',\xi''}$ normalises into $\mathfrak{Fax}_{\xi,\xi''}$[†]. General delocations and the associated faxes are introduced in Example 18, p. 338. To summarise, let us write the equations:

$$[\![\mathfrak{Fax}, \mathfrak{D}]\!] = \rho(\mathfrak{D}) \tag{17}$$

$$[\![\mathfrak{Fax}, \mathfrak{E}]\!] = \rho^{-1}(\mathfrak{E}) \tag{18}$$

$$[\![\mathfrak{Fax}, \mathfrak{D}, \mathfrak{E}]\!] = [\![\mathfrak{D}, \rho^{-1}(\mathfrak{E})]\!] = [\![\rho(\mathfrak{D}), \mathfrak{E}]\!] \tag{19}$$

It may be of interest to see what happens with the pseudo-fax of Example 3, p. 310. It basically normalises like the fax, but only keeps actions (ξ, I) (or (ξ', I)) with $I \in \{\{3,7\},\{4,7\}\}$. Concretely, this means that a cut with a negative design whose last rule is $(\xi', \mathscr{N})$ normalises as a design with last rule $(\xi, \mathscr{N} \cap \{\{3,7\},\{4,7\}\})$, and that a cut with a positive design whose last rule is (ξ, I) normalises (that is, converges) exactly when $I \in \{\{3,7\},\{4,7\}\}$, in which case the normal form is the delocation of $\mathfrak{D}$.

3.1.5. *Discussion* Normalisation is not that obvious to grasp, I hope that the following comments may help.

1 Normalisation roughly imitates the usual syntactical normalisation in **MALL**, that is, multiplicative-additive linear logic, modulo focalisation: it just forgets everything about the formulas, remembering only the loci. This first approximation is not sufficient, since:

- The correct analogy would rather be an affine version of **MALL**, in which weakening is allowed on positive sequents.
- There is a new rule, the daimon, which is a sort of arbitrary axiom; but a cut with such an axiom can only be normalised by means of another axiom.
- It is in general impossible to get consistent 'decorations' of a cut-net, for example, when the designs $\mathfrak{D}, \mathfrak{E}$ of bases $\vdash \xi$ and $\xi \vdash \lambda$ 'come from' proofs of $\vdash A$ and $B \vdash C$ with $A \neq B$. Typically, think of $A = A' \oplus A''$, $B = A' \oplus B'$; if A'' were equal to B'', then we would know what to do… because *the syntactical constraints* force a matching between the possible last rules (ramifications) of both proofs, see the discussion on page 317: when the ramifications do not match, that is, when $I \notin \mathscr{N}$, the process of normalisation diverges.

2 The closed case is the most important one: the closure principle will anyway enable us to reduce normalisation to this case. When I normalise a closed net $\{\mathfrak{D}, \mathfrak{E}\}$, the normalisation appears as:

[†] If both $\xi \vdash \xi'$ and $\xi' \vdash \xi''$ are paritary, then $\xi \vdash \xi''$ is not paritary.

https://doi.org/10.1017/S096012950100336X

- A sequence of conversions, corresponding to actions performed in $\mathfrak{D}, \mathfrak{E}$. In fact each step corresponds to a pair $\kappa, \widetilde{\kappa}$ of *opposite* actions, one in $\mathfrak{D}$, one in $\mathfrak{E}$. Of these two actions, the positive one is obviously *active*, since it discriminates among the possibilities offered on the other side. It is to be noted that each step of normalisation swaps the leader, if the active role (main rule) has been played by $\mathfrak{D}$ at stage n, then $\mathfrak{E}$ will be active at stage $n+1$.
- The normalisation can only end with a daimon, which is the only possible output; in particular, contrary to a superficial impression, a conversion involving an empty ramification is not a terminating step.
- An old misogynist proverb says: man proposes; woman disposes. In particular, the active side may propose a positive action κ whose opposite is not present in the passive side, this is immediate failure...
- This is not to be confused with infinite failure, which corresponds to an endless dialogue between the partners. General considerations about normalisation, especially associativity, force us to conceptually identify the two forms of failure, since $\mathfrak{D}, \mathfrak{E}$ may in turn be given dynamically, that is, as the result of normalisation, and the missing $\widetilde{\kappa}$ may be nothing but the result of a local divergence.

3 In the general (open) case, one may need the commutations. This is because the base of the normal form consists in the uncut loci, and we have to provide the adequate rules.

- These rules are provided anyway by $\mathfrak{D}, \mathfrak{E}$; more precisely they are to be found in those rules whose focus is not a sublocus of a cut.
- A way to think of this is that we are not trying to construct the full normal form $[\![\mathfrak{R}]\!]$, but only a finite branch – an arbitrary one – (that is, a chronicle) of the normal form. If the base is positive, we must find a first rule, daimon (termination) or a proper positive rule. In such a case, we must proceed with the premises $[\![\mathfrak{R}_i]\!]$, but since we are only interested in one branch, we shall only look at one of these premises.
- The process of building a chronicle is called a *dispute*, see Definition 15, p. 326; a dispute is a trip through the net that never visits the same locus twice.

3.2. *Normalisation of desseins*

3.2.1. *Cut-nets* Let us discuss normalisation as it should be defined, that is, in terms of *desseins*, not dessins. The problem is that there is no hint as to the splitting of contexts, and this causes a mess with conversion. For that reason we give a definition of cut-nets adapted to desseins.

Definition 11. (Cut-nets with desseins) A *cut-net* is a non-empty finite set $\mathfrak{R} = \{\mathfrak{D}_0, \ldots, \mathfrak{D}_n\}$ of designs of respective bases $\Xi_p \vdash \Lambda_p$ such that:

Disjunction: The *loci* occurring in the bases are pairwise disjoint or equal.

Cuts: A *locus* cannot appear in two handles; a *cut* is a *locus* that occurs once as a handle and at least once as a tine.

https://doi.org/10.1017/S096012950100336X

Connected/acyclic: For each cut ξ draw an edge between the pitchfork with handle ξ and *one of* the pitchforks with tine ξ (this is called a *switching*). For each switching the graph obtained must be connected and acyclic.

Propagation: If σ is a tine in both $\Xi_p \vdash \Lambda_p$ and $\Xi_q \vdash \Lambda_q$, and if actions of focus σ are performed in both of $\mathfrak{D}_p, \mathfrak{D}_q$, then $p = q$.

Example 12. A typical example would be a net $\{\mathfrak{D}, \mathfrak{D}', \mathfrak{D}''\}$ made of designs of respective bases $\xi \vdash \sigma$, $\xi' \vdash \sigma$, $\vdash \xi, \xi'$. Propagation tells us that focus σ cannot be used in both of $\mathfrak{D}, \mathfrak{D}'$.

The subtlety of the definition is that σ is eventually active in at most one of the $\mathfrak{D}_p$ (by propagation, a condition strictly analogous to the condition of Definition 6, p. 313), but we do not know which one, and hence we require the connected/acyclic condition for all possibilities that may occur later, that is, all switchings. The basic case of conversion is handled in a simple way: each design $\mathfrak{D}_i$ receives the full context, hence a *locus* of the context is now present in several pitchforks, but as a tine.

Remark 4. The most convenient representation of cut-nets as desseins is to see them as sets of chronicles, that is, $\mathfrak{R} = \mathfrak{D}_0 \cup \ldots \cup \mathfrak{D}_n$. But, due to the fact that some loci are both handles and tines, a focus may be used both positively and negatively: we must be pedantic and carefully distinguish between $\kappa = (\epsilon, \xi, I)$ and its *opposite* $\widetilde{\kappa} = (-\epsilon, \xi, I)$.

3.2.2. *Slices and mauls* We now give a precise definition of normalisation in terms of *desseins.*

Definition 12. (Slices) A *slice* is a design (more generally a cut-net) $\mathfrak{S}$ in which all negative rules are at most unary: if $\mathfrak{c} * (-1, \xi, I), \mathfrak{c} * (-1, \xi, I') \in \mathfrak{S}$, then $I = I'$. A slice of a design $\mathfrak{D}$ (more generally of a cut-net $\mathfrak{R}$) is any slice $\mathfrak{S} \subset \mathfrak{D}$ ($\mathfrak{S} \subset \mathfrak{R}$).

In a slice, every proper action occurs at most once, hence we can define an *arborescent*[†] order between the proper actions of a slice:

$$\kappa <_{\mathfrak{S}} \kappa' \text{ iff } \mathfrak{S} \text{ contains a chronicle } \mathfrak{c} * \kappa * \mathfrak{c}' * \kappa'.$$

We shall identify $\mathfrak{S}$ with its proper actions, equipped with the order $<_{\mathfrak{S}}$: the order enables one to recover the proper chronicles; the improper chronicles can be recovered from the requirement of totality, typically, if κ is maximal and negative, a daimon is performed just after κ. In what follows, we use the expression 'κ is a *hidden* action' (with respect to a given slice $\mathfrak{S}$) to mean that κ is proper and its focus ξ is a sublocus of a cut.

Definition 13. (Balance) A finite slice $\mathfrak{S}$ is *balanced* when the following holds:
If κ is hidden, then

$$\kappa \in \mathfrak{S} \;\Rightarrow\; \widetilde{\kappa} \in \mathfrak{S}$$

[†] Remember that a *forest* is an order in which every initial segment is a finite total order.

Definition 14. (Mauls) The *maul* of a balanced slice $\mathfrak{S}$ is obtained by identifying any action with its opposite – notation $(\pm 1, \xi, I)$ (we speak of a *neutral* action). This induces a quotient of the order $<_{\mathfrak{S}}$, which we write $\ll_{\mathfrak{S}}$.

Proposition 1. If $\mathfrak{S}$ is balanced, its maul $\ll_{\mathfrak{S}}$ is a forest.

Proof. Without loss of generality, we can assume that the finite slice $\mathfrak{S}$ does not use the daimon: for instance, if a daimon is performed just above $\vdash \Lambda$, choose a fresh locus τ[†] and add τ to the right of the appropriated pitchforks, in order to replace $\vdash \Lambda$ with $\vdash \Lambda, \tau$; instead of performing a daimon, perform a $(\tau, \emptyset)$. If the property holds for this modified slice $\mathfrak{S}$, it surely holds for $\mathfrak{S}$.

Let $\mathfrak{T}$ be a maximal balanced subforest of $\mathfrak{S}$ such that $\ll_{\mathfrak{T}}$ is an arborescent order. Let κ be minimal in $\mathfrak{S} - \mathfrak{T}$ with respect to $<_{\mathfrak{S}}$:

1 If κ is positive and *visible*, that is, not hidden, then $\mathfrak{U} = \mathfrak{T} \cup \{\kappa\}$ is still an order, moreover it is a forest, that is, the set $\{\kappa' \ll_{\mathfrak{U}} \kappa\}$ is a linear order.

2 If κ is negative and visible, then $\mathfrak{U} = \mathfrak{T} \cup \{\kappa\}$ is still a forest.

3 Assume that none of the previous cases apply. Then all possible κ are hidden. The idea is still to add κ, but, since we must identify κ with $\widetilde{\kappa}$, it would be wiser to look for a κ (not necessarily our initial choice) such that $\kappa, \widetilde{\kappa}$ are minimal in $\mathfrak{S} - \mathfrak{T}$ with respect to $<_{\mathfrak{S}}$... and after that, see what happens.

(a) If κ is positive and κ' (positive) stands just below $\widetilde{\kappa}$ with respect to $<_{\mathfrak{S}}$, then κ' is hidden and $\widetilde{\kappa'} <_{\mathfrak{S}} \kappa$. From this we conclude that $\widetilde{\kappa}$ is minimal too.

(b) If κ is negative, consider the smallest action $\kappa_1 \leqslant_{\mathfrak{S}} \widetilde{\kappa}$ that is not in $\mathfrak{T}$:

i If $\kappa_1 = \widetilde{\kappa}$ or κ_1 is positive, we are done: choose κ_1.

ii Otherwise, iterate the process with the negative κ_1, up to the moment we find the appropriate κ_n. We must, however, show that the process eventually ends (no loop). First observe that the κ_i cannot endlessly be minimal in $\mathfrak{S}$, for if κ_i is minimal, the handle of the base of κ_i is a tine of the base of κ_{i+1}, and a loop would contradict acyclicity: some κ_i must eventually be non-minimal, and it is an easy remark that the κ_j $(j > i)$ are non-minimal also. Assume that the construction runs forever, that is, that all κ_j are negative, and (for $j > i$) let $\kappa'_j \in \mathfrak{T}$ stands for the maximum (positive) action $<_{\mathfrak{S}} \kappa_j$; the focus of κ_j has been created by κ'_j and therefore $\widetilde{\kappa'_j} <_{\mathfrak{S}} \widetilde{\kappa}_j$, which forces $\widetilde{\kappa'_j} <_{\mathfrak{S}} \kappa_{j+1}$, and hence $\widetilde{\kappa'_j} \leqslant_{\mathfrak{S}} \kappa'_{j+1}$, which is indeed a strict inequality, since these actions are of opposite parities... But then $\kappa'_j \ll_{\mathfrak{S}} \kappa'_{j+1}$, and the sequence κ'_j is strictly increasing with respect to $\ll_{\mathfrak{S}}$ in the finite forest $\mathfrak{T}$.

Once the adequate κ has been found, let $\mathfrak{U} = \mathfrak{T} \cup \{\kappa, \widetilde{\kappa}\}$, and assume without loss of generality that κ is positive. κ is clearly maximal with respect to $\ll_{\mathfrak{U}}$, but we must also check that $\ll_{\mathfrak{U}}$ is still a forest, which amounts to showing that the union $\mathfrak{L} = \mathfrak{L}' \cup \mathfrak{L}''$ of the linear orders $\mathfrak{L}' = \{\kappa'; \kappa' \ll \kappa\}$, $\mathfrak{L}'' = \{\kappa''; \kappa'' \ll \widetilde{\kappa}\}$ is a linear order. But the maximal elements of $\mathfrak{L}', \mathfrak{L}''$ are the actions κ' (negative) and

[†] In the case of a closed net, it might be necessary to delocate first.

κ'' (positive), respectively, performed just before... and observe that necessarily $\widetilde{\kappa''} \leqslant_{\mathfrak{S}} \kappa'$. Hence $\mathfrak{L} = \mathfrak{L}'$, that is, only κ contributes to the gluing. □

3.2.3. *Pull-backs*

Theorem 1. (Pull-backs)

1 If $\mathfrak{S}$ is balanced, $\llbracket \mathfrak{S} \rrbracket$ consists of the visible actions of $\mathfrak{S}$, with the order induced by $\ll_{\mathfrak{S}}$.

2 Conversely, if $\mathfrak{R}$ is a net and $\mathfrak{S}$ is a finite slice of $\llbracket \mathfrak{R} \rrbracket$, there exists a *unique* balanced slice $\mathfrak{T} \subset \mathfrak{R}$, which is the *pull-back of* $\mathfrak{S}$ *along* $\mathfrak{R}$, such that $\mathfrak{S} = \llbracket \mathfrak{T} \rrbracket$.

Proof. We will only sketch the proof. As before, we get rid of daimons by encoding them by means of *ad hoc* positive actions.

1 Take a maximal branch in the maul $\ll_{\mathfrak{S}}$, and interpret the previous proof as an inductive construction of the branch. Observe that

(a) Case 1 can only be followed by case 2; the branch must end with a case 1.

(b) Case 2 can only be followed by cases 1 or 3.

(c) Case 3 can only be followed by cases 2 or 3.

From this it is easy to conclude that what remains of the branch after removal of the hidden actions is in fact a chronicle[†]. One has therefore constructed a design with the same base as $\mathfrak{S}$. It should now be time to check that this design is actually equal to $\llbracket \mathfrak{S} \rrbracket$ – but this is almost obvious: if we were normalising $\mathfrak{S}$ as a dessin, cases 1, 2 and 3 would correspond to positive commutation, negative commutation and conversion, respectively.

2 First of all, unicity: if $\mathfrak{T}, \mathfrak{U}$ are balanced slices of $\mathfrak{R}$, then the respective maximal elements for $\ll_{\mathfrak{T}}$ and $\ll_{\mathfrak{U}}$ are those of $\llbracket \mathfrak{T} \rrbracket$ and $\llbracket \mathfrak{U} \rrbracket$... hence the same, since $\llbracket \mathfrak{T} \rrbracket = \llbracket \mathfrak{U} \rrbracket = \mathfrak{S}$. Below such a maximal element κ, the sets $\{\lambda; \lambda \ll_{\mathfrak{T}} \kappa\}$ and $\{\lambda; \lambda \ll_{\mathfrak{U}} \kappa\}$ are linearily ordered, hence they have a common initial segment; it is easily checked that this segment cannot be strict in either orders, hence $\{\lambda; \lambda \ll_{\mathfrak{T}} \kappa\} = \{\lambda; \lambda \ll_{\mathfrak{U}} \kappa\}$. From this we conclude that $\mathfrak{T} = \mathfrak{U}$.

Returning to the normalisation of desseins, we have just observed that cases 1, 2, 3 correspond to positive commutation, negative commutation and conversion. Indeed, a given chronicle $\mathfrak{c} \in \llbracket \mathfrak{R} \rrbracket$ ending with a positive action comes from a well-defined sequence of actions (which we shall call a dispute, see below), positive, negative and neutral. These actions altogether obviously form a balanced slice, which is the pull-back of $\mathfrak{c}$. Obviously, the pull-back of $\mathfrak{S}$ is the union of the pull-backs of the $\mathfrak{c}$, when $\mathfrak{c}$ varies through $\mathfrak{R}$. □

Corollary 1.1. A positive net converges iff it contains at least one balanced slice.

[†] In the case of a paritary slice, the number of hidden actions removed between two consecutive visible actions is always even, which is yet another example of type checking.

Corollary 1.2. A closed net converges iff it contains a balanced slice; this slice $\mathfrak{S}$ is unique, and is linearly ordered by $\ll_{\mathfrak{S}}$.

Proof. In the absence of daimon, the maximal elements of $\ll_{\mathfrak{S}}$ are visible; the closed case corresponds to only one visible action (which we replaced for convenience with some $(\xi, \emptyset)$ in the proof of Proposition 1). A forest with one maximal element is a linear order. □

3.2.4. *Disputes* More generally, the pull-back of a slice is a linear maul.

Definition 15. (Disputes) A *dispute* of $\mathfrak{R}$ is a balanced slice $\mathfrak{S} \subset \mathfrak{R}$ such that $\ll_{\mathfrak{S}}$ is a linear order. Equivalently a dispute is the pull-back of a chronicle of $[\![\mathfrak{R}]\!]$.

Disputes correspond to all possible ways travelled through $\mathfrak{R}$ during normalisation. Cases 1 and 2 are basically 'going' upwards: in a design seen as trivial net, only these cases occur and disputes are just chronicles. But the most important case is 3, which enables one to jump like Tarzan from branch to branch... I advise you to try some examples, for instance on a closed net $\{\mathfrak{Fax}, \mathfrak{D}, \mathfrak{E}\}$; we know that (19)

$$[\![\mathfrak{Fax}, \mathfrak{D}, \mathfrak{E})]\!] = [\![\mathfrak{D}, \rho^{-1}(\mathfrak{E})]\!] = [\![\rho(\mathfrak{D}), \mathfrak{E}]\!]$$

which basically means that, up to delocation, the fax establishes an interaction between $\mathfrak{D}$ and $\mathfrak{E}$... and look at the structure of the fax, p. 310: there is a systematic *under-focusing*, that is, the focus of a positive action is not chosen among the foci just created, but among foci created several (here two) steps earlier. Without under-focusing, one would always stay in the same branch. See also Remark 6, p. 329.

Slightly anticipating Definition 17, p. 327, of orthogonality, we can suggest a bridge towards denotational semantics.

Definition 16. Assume that $\mathfrak{D} \perp \mathfrak{E}$. Then the pull-back of $\langle \maltese \rangle$ along $\{\mathfrak{D}, \mathfrak{E}\}$ (which is denoted $[\mathfrak{D} \rightleftharpoons \mathfrak{E}]$) is the *dispute* generated by $\mathfrak{D}, \mathfrak{E}$. A dispute corresponds to the sequence $\langle \kappa_0, \ldots, \kappa_{n-1}, \maltese \rangle$ consisting of the $n-1$ conversions performed, followed by the final daimon.

One of the basic intuitions about ludics is to identify a design $\mathfrak{D} \in \mathbf{G}$ with the set $Dsp_{\mathbf{G}}(\mathfrak{D}) = \{[\mathfrak{D} \rightleftharpoons \mathfrak{E}]; \mathfrak{E} \in \mathbf{G}^{\perp}\}$; designs appear as sets of disputes, and since disputes can be seen as the points of a sort of coherent space, this forms a bridge between ludics and denotational semantics, see Exercise 5, p. 335. However, this theory will not be developed in this monograph.

Remark 5. We can extend Definition 16 to the case where $\mathfrak{D}, \mathfrak{E}$ are not orthogonal: the *partial* dispute $[\mathfrak{D} \rightleftharpoons \mathfrak{E}]$ is the unique sequence of conversions performed during the (attempted) normalisation of the closed net $\{\mathfrak{D}, \mathfrak{E}\}$. This sequence is finite or infinite.

4. Behaviours

The idea of a behaviour conveys the familiar intuitions of a type, of a logical formula, *etc.* in the usual syntax, and of a Scott domain, a coherent space, *etc.* in denotational semantics. Since designs convey the idea of term, proof, function, clique, *etc*., behaviours will be sets of designs, $\mathfrak{D} \in \mathbf{G}$ being something like 't is a term of type σ' or 'π is a proof of A', or even '$\mathscr{M}$ is a model of A'.

4.1. *The main analytical theorems*

These theorems are the very essence of ludics: they are *analytical*, since they are about designs, the laymen of behaviours[†].

Separation: The ludic analogue of the famous theorem (Böhm 1968; Barendregt 1984) – by the way, designs look like *Böhm trees*...

Associativity: The ludic analogue of the Church–Rosser property of λ-calculus (Church and Rosser 1936; Barendregt 1984).

Monotonicity: Takes into account the input of old-style denotational semantics – typically, Scott domains (Scott 1976; Amadio and Curien 1998).

Stability: Takes into account the input of denotational semantics of the second generation – typically, coherent spaces (Girard 1995b; Amadio and Curien 1998).

Note that all these properties hold for a single object, the *design*, instead of being split between syntax and semantics. The main notion is *orthogonality*, which enables one to formulate these principles symmetrically: with respect to λ-calculus, ludics introduces the idea of a symmetry between a term and its environment. The symmetrical nature of ludics is summarised by the *closure principle*, which is nothing but a combination of separation and associativity.

4.1.1. *Orthogonality* The first fundamental notion is *orthogonality*: two designs of opposite bases are orthogonal when they form a converging net.

In what follows we fix a *base* $\Xi \vdash \Lambda$.

Definition 17. (Orthogonality) Let $\mathfrak{D}$ be a design of base $\Xi \vdash \Lambda$, and let $\mathfrak{E}_\sigma$ be designs of respective bases $\vdash \sigma$ (if $\sigma \in \Xi$) and $\sigma \vdash$ (if $\sigma \in \Lambda$). We use the notation $\ll \mathfrak{D} \mid (\mathfrak{E}_\sigma) \gg$ for the normal form $[\![\mathfrak{D}, \ldots, \mathfrak{E}_\sigma, \ldots]\!]$ ($\mathfrak{Dai}$ or $\mathfrak{Fid}$ in case of divergence) of the net $\{\mathfrak{D}, \ldots, \mathfrak{E}_\sigma, \ldots\}$ (a sort of 'bilinear form'). $\mathfrak{D}$ and the family $(\mathfrak{E}_\sigma)$ are *orthogonal* when the normal form is total, that is, when $\ll \mathfrak{D} \mid (\mathfrak{E}_\sigma) \gg = \mathfrak{Dai}$, notation $\mathfrak{D} \perp (\mathfrak{E}_\sigma)$.

The base $\Xi \vdash \Lambda$ will be known as the *pro-base*, whereas the bases $\sigma \vdash$ or $\vdash \sigma$ will be called *anti-bases* or *counter-bases*. A design based on the pro-base will be a *pro-design*, a design based on one of the anti-bases will be an *anti-design* or *counter-design*.

By far the most important case is that of an atomic base: then there is exactly one $\mathfrak{E}_\sigma$ and we use the simplified notations $\ll \mathfrak{D} \mid \mathfrak{E} \gg$ and $\mathfrak{D} \perp \mathfrak{E}$. In that case there is only one anti-base and the two bases play symmetric roles. We shall – especially in proofs, where the general case is hardly more than notational noise – try to restrict to the atomic case.

[†] Designs have in turn their own laymen, *disputes*!

Nobody forbids us to extend the relation of orthogonality to partial designs: the pseudo-design is orthogonal to nobody!

4.1.2. *Separation*

Definition 18. (Precedence) The set of designs of base $\Xi \vdash \Lambda$ is equipped with the topology generated by the sets $(\mathfrak{E}_\sigma)^\perp$. The preorder (*precedence*) relation is defined by $\mathfrak{D}^\perp \subset \mathfrak{D}'^\perp$.

Since the closure of a point $\mathfrak{D}$ is the biorthogonal $\mathfrak{D}^{\perp\perp}$, the preorder can be defined by $\mathfrak{D} \preceq \mathfrak{D}' \leftrightarrow \mathfrak{D}' \in \mathfrak{D}^{\perp\perp}$. The topology will be $\mathscr{T}_0$ (the weakest form of separation) exactly when $\preceq$ is a partial order, which we now prove, by providing an explicit characterisation of the order. We first need to introduce some specific designs.

Definition 19. If $\mathfrak{c}$ is a proper chronicle over an atomic pro-base, we define the anti-design $\mathfrak{Opp}_\mathfrak{c}$ as follows:

- If $\mathfrak{c}$ ends with a negative action, then $\mathfrak{Opp}_c$ consists exactly of the (opposites of the) actions performed in $\mathfrak{c}$.
- If $\mathfrak{c}$ ends with a positive action, then $\mathfrak{Opp}_c$ consists of the (opposites of the) actions performed in $\mathfrak{c}$, together with an appropriate daimon.

(Indeed the definition would make sense for an arbitrary base: $\mathfrak{Opp}_\mathfrak{c}$ becomes a family of designs; the definition just given yields the *union* of the family.)

Does this make sense? A design (anti-design) is a set of chronicles, not a set of actions... and there may be several designs with exactly the same actions, for example, one made from $\langle \kappa_0, \kappa_1, \kappa_2, \kappa_3, \maltese \rangle$ and its sub-chronicles, one made from $\langle \kappa_2, \kappa_1, \kappa_0, \kappa_3, \maltese \rangle$. But, since these actions come from a single chronicle $\mathfrak{c}$, we can observe:

- That the negative rules of $\mathfrak{Opp}_\mathfrak{c}$ are at most unary: for in $\mathfrak{c}$ each focus is used exactly once: $\mathfrak{Opp}_\mathfrak{c}$ is a finite slice.
- That there is no ambiguity as to the order in which positive actions are performed: each time a chronicle of $\mathfrak{Opp}_\mathfrak{c}$ must choose a positive action, there is only one choice available. To sum up, the anti-design is unique...
- Provided it exists (!), there is no problem with coherence, which deals with different positive anti-actions with the same focus. But we could meet problems with positivity. Indeed, if the last action of $\mathfrak{c}$ is positive, then we do not get a design: we must add a daimon to ensure positivity.

Theorem 2. (Separation) $\preceq$ is a partial order, that is, the topology is $\mathscr{T}_0$. In fact $\mathfrak{D} \preceq \mathfrak{D}'$ iff $\mathfrak{D}$ is *more defined* than $\mathfrak{D}'$, that is, if every chronicle $\mathfrak{c} \in \mathfrak{D} - \mathfrak{D}'$ can be written $\mathfrak{c}' * \mathfrak{d}$ for a certain $\mathfrak{c}'$ such that $\mathfrak{c}' * \maltese \in \mathfrak{D}'$.

Proof. We assume that the base is atomic, and we use the temporary notation $\mathfrak{D} \leqslant \mathfrak{D}'$ for $\mathfrak{D}$ is *more defined* than $\mathfrak{D}'$.

- Assume that $\mathfrak{D} \leqslant \mathfrak{D}'$; then define a function ϕ from $\mathfrak{D}$ to $\mathfrak{D}'$ by $\phi(\mathfrak{c}) = \mathfrak{c}$ when $\mathfrak{c} \in \mathfrak{D}'$, $\phi(\mathfrak{c}) = \mathfrak{c}' * \maltese$ where $\mathfrak{c}'$ is defined by the condition of the theorem when $\mathfrak{c} \in \mathfrak{D} - \mathfrak{D}'$, and let $\mathfrak{D}'' = \phi(\mathfrak{D}')$. Then $\mathfrak{D}''$ is a design and $\mathfrak{D} \leqslant \mathfrak{D}'' \leqslant \mathfrak{D}'$. We shall prove that $\mathfrak{D} \preceq \mathfrak{D}'' \preceq \mathfrak{D}'$[†].

- $\mathfrak{D}'' \subset \mathfrak{D}'$, which means that, in terms of dessins, $\mathfrak{D}''$ is obtained by restricting the negative rules of $\mathfrak{D}'$, rules $(\xi, \mathscr{N})$ being replaced with rules $(\xi, \mathscr{N}')$ with $\mathscr{N} \subset \mathscr{N}'$. If $\mathfrak{D}'' \perp \mathfrak{E}$, then the fact of adding more premises in the negative rules cannot ruin convergence, and we conclude that $\mathfrak{D}' \perp \mathfrak{E}$.
- In terms of designs, $\mathfrak{D}''$ is obtained by replacing certain subdesigns of $\mathfrak{D}$ by daimons: this amounts to replacing positive rules with daimons. If $\mathfrak{D} \perp \mathfrak{E}$, and the normalisation proceeds so as to have such a modified rule as main rule, then the replacement of the rule by a daimon ensures a quicker convergence, so $\mathfrak{D}'' \perp \mathfrak{E}$.

— Assume that $\mathfrak{D} \not\preceq \mathfrak{D}'$. This means that some chronicle $\mathfrak{c} \in \mathfrak{D} - \mathfrak{D}'$ is such that for all decompositions $\mathfrak{c} = \mathfrak{c}' * \mathfrak{d}$, $\mathfrak{c}' * \maltese \notin \mathfrak{D}'$. Taking $\mathfrak{c}$ of minimal length, we eventually get $\mathfrak{c} = \mathfrak{c}' * (\xi, I)$ with $\mathfrak{c}' \in \mathfrak{D}' \cup \{\langle\rangle\}$, and either (ξ, I) is negative, or $\mathfrak{c}' * \maltese \notin \mathfrak{D}'$. There are two cases:

1 If (ξ, I) is a negative action, then there is a unique positive action κ such that $\mathfrak{c} * \kappa \in \mathfrak{D}$. There are two subcases:

(a) $\kappa = \maltese$; let $\mathfrak{E} = \mathfrak{Opp}_{\mathfrak{c}}$.

(b) κ proper; let $\mathfrak{E} = \mathfrak{Opp}_{\mathfrak{c}*\kappa}$.

2 If (ξ, I) is positive, then $\mathfrak{c}' * \kappa' \in \mathfrak{D}'$ for an appropriate *proper* $\kappa' \neq (\xi, I)$; let $\mathfrak{E} = \mathfrak{Opp}_{\mathfrak{c}}$.

The normalisations of $\ll \mathfrak{D} \mid \mathfrak{E} \gg$ and $\ll \mathfrak{D}' \mid \mathfrak{E} \gg$ involve the same initial sequence of conversions, corresponding to the actions of $\mathfrak{c}'$, up to the moment we arrive at a cut-net with main action (ξ, I) (or κ'). Then $\ll \mathfrak{D} \mid \mathfrak{E} \gg$ converges whereas $\ll \mathfrak{D}' \mid \mathfrak{E} \gg$ diverges:

1a After a conversion between the action $(-1, \xi, I)$ and the anti-action $(+1, \xi, I)$, the daimon of $\mathfrak{D}$ makes $\ll \mathfrak{D} \mid \mathfrak{E} \gg$ converge. $\ll \mathfrak{D}' \mid \mathfrak{E} \gg$ diverges because there is no anti-action in $\mathfrak{E}$ that matches the actions $(-1, \upsilon, J)$ of $\mathfrak{D}'$ that follow $\mathfrak{c}'$.

1b The same argument shows that $\ll \mathfrak{D}' \mid \mathfrak{E} \gg$ diverges in this case too. However, $\ll \mathfrak{D} \mid \mathfrak{E} \gg$ converges because of the daimon following $\widetilde{\kappa}$ in $\mathfrak{E}$.

2 After a conversion between the action $(+1, \xi, I)$ and the anti-action $(-1, \xi, I)$, the daimon of $\mathfrak{E}$ makes $\ll \mathfrak{D} \mid \mathfrak{E} \gg$ converge. $\ll \mathfrak{D}' \mid \mathfrak{E} \gg$ diverges because there is no anti-action in $\mathfrak{E}$ that matches the action κ' of $\mathfrak{D}'$.

— It remains to check that the relation $\mathfrak{D} \leqslant \mathfrak{D}'$ is antisymmetric. This is more or less obvious (see also the discussion below). □

Remark 6. It is perhaps interesting to pay a visit to Tarzan. $\mathfrak{c}$ is linear (a single chronicle), maybe with many under-focusings, whereas $\mathfrak{Opp}_{\mathfrak{c}}$ is branching, but with no under-focusing. $\mathfrak{c}$ can be seen as a non-linear travel plan for exploring $\mathfrak{Opp}_{\mathfrak{c}}$, whereas $\mathfrak{Opp}_{\mathfrak{c}}$ can

[†] Curien views this as an instance of a general decomposition of the extensional ordering ($\mathfrak{D} \leqslant \mathfrak{D}'$) via the stable ordering ($\mathfrak{D}'' \subset \mathfrak{D}'$), see Curien *et al.* (2000) and Remark 8, page 333.

only explore in a linear way, typically $\mathfrak{c}$ or a permutation of $\mathfrak{c}$. $\mathfrak{Opp}_{\mathfrak{c}}$ is linear exactly when there is no under-focusing in $\mathfrak{c}$.

$\mathfrak{D} \preceq \mathfrak{D}'$ means that the design $\mathfrak{D}$ is more defined than the design $\mathfrak{D}'$[†] when $\mathfrak{D}'$ is obtained from $\mathfrak{D}$ by means of an enlargement of the negative rules and a replacement of some positive rules by daimons. But why do we say that $\mathfrak{D}$ is more defined? This is clear concerning the daimons, a daimon is more opportunistic (that is, less informative) than a proper action. As to negative rules, we have to think that $\mathfrak{D}$ takes more chances, he knows more what he wants or does not want. Typically, the negative daimon who has a full branching does not carry any information: if you say 'yes' to everybody, who are you?

However, the idea of wider negative rules and shorter branches does not quite work with dessins. In the first half of the previous proof, we were able to produce common dessins for $\mathfrak{D}, \mathfrak{D}''$ and $\mathfrak{D}'', \mathfrak{D}'$, but this is generally not possible between $\mathfrak{D}, \mathfrak{D}'$. So this is a convenient intuition, and nothing more. One can summarise this principle by the inequality:

$$\Omega \preceq (\xi, I) \preceq \maltese \tag{20}$$

which means that we can replace Ω (absent premise of negative rule) with a real premise, and that a proper positive action can be replaced with a daimon. By the way, what are the maximal, minimal elements of $\preceq$? The daimons are clearly maximal with respect to $\preceq$: simply because they are absolutely opportunistic, that is, they are orthogonal to anybody.

Example 13. When the base is negative, there is a smallest element, namely the empty design $\mathfrak{Sk} = \emptyset$, the *Skunk*:

$$\dfrac{}{\xi \vdash \Lambda}\,(\xi,\emptyset) \tag{21}$$

The minimal designs of base $\vdash \Lambda$ are of the form $\mathfrak{Sk}_{(\lambda,I)}$, the *positive Skunk*:

$$\mathfrak{Sk}_{(\lambda,I)} = \dfrac{\dots \quad \dfrac{}{\lambda * i \vdash}\,(\lambda{*}i,\emptyset) \quad \dots}{\vdash \Lambda}\,(\lambda,I) \tag{22}$$

with $\lambda \in \Lambda$[‡].

Example 14. The maximal *proper* designs of base $\vdash \Lambda$ are of the form $\mathfrak{Ram}_{(\lambda,I)}$, *Ramification*:

$$\mathfrak{Ram}_{(\lambda,I)} = \dfrac{\dots \quad \dfrac{\dots \quad \dfrac{}{\vdash \lambda * i * J}\,\maltese \quad \dots}{\lambda * i \vdash}\,(\lambda{*}i,\wp_f(\mathbb{N})) \quad \dots}{\vdash \Lambda}\,(\lambda,I) \tag{23}$$

[†] Sorry for the mismatch, but there is a conflict between 'more defined' and 'more easily normalised'. After some hesitation, I decided to order $\preceq$ this way on the grounds that it corresponds to tradition, and that inclusion (see Subsection 4.1.3, p. 331) is a particular case of $\preceq$.

[‡] If we enlarge the discussion so as to allow partial designs, $\mathfrak{Fid}$ is always the minimum.

4.1.3. *Stability*

Theorem 3. (Stability) Normalisation commutes with compatible intersections: if K is non-empty and $\mathfrak{R}_k \subset \mathfrak{R}$ for all k, then

$$[\![\bigcap_{k\in K} \mathfrak{R}_k]\!] = \bigcap_{k\in K} [\![\mathfrak{R}_k]\!] \tag{24}$$

Proof. The inclusion $[\![\bigcap_k \mathfrak{R}_k]\!] \subset \bigcap_k [\![\mathfrak{R}_k]\!]$ is immediate. Conversely, assume that $\mathfrak{c} \in \bigcap_k [\![\mathfrak{R}_k]\!]$. Then $\mathfrak{c}$ has a pull-back (see Theorem 1, p. 325) $\mathfrak{r}_k \subset \mathfrak{R}_k$; but $\mathfrak{r}_k \subset \mathfrak{R}$, and by unicity of the pull-back $\mathfrak{r}$ of $\mathfrak{c}$ with respect to $\mathfrak{R}$, $\mathfrak{r}_k = \mathfrak{r}_{k'} = \mathfrak{r}$ for any $k, k' \in K$. Then $\mathfrak{r} \subset \bigcap_k \mathfrak{R}_k$ and $\mathfrak{c} \in [\![\bigcap_k \mathfrak{R}_k]\!]$. □

Remark 7.

1 The result works without a hypothesis of totality, which makes it modular. However, observe that the theorem holds replacing the hypothesis that the $\mathfrak{R}_k$ are included in some net $\mathfrak{R}$ with the assumption that $\bigcap_k \mathfrak{R}_k$ is a total net (that is, is made of total designs): each $\mathfrak{r}_k$ is a *dispute*, that is, is linearly ordered by $\ll_{\mathfrak{r}_k}$, and we can look at the earliest point of disagreement between any two of these disputes, and observe that the condition of positivity (or totality) breaks down. This peculiarity can be explained by the fact that $\bigcap_k \mathfrak{R}_k$ total implies that $\bigcup_k \mathfrak{R}_k$, although maybe not a net (see exercise below) is a sort of… lax net. This remark suggests a theory of *lax designs*, which is beyond the scope of this monograph.

2 A typical example of stability is given by

$$\ll \mathfrak{D} \mid \bigcap_k \mathfrak{E}_k \gg = \bigcap_k \ll \mathfrak{D} \mid \mathfrak{E}_k \gg \tag{25}$$

which will be used to define incarnation.

3 Observe that ‘double’ stability:

$$\ll \mathfrak{D}_1 \cap \mathfrak{D}_2 \mid \mathfrak{E}_1 \cap \mathfrak{E}_2 \gg = \ll \mathfrak{D}_1 \mid \mathfrak{E}_1 \gg \cap \ll \mathfrak{D}_2 \mid \mathfrak{E}_2 \gg \tag{26}$$

cannot be reduced to the previous case.

4 On coherent spaces, the stable ordering corresponds to inclusion: the *stability property* of Berry (Berry 1978) formulated in coherent spaces (Girard 1995b) states that

$$f(a \cap b) = f(a) \cap f(b) \text{ provided } a \cup b \text{ is a clique.}$$

The stability of designs yields

$$\ll \mathfrak{D} \mid \mathfrak{E} \cap \mathfrak{E}' \gg = \ll \mathfrak{D} \mid \mathfrak{E} \gg \cap \ll \mathfrak{D} \mid \mathfrak{E}' \gg \tag{27}$$

(without any assumption as to $\mathfrak{E} \cup \mathfrak{E}'$, except for the implicit hypothesis that $\mathfrak{E} \cap \mathfrak{E}'$ is a design).

Exercise 1. Find designs $\mathfrak{E}, \mathfrak{E}'$ such that $\mathfrak{E} \cap \mathfrak{E}'$ is a design, but $\mathfrak{E} \cup \mathfrak{E}'$ is not a design. *(Hint: play on the splitting of contexts.)*

4.1.4. *Associativity* Strictly speaking, since normalisation is deterministic, there is no need for a Church–Rosser property. But besides the technical meaning of Church–Rosser, there

https://doi.org/10.1017/S096012950100336X

is a deeper one, namely that in the presence of two cuts, the output of normalisation is the same, whether we normalise them together, or one after the other, something like $ABC = (AB)C$:

Theorem 4. (Associativity) Normalisation is associative: let $\{\mathfrak{R}_0, \ldots, \mathfrak{R}_n\}$ be a net of nets, then

$$\llbracket \mathfrak{R}_0 \cup \ldots \cup \mathfrak{R}_n \rrbracket = \llbracket \llbracket \mathfrak{R}_0 \rrbracket, \ldots, \llbracket \mathfrak{R}_n \rrbracket \rrbracket \tag{28}$$

Furthermore (and this is by far the most important point), the result holds without hypotheses of totality.

Proof. This is an immediate application of pull-backs: if $\mathfrak{c} \in \llbracket \llbracket \mathfrak{R}_0 \rrbracket, \ldots, \llbracket \mathfrak{R}_n \rrbracket \rrbracket$, then consider its pull-back $\{\llbracket \mathfrak{c}_0 \rrbracket, \ldots, \llbracket \mathfrak{c}_n \rrbracket\}$, then pull-back again the $\mathfrak{c}_i$ along the $\mathfrak{R}_i$, which yields $\mathfrak{d}_0 \subset \mathfrak{R}_0, \ldots, \mathfrak{d}_n \subset \mathfrak{R}_n$, and conclude that $\llbracket \mathfrak{d}_0 \cup \ldots \cup \mathfrak{d}_n \rrbracket = \mathfrak{c}$, that is, that $\mathfrak{c} \in \llbracket \mathfrak{R}_0 \cup \ldots \cup \mathfrak{R}_n \rrbracket$. The other direction is established in a similar way. □

In other words, given $\mathfrak{S} = \{\mathfrak{R}_0, \ldots, \mathfrak{R}_n\}$, we get the same result, whether we normalise each $\mathfrak{R}_p$ and then normalise the net $\mathfrak{S}' = \{\mathfrak{D}_0, \ldots, \mathfrak{D}_n\}$ made with their normal forms, or directly normalise the net $\mathfrak{S}'' = \mathfrak{R}_0 \cup \ldots \cup \mathfrak{R}_n$.

Technically speaking, the theorem is hardly more than the Church–Rosser property for multiplicative-additive logic formulated in terms of proof-nets (Girard 1996), with some minor modifications.

4.1.5. *The closure principle* The principle basically states that we can restrict attention to closed nets. Typically, if $\mathfrak{D}, \mathfrak{E}$ are designs of bases $\xi \vdash \lambda$ and $\vdash \xi$, respectively, the normal form $\llbracket \mathfrak{D}, \mathfrak{E} \rrbracket$ is the unique design $\mathfrak{D}'$ of base $\vdash \lambda$ such that for every $\mathfrak{F}$ of base $\lambda \vdash$:

$$\llbracket \mathfrak{D}', \mathfrak{F} \rrbracket = \llbracket \mathfrak{D}, \mathfrak{E}, \mathfrak{F} \rrbracket \tag{29}$$

The normal form of a net $\mathfrak{S}$ is determined by the normal forms of all completions of $\mathfrak{S}$ into a closed net. The principle is very useful, since closed nets do not need commutative conversions.

Theorem 5. (Closure principle) Let $\mathfrak{R}$ be a net of base $\Xi \vdash \Lambda$. Then the normal form $\llbracket \mathfrak{R} \rrbracket$ of $\mathfrak{R}$ is the unique design $\mathfrak{D}$ such that for any family of anti-designs $(\mathfrak{E}_\sigma)$, we have $\mathfrak{D} \perp (\mathfrak{E}_\sigma)$ iff the closed net $\llbracket \mathfrak{R} \cup \ldots \cup \mathfrak{E}_\sigma \cup \ldots \rrbracket = \mathfrak{Dai}$, that is, converges.

Proof. $\mathfrak{R}$ satisfies the condition from associativity, and unicity is guaranteed by separation. □

The closure principle stands behind the associativity of multiplicatives, see Chapter 6, for it basically asserts the possibility of defining adjoints by equations of the style (68), see p.350; such an equation is reminiscent of the familiar Hilbert space definition:

$$\langle u(x) \mid y \rangle = \langle x \mid u^*(y) \rangle \tag{30}$$

Observe that – such a thing may sometimes happen – ludics is better behaved than Hilbert spaces on this sole question, for if u is an operator, we cannot define u by means

https://doi.org/10.1017/S096012950100336X

of something like $\langle u(x) \mid y\rangle = \langle u \mid x^{*} \otimes y\rangle$, whereas in ludics we can indifferently write (when $\mathfrak{U}, \mathfrak{X}, \mathfrak{Y}$ are of bases $\vdash \xi, \xi' \quad \xi \vdash$ and $\xi' \vdash$, respectively):

$$\ll [\![\mathfrak{U}, \mathfrak{X}]\!] \mid \mathfrak{Y} \gg \; = \; \ll [\![\mathfrak{U}, \mathfrak{Y}]\!] \mid \mathfrak{X} \gg \; = [\![\mathfrak{U}, \mathfrak{X}, \mathfrak{Y}]\!] \tag{31}$$

Exercise 2. Let $\mathfrak{F}$ be a design of base $\xi \vdash \xi'$ such that, for all $\mathfrak{E}$ of base $\vdash \xi$:

$$[\![\mathfrak{F}, \mathfrak{E}]\!] \supset \rho(\mathfrak{E}) \tag{32}$$

where ρ is the *delocation* of Example 11, p. 320. Prove that $\mathfrak{F} = \mathfrak{Fax}_{\xi,\xi'}$[†].

4.1.6. *Monotonicity*

Theorem 6. (Monotonicity) Normalisation is increasing with respect to the order $\preceq$: if $\mathfrak{D}_0 \preceq \mathfrak{E}_0 \ldots \mathfrak{D}_n \preceq \mathfrak{E}_n$, then $[\![\mathfrak{D}_0, \ldots \mathfrak{D}_n]\!] \preceq [\![\mathfrak{E}_0, \ldots \mathfrak{E}_n]\!]$.

Proof. The proof is a typical application of the closure principle. For instance, assume that $\mathfrak{F}$ and $\mathfrak{D}$ are of bases $\vdash \Upsilon, \xi$ and $\xi \vdash \Sigma$, respectively, and that $\mathfrak{D} \preceq \mathfrak{D}'$. Then $[\![\mathfrak{D}, (\mathfrak{E}_\sigma)]\!] \preceq [\![\mathfrak{D}', (\mathfrak{E}_\sigma)]\!]$ for all anti-designs $\{\mathfrak{E}_\sigma ; \sigma \in \Sigma, \xi\}$. Cut $\mathfrak{D}, \mathfrak{D}'$ with $\mathfrak{F}$ and consider anti-designs $\{\mathfrak{C}_\tau ; \tau \in \Sigma \cup \Upsilon\}$ for $[\![\mathfrak{F}, \mathfrak{D}]\!]$. Define $\mathfrak{E}_\sigma = \mathfrak{C}_\sigma$ for $\sigma \in \Sigma$ and $\mathfrak{E}_\sigma = [\![\mathfrak{F}, (\mathfrak{C}_\upsilon)]\!]$ for $\sigma = \xi$. We get[‡]

$$[\![\mathfrak{D}, (\mathfrak{E}_\sigma)]\!] \preceq [\![\mathfrak{D}', (\mathfrak{E}_\sigma)]\!] \tag{33}$$

and using associativity, we also have $[\![[\![\mathfrak{F}, \mathfrak{D}]\!], (\mathfrak{C}_\tau)]\!] \preceq [\![[\![\mathfrak{F}, \mathfrak{D}']\!], (\mathfrak{C}_\tau)]\!]$. From which we conclude that $[\![\mathfrak{F}, \mathfrak{D}]\!] \preceq [\![\mathfrak{F}, \mathfrak{D}']\!]$. □

As for the other analytical theorems, the theorem persists for partial nets.

Remark 8. In denotational semantics, two orders coexist, the stable order, and the more traditional *extensional*[§] order, which is defined in the style

$$f \preceq g \quad \text{iff} \quad f(a) \preceq g(a) \quad \text{for all} \;\; a. \tag{34}$$

In ludics the role of the extensional order is played by $\preceq$, whereas the stable order corresponds to plain inclusion.

4.2. *Behaviours and incarnation*

4.2.1. *Behaviours*

Definition 20. A *behaviour* is a set $\mathbf{G}$ of designs of a given base equal to its biorthogonal. A behaviour is *positive* or *negative* according to the polarity of its base.

[†] For cheaters, the proof can be found on page 388.

[‡] $\mathfrak{E}_\xi$ may be partial, but then (33) holds for trivial reasons.

[§] The expression refers to obsolete nonsense, for example, the Jurassic opposition between intensional and extensional. 'Extrospective' – which means that the order only refers to the *result* of normalisation – is more accurate, and free from ideological commitments.

https://doi.org/10.1017/S096012950100336X

Here ‘Set’ is to be taken in its straightforward meaning, with no lurking constructivist, predicativist, *etc*. restriction.

Example 15.

1 The set of all designs of a given base is a behaviour, the *Skunk*, equal to $\emptyset^{\perp}$. We use the notation $\top^{\epsilon}$ where ϵ is the polarity of the base or simply $\top$ when the base is negative.
2 The set $\{\mathfrak{Dai}^{\epsilon}\}$, is a behaviour, the *Daimon*, indeed the smallest one, equal to $\emptyset^{\perp\perp}$. We use the notation $\mathbf{0}^{\epsilon}$ or simply $\mathbf{0}$ when the base is positive.
3 More generally, if $\mathbf{E}$ is any set of anti-designs, then $\mathbf{E}^{\perp}$ is a behaviour, and any behaviour is of this form (take $\mathbf{E} = \mathbf{G}^{\perp}$).

Example 16. There is a smallest behaviour (think of the ‘principal type’) containing a given design $\mathfrak{D}$, namely $\mathfrak{D}^{\perp\perp}$. This *principal behaviour* is given by the formula

$$\mathfrak{D}^{\perp\perp} = \{\mathfrak{D}' ; \mathfrak{D} \preceq \mathfrak{D}'\} \tag{35}$$

Since the daimons are orthogonal to all designs, every behaviour contains the daimon of the right polarity. The nightmare of *empty types* is definitely fixed. Behaviours enjoy certain immediate closure properties.

Theorem 7. (Closure) If $\mathfrak{D} \preceq \mathfrak{E}$ and $\mathfrak{D} \in \mathbf{G}$, then $\mathfrak{E} \in \mathbf{G}$. If K is non-empty and $\mathfrak{D}_k \in \mathbf{G}$ for all $k \in K$ and $\bigcup_k \mathfrak{D}_k$ is a design, then $\bigcap_k \mathfrak{D}_k \in \mathbf{G}$.

Proof. The proof is immediate from the definition of $\preceq$ (Definition 18 p. 328), and the Stability Theorem (Theorem 3, p. 331). □

4.2.2. *Incarnation* If $\mathfrak{D} \in \mathbf{G}$ and $\mathfrak{D} \subset \mathfrak{E}$, then $\mathfrak{E} \in \mathbf{G}$, but for ‘bad’ reasons: none of the new chronicles in $\mathfrak{E}$ is useful for guaranteeing the membership of $\mathbf{G}$. In a manner of speaking, $\mathfrak{D}, \mathfrak{E}$ are ‘equivalent’ in $\mathbf{G}$, and $\mathbf{G}$ is naturally equipped with an equivalence relation, which is the symmetric and transitive closure of inclusion. But, fortunately, one can distinguish one design $|\mathfrak{D}|_{\mathbf{G}}$ in each class, so that $\mathfrak{D} \simeq \mathfrak{E} \Leftrightarrow |\mathfrak{D}|_{\mathbf{G}} = |\mathfrak{E}|_{\mathbf{G}}$.

Theorem 8. (Incarnation) Given $\mathfrak{E} \in \mathbf{G}$, there is a smallest design $\mathfrak{D} \subset \mathfrak{E}$ such that $\mathfrak{D} \in \mathbf{G}$.

Proof. The set of designs of $\mathbf{G}$ included in $\mathfrak{E}$ is a non-empty family whose union is a design. By the Closure Theorem (Theorem 7), the intersection $\mathfrak{D}$ of this family does belong to $\mathbf{G}$. □

Definition 21. (Incarnation) The design $\mathfrak{D}$ of Theorem 8, is called the *incarnation* of $\mathfrak{E}$ and written $|\mathfrak{E}|$, or $|\mathfrak{E}|_{\mathbf{G}}$ if we want to be precise.

$$|\mathfrak{E}|_{\mathbf{G}} = \bigcap \{\mathfrak{E}' ; \mathfrak{E}' \subset \mathfrak{E} \text{ and } \mathfrak{E}' \in \mathbf{G}\} \tag{36}$$

A design $\mathfrak{D} \in \mathbf{G}$ is *incarnated* or *material* when $\mathfrak{D} = |\mathfrak{D}|$. We define the incarnation $|\mathbf{G}|$ of $\mathbf{G}$ to be the set of its material designs.

The incarnation of $\mathfrak{E}$ is the part of $\mathfrak{E}$ that can be interactively recognised via cuts with anti-designs taken from $\mathbf{G}^{\perp}$. The incarnation is contravariant, that is,

$$\mathbf{G} \subset \mathbf{H} \Rightarrow |\mathfrak{E}|_{\mathbf{H}} \subset |\mathfrak{E}|_{\mathbf{G}} \tag{37}$$

Hence the incarnation of $\mathfrak{E}$ is maximum when $\mathbf{G}$ is the principal behaviour $\mathfrak{E}^{\perp\perp}$ containing $\mathfrak{E}$; in this case $|\mathfrak{E}| = \mathfrak{E}$ (which is an easy consequence of the Separation Theorem). The incarnation is minimum when $\mathbf{G}$ is the greatest behaviour $\top^{\epsilon}$. For instance, when the base is negative

$$|\mathfrak{E}|_{\top} = \mathfrak{Sk} \tag{38}$$

That is, the incarnation of $\mathfrak{E}$ is the empty set.

Exercise 3. Show that

$$\mathfrak{D} \preceq \mathfrak{E} \Rightarrow |\mathfrak{D}| \preceq |\mathfrak{E}| \tag{39}$$

Hint: use the 'more defined than' characterisation of $\mathfrak{D} \preceq \mathfrak{E}$, *see Theorem 2, p. 328.*

Exercise 4. If $\mathfrak{S}$ is a finite slice, show that the principal behaviour $\mathfrak{S}^{\perp\perp}$ has only finitely many material designs.

Exercise 5. If $\mathfrak{D}, \mathfrak{D}' \in \mathbf{G}$, show that $|\mathfrak{D}|_{\mathbf{G}} = |\mathfrak{D}'|_{\mathbf{G}}$ iff for all $\mathfrak{E} \in \mathbf{G}^{\perp}$ the disputes $[\mathfrak{D} \rightleftharpoons \mathfrak{E}]$ and $[\mathfrak{D}' \rightleftharpoons \mathfrak{E}]$ are equal.

4.2.3. *Behaviours as games* Our basic objection to 'game semantics' is that we do not want any lurking referee. But in no way do we deny games as a useful *intuition*. Indeed, ludics is about games, but games by *consensus*, not games with an external rule. In what follows we give the game-theoretic translation of a closed net $[\![\mathfrak{D}, \mathfrak{E}]\!]$.

Players: They are nicknamed Even and Odd . Once we take the viewpoint of one of these players, he is called Proponent (positive), the other player being styled Opponent (negative) – 'Me' *vs.* 'You'.

Plays: A play is the sequence of conversions involved in the converging normalisation of $\ll \mathfrak{D} \mid \mathfrak{E} \gg$, which we have called a *dispute*, see Subsection 3.2.4, p. 326. Once we take the viewpoint of one player, for example, the side of $\mathfrak{D}$, then the positive actions are moves of Proponent, and their parity is the parity of $\mathfrak{D}$, that is, of Proponent.

Strategies: Clearly $\mathfrak{D}, \mathfrak{E}$ play the roles of strategies for Proponent and Opponent.

Winner: Who wins and who loses is to be defined in Section 9.1. For the moment, let us mention only one of the winning conditions: not to play the daimon, that is, not to *give up*.

This description is not satisfactory, since usually in a game the idea is, more or less, to win: assume that $\mathfrak{D}$ has a positive base $\vdash \xi$, then the design

$$\mathfrak{One} = \frac{}{\vdash \xi}\,(\xi, \emptyset) \tag{40}$$

is perfectly winning, since Opponent can only react by a *daimon*, that is, by giving up: the only chronicle extending $\langle(-1, \xi, \emptyset)\rangle$ is $\langle(-1, \xi, \emptyset), \maltese\rangle$! In other words, Proponent can use the atomic weapon, not quite an exciting interactive idea! But this is because we are dealing with *unrestricted* designs.

Rule: In a behaviour, `Proponent` and `Opponent` *behave* according to two sets $\mathbf{G}, \mathbf{G}^{\perp}$ of designs. The orthogonality condition can be seen as a constraint, that is, that designs in $\mathbf{G}$ must be orthogonal to $\mathbf{G}^{\perp}$. So we have $\mathbf{G}^{\perp}$ is 'the rule of the game $\mathbf{G}$', and conversely $\mathbf{G}$ is 'the rule of the game $\mathbf{G}^{\perp}$'. Only a few games can be described in this *interactive* way: we are therefore limiting our 'language', to use an expression from another century, but what can be expressed in this restricted fashion has exceptional properties.

The remainder of this monograph will demonstrate the power of the method, so let us just give an example, that is, how we can interactively help `Proponent` from using the atomic weapon $(\xi, \emptyset)$: it is enough to put in $\mathbf{G}^{\perp}$ the negative design $\mathfrak{E} = \mathfrak{Dir}_{\wp_f(\mathbb{N})-\{\emptyset\}}$

$$\mathfrak{E} = \cfrac{\cdots \quad \cfrac{}{\vdash \xi * I}\maltese \quad \cdots}{\xi \vdash}\,(\xi,\wp_f(\mathbb{N})-\{\emptyset\}) \tag{41}$$

since $\ll \mathfrak{One} \mid \mathfrak{E} \gg = \mathfrak{Fid}$, that is, $\mathfrak{One} \not\perp \mathfrak{E}$: in a manner of speaking, the design $\mathfrak{E}$ 'recuses' the action $(\xi, \emptyset)$ – there is a *dissensus*.

The design (41) is not quite a strategy, in the sense that `Opponent` has not the slightest chance to win[†]. If we consider that the output of a non-converging net is a draw, `Opponent` is playing for a *draw*.

Inside a behaviour $\mathbf{G}$ a design $\mathfrak{D}$ plays the role of a strategy, but two designs with the same incarnation induce the same plays, that is, the same strategy. This is why we can think of the incarnation $|\mathfrak{D}|_{\mathbf{G}}$ as the *strategy* induced by $\mathfrak{D}$ in the 'game' $\mathbf{G}$.

4.2.4. *Behaviours as syntax or semantics* We can more or less imagine that a design is like a proof, but we could as well see a design as a sort of model. Then the duality between $\mathbf{G}$ and $\mathbf{G}^{\perp}$ can be seen as syntax/semantics, if we decide to put the syntax on the side of $\mathbf{G}$, the semantics on the side of $\mathbf{G}^{\perp}$. The orthogonality relates a would-be proof of $\mathbf{G}$ with a would-be refutation of the same $\mathbf{G}$. We have abolished, in principle, this syntax/semantics schizophrenia, but there is still a real difference, namely that syntax is usually given by a set of rules, which in turn gives rise to a semantics, and then the question of the completeness of the syntax is posed.

Definition 22. (Completeness) An *ethics* is a set $\mathbf{E}$ of designs of a given base $\Xi \vdash \Upsilon$; if $\mathbf{E}^{\perp\perp} = \mathbf{G}$, one says that $\mathbf{E}$ *is an ethics for* $\mathbf{G}$. An ethics is *complete* when it contains the incarnation of its biorthogonal, that is, when $|\mathbf{E}^{\perp\perp}| \subset \mathbf{E}$.

Usually a behaviour $\mathbf{G}$ will be *presented* by an ethics, that is, $\mathbf{G} = \mathbf{E}^{\perp\perp}$. The typical example of an ethics is the set $\mathbf{E}$ of the designs that correspond to the syntactical proofs of a given formula A. The counter-models of A can be replaced with $\mathbf{E}^{\perp}$, so that $\mathbf{E}^{\perp\perp}$ stands for what is validated by all counter-models of A, so completeness for A is basically the fact that the biorthogonal is not needed. Sometimes (especially in the negative case) $\mathbf{E}^{\perp\perp} = \mathbf{E}$ becomes wrong for stupid reasons: this is why completeness is only *modulo*

[†] Unless `Proponent` is stupid enough to start with a daimon.

incarnation. A typical example is given by the maximum behaviour $\top$, which admits the finite ethics $\{\mathfrak{Sk}\}$!

4.3. *Connectives*

The next three chapters will analyse behaviours, so as to understand how behaviours are created from simpler ones, that is, how they *socialise*. A connective may be covariant or contravariant in each parameter. Using negation, we can restrict attention to covariant connectives: a covariant connective is increasing with respect to inclusion, that is, compatible with 'subtyping'.

We conclude this chapter with some elementary examples of connectives.

4.3.1. *Delocation*

Definition 23. (Delocation) A *delocation* from *locus* ξ to *locus* ξ' is a partial injective map θ from the subloci of ξ to the subloci of ξ' such that:

— $\theta(\xi) = \xi'$.
— For all σ there is a function θ_σ from biases to biases such that $\theta(\sigma * i) = \theta(\sigma) * \theta_\sigma(i)$.

A delocation is *positive* when ξ, ξ' have the same parity, and *negative* otherwise.

If $\mathfrak{c} = \langle \ldots, (\sigma_p, I_p), \ldots \rangle$ is a proper chronicle of base $\vdash \xi$ (respectively, $\xi \vdash$), one defines $\theta(\mathfrak{c})$ of base $\vdash \xi'$ (respectively, $\xi' \vdash$) as $\theta(\mathfrak{c}) = \langle \ldots, (\theta(\sigma_p), \theta_{\sigma_p}(I_p)), \ldots \rangle$. If $\mathfrak{c}$ is an improper chronicle $\mathfrak{d} * \maltese$, one defines $\theta(\mathfrak{c}) = \theta(\mathfrak{d}) * \maltese$.

If $\mathfrak{D}$ is a design of base $\vdash \xi$ (respectively, $\xi \vdash$) one defines $\theta(\mathfrak{D}) = \{\theta(\mathfrak{c}); \mathfrak{c} \in \mathfrak{D}\}$, a design of base $\vdash \xi'$ (respectively, $\xi' \vdash$).

Example 17. We define two delocations φ, ψ from $\langle\rangle$ to itself:

$$\varphi(i * \sigma) = 3i * \sigma \tag{42}$$

$$\psi(i * \sigma) = (3i + 1) * \sigma \tag{43}$$

These delocations are in the style of the *Hilbert hotel*: 'Not enough rooms, you are kidding! Just rename room $\sharp n$ as $\sharp 3n$... (respectively, $\sharp 3n + 1$)': two hotels can be accommodated inside a single one, moreover, there is still room for a third hotel (the rooms $\sharp 3n + 2$). So when we need a *spiritual* binary connective, we shall use the delocating facility, typically, $\mathbf{G} \oplus \mathbf{H} = \varphi(\mathbf{G}) \uplus \psi(\mathbf{H})$ – see Section 5.2.2, p. 342. These delocations are *almost disjoint*: this means that the ranges $\Im(\varphi)$ and $\Im(\psi)$ intersect on the sole $\langle\rangle$. This peculiarity will be used to define *spiritual connectives*.

Definition 24. (Delocation: behaviours) If $\mathbf{G}$ is a behaviour of base $\vdash \xi$ (respectively, $\xi \vdash$) and θ is a *total* delocation from ξ to ξ', define the behaviour $\theta(\mathbf{G})$ of base $\vdash \xi'$ (respectively, $\xi' \vdash$) by:

$$\theta(\mathbf{G}) = \{\theta(\mathfrak{D}); \mathfrak{D} \in \mathbf{G}\}^{\perp\perp} \tag{44}$$

Proposition 2. The material designs of $\theta(\mathbf{G})$ are exactly the images under θ of the material designs of $\mathbf{G}$.

Typically, the image under θ of the behaviour $\top$ is not $\top$, unless θ is surjective. But it contains the only material design of $\theta(\top) = \top$, that is, the Skunk.

Corollary 2.1. The image under θ of $\mathbf{G}$ is a complete ethics for $\theta(\mathbf{G})$.

Delocation is not exactly a connective: this convenient operation, which is essential for defining spiritual operations such as the usual connectives of logic, hardly changes anything – it is, rather, a way of creating 'occurrences'. Delocation is essential to second-order quantification, since we must speak of different occurrences of the same propositional variable X: as a variable, X stands for a positive behaviour $\mathbf{X}$ of base $\vdash \langle\rangle$, as an occurence, X stands for a delocation $\theta(\mathbf{X})$.

Example 18. The general form of the identity axiom is a 'delocation axiom', implemented by a sort of fax that generalises our basic fax of Example 2, p. 309. We assume that θ, ρ are delocations from $\langle\rangle$ to the disjoint ξ, ξ'. We want to define $\mathfrak{Fax}_{\xi,\xi'}$[†], a design of base $\xi \vdash \xi'$, with the property that, for all $\mathfrak{D}, \mathfrak{E}$ of bases $\vdash \langle\rangle$ and $\langle\rangle \vdash$, respectivly:

$$[\![\theta(\mathfrak{D}), \mathfrak{Fax}]\!] = \rho(\mathfrak{D}) \tag{45}$$

$$[\![\rho(\mathfrak{E}), \mathfrak{Fax}]\!] = \theta(\mathfrak{E}) \tag{46}$$

The solution is given by:

$$\dfrac{\dots \qquad \dfrac{\begin{array}{c}\vdots\ \mathfrak{Fax}_{\xi'*\rho(i),\xi*\theta(i)}\\ \dots\ \xi' * \rho(i) \vdash \xi * \theta(i) \dots\end{array}}{\vdash \xi', \xi * \theta(I)}\,{\scriptstyle (\xi',\rho(I))} \qquad \dots}{\xi \vdash \xi'}\,{\scriptstyle (\xi,\theta(\wp_f(\mathbb{N})))} \tag{47}$$

The fax $\mathfrak{Fax}_{\xi'*\rho(i),\xi*\theta(i)}$ corresponds to the delocations $\rho_i(\sigma) = \rho(i * \sigma)$, $\theta_i(\sigma) = \theta(i * \sigma)$.

The example just given will be the typical inhabitant of the *sequent of behaviours* $\theta(\mathbf{G}) \vdash \rho(\mathbf{G})$: see Section 6.3.

4.3.2. *The shift* This seemingly minor operation is one of the true technical novelties of ludics. It basically changes the polarity, by adding a 'dummy action'.

Definition 25. (Shift) Let $\mathfrak{c}$ be a chronicle of base $\vdash \Lambda, \xi * i$ (respectively, $\xi * i \vdash \Lambda$); the *shift* $\updownarrow\mathfrak{c}$ of $\mathfrak{c}$ is the chronicle $(\xi, \{i\}) * \mathfrak{c}$ of base $\xi \vdash \Lambda$ (respectively, $\vdash \xi, \Lambda$).

If $\mathfrak{D}$ is a design of base $\vdash \Lambda, \xi * i$ (respectively, $\xi * i \vdash \Lambda$), the *shift* of $\mathfrak{D}$ is the design $\updownarrow\mathfrak{D} = \{\updownarrow\mathfrak{c}; \mathfrak{c} \in \mathfrak{D}\} \cup \{\langle(\xi, \{i\})\rangle\}$ of base $\xi \vdash \Lambda$ (respectively, $\vdash \xi, \Lambda$).

If $\mathbf{G}$ is a behaviour of base $\vdash \xi * i$ (respectively, $\xi * i \vdash$), the *shift* of $\mathbf{G}$ is the behaviour $\updownarrow\mathbf{G} = \{\updownarrow\mathfrak{D}; \mathfrak{D} \in \mathbf{G}\}^{\perp\perp}$ of base $\xi \vdash$ (respectively, $\vdash \xi$).

A *prime* behaviour is any behaviour of the form $\updownarrow\mathbf{G}$.

[†] We should indicate θ, ρ, but the notation would be too heavy.

It is convenient to replace $\updownarrow$with $\downarrow$(respectively,$\uparrow$) when the polarity of the shift is positive (respectively, negative)[†].

Proposition 3. If $\mathbf{G}$ is negative, then $\downarrow\mathbf{G} = \{\downarrow\mathfrak{D}; \mathfrak{D} \in \mathbf{G}\} \cup \{\maltese\}$. If $\mathbf{G}$ is positive, then $\{\uparrow\mathfrak{D}; \mathfrak{D} \in \mathbf{G}\}$ is a complete ethics for $\uparrow\mathbf{G}$. Moreover, $(\updownarrow\mathbf{G})^{\perp} =\updownarrow(\mathbf{G}^{\perp})$.

This (obvious) proposition establishes the completeness of the shift, by providing complete ethics. In the case of a positive shift, the daimon has to be added, that is, there is the possibility of giving up; in particular, the shift is not involutive, even up to isomorphism. In the negative case, the set of shifts is only a complete ethics, since we can enlarge our designs with chronicles not beginning with $(\xi, \{i\})$, without changing the incarnation.

The shift enables one to define connectives when there is a mismatch of polarities: typically, we can define $\mathbf{G} \otimes \mathbf{H}$ by $\mathbf{G}\otimes \downarrow\mathbf{H}, \downarrow\mathbf{G} \otimes \mathbf{H}, \downarrow\mathbf{G}\otimes \downarrow\mathbf{H}$, when at least one of $\mathbf{G}, \mathbf{H}$ is negative.

5. Additives

In the (informal) hierarchy of connectives, additives stand below multiplicatives, simply because implication (by far the most important connective – think of syllogisms) is basically multiplicative. Indeed these connectives are more difficult to grasp than multiplicatives, for several reasons:

- λ-calculus is basically multiplicative/exponential. Of course a useful pairing function can be defined in λ-calculus, but it is not the real thing.
- The tradition of natural deduction is unsatisfactory too, since intuitionistic disjunction (which is not too far from the additive disjunction) is badly behaved. The ‘commutative rules’ of disjunction clearly indicate a mismatch[‡].
- With linear logic, additives became primitive connectives, but they were still mistreated. For instance, multiplicative proof-nets avoid in a perfect way commutative conversions for the positive $\otimes$, but additive proof-nets are not that simple. Geometry of interaction works simply for multiplicatives and exponentials (Girard 1989a), but its extension to additives (Girard 1995a) is not that successful.

In ludics additives are really given their full space; indeed ludics basically comes from a reflection on additives, especially neutrals, and this is why additives appear as the connectives with the best properties, if not the most important ones. They are also easy to understand, which is why we start with them.

General locative additives $⩀, \uplus$ are introduced, $⩀$ being the general form of an ‘intersection type’. These operations have no exceptional properties, but for the fact of being *strictly* commutative and associative. The connectives $\&, \oplus$ correspond to the particular

[†] You can use the following to help remember the notation: $\downarrow$ changes a negative behaviour into a positive one, like the exponential ‘!’.

[‡] This is due to the fact that, unlike the negative connectives $\Rightarrow, \wedge, \forall$, disjunction and existence are positive. Natural deduction, centred on implication, is negative, that is, ‘proceeds downwards’, whereas the positive operations like disjunction proceed upwards.

case of *disjoint* behaviours. The additive decomposition writes any behaviour as a $\oplus$ (positive case) or $\&$ (negative case) of *connected* behaviours, the decomposition being unique. Complete ethics are given for $\oplus$ and $\&$, under the form of the two remarkable theorems of this chapter:

Disjunction property: $\mathbf{G} \oplus \mathbf{H}$ is the union $\mathbf{G} \cup \mathbf{H}$.

Mystery of incarnation: $\&$ behaves like the Cartesian product *on incarnations.*

In this chapter, the base is atomic: either $\vdash \langle\rangle$ or $\langle\rangle \vdash$.

5.1. *Locative additives*

5.1.1. *Directories* The designs $\mathfrak{Ram}_{(\langle\rangle,I)}$ (positive) have already been introduced (Example 14, p. 330); we now introduce the designs $\mathfrak{Dir}_{\mathcal{N}}$ (negative) *Directory*:

$$\mathfrak{Dir}_{\mathcal{N}} = \frac{\cdots \quad \dfrac{}{\vdash I}\maltese \quad \cdots}{\langle\rangle \vdash}\,(\langle\rangle,\mathcal{N}) \tag{48}$$

Definition 26. A *directory* is a set $\mathcal{N}$ of ramifications. If $\mathbf{G}$ is a positive behaviour, the *directory of* $\mathbf{G}$ is the set $\P\mathbf{G} = \{I\,;\mathfrak{Ram}_{(\langle\rangle,I)} \in \mathbf{G}\}$. If $\mathbf{G}$ is a negative behaviour, the incarnation $|\mathfrak{Dai}^-|_{\mathbf{G}}$ of the negative daimon is of the form $\mathfrak{Dir}_{\mathcal{N}}$ for some directory $\mathcal{N}$, which is by definition the *directory of* $\mathbf{G}$: $|\mathfrak{Dai}^-|_{\mathbf{G}} = \mathfrak{Dir}_{\P\mathbf{G}}$.

A *reservoir* is any set of biases; the *reservoir* of a behaviour is defined as $\S\mathbf{G} = \bigcup \P\mathbf{G}$.

Proposition 4. If $\mathbf{G}$ is positive, then $\P\mathbf{G}$ consists of those I such that $(\langle\rangle,I)$ is the first action of a design $\mathfrak{D} \in \mathbf{G}$. Moreover, $\P\mathbf{G}^{\perp} = \P\mathbf{G}$.

Proof. If $(\langle\rangle,I)$ is the first action of $\mathfrak{D} \in \mathbf{G}$, then $\mathfrak{D} \preceq \mathfrak{Ram}_{(\langle\rangle,I)}$ and $\mathfrak{Ram}_{(\langle\rangle,I)} \in \mathbf{G}$. The incarnation of $\mathfrak{Dai}^-$ in $\mathbf{G}^{\perp}$ is clearly of the form $\mathfrak{Dir}_{\mathcal{N}}$, and it is immediate that $I \in \mathcal{N}$ exactly when $(\langle\rangle,I)$ is the first action of some $\mathfrak{D} \in \mathbf{G}$. □

5.1.2. *The connective 'Inter'*

Definition 27. (Inter) Let $\mathbf{G}_k$ be a family of behaviours of the same base. Then we define $\bigcap_k \mathbf{G}_k$ as the intersection of the $\mathbf{G}_k$.

The definition makes sense because an intersection of orthogonals is the orthogonal of a union. We will use the more specific notations $\mathrel{\cap\mkern-11mu-}, \mathrel{\cap\mkern-11mu+}$ to indicate the polarity of the $\mathbf{G}_k$.

Proposition 5. The connective $\bigcap$ is *strictly* commutative and associative. Its neutral element is the empty intersection, namely the Skunk $\top^{\epsilon}$, and its absorbing element is the smallest negative behaviour, the Daimon $\mathbf{0}^{\epsilon}$.

The most important point is 'strictly' – we are speaking of equalities, not of vague isomorphisms.

5.1.3. *The connective 'Union'* Assume that the base is positive.

Definition 28. (Union) Let $\mathbf{G}_k$ be a family of behaviours. Then we define $\overset{*}{\cup}_k \mathbf{G}_k$ as $(\bigcup_k \mathbf{G}_k)^{\perp\perp}$.

We will use the more specific notations $\overset{-}{\cup}, \overset{+}{\cup}$ to indicate the polarity of the $\mathbf{G}_k$.

Proposition 6. The connective $\overset{*}{\cup}$ is strictly commutative and associative. Its neutral element is the Daimon $\mathbf{0}^{\epsilon}$ and its absorbing element is the Skunk $\top^{\epsilon}$.

Completeness fails for $\overset{*}{\cup}$, since there is no general way to remove the biorthogonal, that is, to find a complete ethics; the discussion in Subsection 7.1.2, p. 359, establishes that one may think of $\overset{*}{\cup}$ as the only incomplete connective.

5.1.4. *Intersection and incarnation*

Theorem 9.

$$|\mathfrak{D}|_{\bigcap_k \mathbf{G}_k} = \bigcup_k |\mathfrak{D}|_{\mathbf{G}_k} \tag{49}$$

Proof. Let $\mathfrak{D}' = |\mathfrak{D}|_{\bigcap_k \mathbf{G}_k}$ $\mathfrak{D}'' = \bigcup_k |\mathfrak{D}|_{\mathbf{G}_k}$ The inclusion $\mathfrak{D}'' \subset \mathfrak{D}'$ is immediate, by contravariance of incarnation. Conversely, observe that $\mathfrak{D}''$ is orthogonal to all designs in the ethics $\bigcup_k \mathbf{G}_k{}^{\perp}$, and so belongs to $\bigcap_k \mathbf{G}_k$. But since $\mathfrak{D}'$ is material, this forces the equality. □

Nothing like this holds for the connective 'union'.

5.1.5. *Intersection and directory* The directory is covariant in the positive case, and contravariant in the negative case; moreover, we have the following proposition.

Proposition 7.

$$\P \underset{k}{\overset{+}{\bigcap}} \mathbf{G}_k = \bigcap_k \P\mathbf{G}_k \tag{50}$$

$$\P \underset{k}{\overset{+}{\bigcup}} \mathbf{G}_k = \bigcup_k \P\mathbf{G}_k \tag{51}$$

Proof. The statement is obvious in the case of $\overset{+}{\cap}$, and almost obvious in the case of $\overset{+}{\cup}$. □

Corollary 7.1.

$$\P \underset{k}{\overset{-}{\bigcap}} \mathbf{G}_k = \bigcup_k \P\mathbf{G}_k \tag{52}$$

$$\P \underset{k}{\overset{-}{\bigcup}} \mathbf{G}_k = \bigcap_k \P\mathbf{G}_k \tag{53}$$

5.2. *Additives*

5.2.1. *Plus and With*

Definition 29. Two behaviours $\mathbf{G}, \mathbf{H}$ of the same polarity are *disjoint* when their directories are disjoint. A behaviour $\mathbf{G}$ is *connected* when its directory $\P\mathbf{G}$ is a singleton $\{I\}$, in which case I is called the *ramification* of the behaviour.

We use the notations $\bigoplus, \&$ instead of $\biguplus, \Cap$ to signify that the operation has been applied to pairwise disjoint behaviours.

Proposition 8. The positive behaviours $\mathbf{G}, \mathbf{H}$ are disjoint iff $\mathbf{G} \cap \mathbf{H} = \mathbf{0}$ $(= \{\mathfrak{Dai}\})$. The negative behaviours $\mathbf{G}, \mathbf{H}$ are disjoint iff for all $\mathfrak{D} \in \mathbf{G}, \mathfrak{E} \in \mathbf{H}$ $|\mathfrak{D}|_{\mathbf{G}} \cap |\mathfrak{E}|_{\mathbf{H}} = \emptyset$.

Proof. The proof is more or less immediate. For instance, in the case of negative behaviours, observe that $|\mathfrak{D}|_{\mathbf{G}} \preceq \mathfrak{Dir}_{\P\mathbf{G}}$[†]. □

5.2.2. *The spiritual dilemma* The connective $\oplus$ is partial, but spiritual logic has addicted us to total operations. We are left with a dilemma:

— Either we keep things as they are, so that the connective remains partial, but with exceptionally good properties, in particular, real equalities, not isomorphisms.

— Or we absolutely want a total operation, corresponding to familiar disjunction. Then we fix two delocations φ, ψ from $\langle\rangle$ to itself, see Example 17, p. 337, and we *redefine* $\mathbf{G} \oplus \mathbf{H}$ as $\varphi(\mathbf{G}) \uplus \psi(\mathbf{H})$. The delocated connective just introduced is total[‡], but no longer handled by equalities, only (canonical) isomorphisms. The equalities proved below become canonical isomorphisms: typically, the disjunction property (Theorem 11, p. 343) becomes $\mathbf{G} \oplus \mathbf{H} = \varphi(\mathbf{G}) \cup \psi(\mathbf{H})$.

Unless otherwise stated, we make the choice of the strict, but partial operations: an equality is nicer than an isomorphism; the same choice will be made for the other connectives, for example, $\&, \otimes, ⅋$. As far as we deal with spiritual issues, that is, questions whose solution does not depend on the location, for example, completeness issues, this methodological bias is a huge simplification. However, the most novel locative features, typically the prenex forms of Chapter 7, crucially depend on the fact that connectives receive their standard meaning: in that chapter they will be treated as total operations.

It must be observed that delocation is not always possible, there is a problem with the tensor unit (see Subsection 6.2.4, p. 355), which uses an empty ramification that cannot be delocated. If we are concerned with spiritual logic, it is therefore advisable to restrict consideration to behaviours $\mathbf{G}$ such that $\emptyset \notin \P\mathbf{G}$.

5.3. *Completeness properties*

The purpose of this section is to provide us with complete ethics for the connectives $\&$ and $\oplus$.

[†] See also Exercise 3.

[‡] As long as $\emptyset \notin \P\mathbf{G}, \P\mathbf{H}$ – see below.

5.3.1. *The mystery of incarnation* We will formulate the result in the binary case, but it holds without restriction.

Theorem 10. (Mystery of incarnation)

$$|\mathbf{G} \mathbin{\&} \mathbf{H}| = |\mathbf{G}| \times |\mathbf{H}| \tag{54}$$

Proof. Assume that $\mathfrak{D} \in \mathbf{G} \mathbin{\&} \mathbf{H}$ is material. Then the two incarnations $\mathfrak{E} = |\mathfrak{D}|_{\mathbf{G}}$ and $\mathfrak{F} = |\mathfrak{D}|_{\mathbf{H}}$ are included in $\mathfrak{D}$. We conclude that $\mathfrak{E} \cup \mathfrak{F} \subset \mathfrak{D}$.

Conversely, if $\mathfrak{E}, \mathfrak{F}$ are incarnated in $\mathbf{G}, \mathbf{H}$, respectively, then Proposition 8, p. 342, shows that they are disjoint, so their union is a design $\mathfrak{D}$. Should $\mathfrak{D}$ not be incarnated in $\mathbf{G} \mathbin{\&} \mathbf{H}$, we would get $\mathfrak{E}' \cup \mathfrak{F}' \subsetneq \mathfrak{D}$ for appropriate $\mathfrak{E}', \mathfrak{F}'$ and one of $\mathfrak{E}, \mathfrak{F}$ would not be incarnated.

To sum up, the material designs in $\mathbf{G} \mathbin{\&} \mathbf{H}$ are exactly the unions of a material design of $\mathbf{G}$ and a material design of $\mathbf{H}$, and such a decomposition is unique. □

Remark 9. There is in fact something fishy about this result, since the left-hand side involves a strictly commutative operation, whereas the Cartesian product is notoriously non-commutative *etc*. In fact, we should have written an isomorphism instead. Our symbol of equality means that among all possible isomorphic definitions of the product, we choose the best one, namely

$$X \mathbin{\bigcup\mkern-15mu\times} Y = \{x \cup y ; x \in X, y \in Y\} \tag{55}$$

This *locative product* is strictly commutative, associative and with $\{\emptyset\}$ as neutral. But when any $x \in X$ is disjoint from any $y \in Y$, we use the notation $X \times Y$. This is therefore a partial operation, which is to the usual Cartesian product what our & is to the usual one.

The mystery of incarnation also makes sense for the delocated version of & (see Subsection 5.2.2, p. 342), and can be written

$$|\mathbf{G} \mathbin{\&} \mathbf{H}| \simeq |\mathbf{G}| \times |\mathbf{H}| \tag{56}$$

But we only get an isomorphism, and we must be fluent in category theory to actually *express* that this isomorphism is 'natural'. There is a better reason, beyond categorical nonsense: an underlying equality!

The locative product is to the usual set-theoretic product what union is to disjunction. If you look at the extant literature, you will surely find many examples of this basic operation, which seems to have been overlooked.

5.3.2. *The disjunction property* The dual version of the mystery of incarnation.

Theorem 11. (Disjunction property) If the index set is non-empty, then

$$\bigoplus_k \mathbf{G}_k = \bigcup_k \mathbf{G}_k \tag{57}$$

Proof. The case of a binary $\oplus$ is enough. If $\mathfrak{D} \in (\mathbf{G} \cup \mathbf{H})^{\perp\perp} - \mathbf{G} \cup \mathbf{H}$, then $\mathfrak{D}$ is not orthogonal to some $\mathfrak{E} \in \mathbf{G}^{\perp}$ and some $\mathfrak{F} \in \mathbf{H}^{\perp}$, and we can assume both of $\mathfrak{E}, \mathfrak{F}$ material.

But we know by Proposition 8, p. 342, that $\mathfrak{E} \cap \mathfrak{F} = \emptyset$: we have $\mathfrak{E} \cup \mathfrak{F}$ is a design in the intersection $\mathbf{G}^{\perp} \cap \mathbf{H}^{\perp}$, not orthogonal to $\mathfrak{D}$, which is a contradiction. □

In other words, we have found a complete ethics for the connective 'Plus'. In particular, $\mathbf{G} \oplus \mathbf{H} = \mathbf{G} \cup \mathbf{H}$, with quite the old familiar meaning 'A cut-free proof of $A \oplus B$ is proof of A or a proof of B', proofs being replaced with designs, and formulas with behaviours. Observe that the disjunction is not exclusive, since $\mathbf{G} \cap \mathbf{H} = \{\mathfrak{Dai}\}$, but since the daimon is *losing* (see Section 9.1, p. 373), it will be exclusive as long as *winning* designs are considered.

5.3.3. *Additive decomposition*

Theorem 12. (Additive decomposition) Any positive behaviour can be written in a unique way as the $\bigoplus$ of connected behaviours

$$\mathbf{G} = \bigoplus_{I \in \P\mathbf{G}} \mathbf{G}_I \tag{58}$$

Proof. $\mathbf{G}_I$ consists of those designs $\mathfrak{D} \in \mathbf{G}$ precisely starting with $(\langle\rangle, I)$, together with the daimon. Since $\mathbf{G} = \bigcup \mathbf{G}_I$ is a behaviour, $\mathbf{G}_I{}^{\perp\perp} \subset \mathbf{G}$ and $\P\mathbf{G}_I = \{I\}$ forces $\mathbf{G}_I$ to be a behaviour. □

Corollary 12.1. Any negative behaviour can be written in a unique way as the $\&$ of connected behaviours

$$\mathbf{G} = \mathop{\&}_{I \in \P\mathbf{G}} \mathbf{G}_I \tag{59}$$

Exercise 6. In the negative case, show that a complete ethics for $\mathbf{G}_I$ is given by the set $\{\mathfrak{D}_I\,; \mathfrak{D} \in \mathbf{G}\}$, where $\mathfrak{D}_I \subset \mathfrak{D}$ only retains those chronicles starting with $(\langle\rangle, I)$.

Remark 10. $\mathbf{G}_I$ has been defined for all I, but does not contribute to the decomposition when $I \notin \P\mathbf{G}$, since $\mathbf{G}_I = \mathbf{0}$ or $\mathbf{G}_I = \top$ depending on the polarity of $\mathbf{G}$.

5.4. *Subtyping*

5.4.1. *Subtyping and incarnation* Let us comment on these results, especially the mystery of incarnation, which relates the two readings of the additive conjunction, intersection and product: this has to do with *subtyping* and *inheritance*. $\mathbf{G} \,\&\, \mathbf{H} \subset \mathbf{H}$ means that every object of type $\mathbf{G} \,\&\, \mathbf{H}$ is of type $\mathbf{H}$. We are therefore dealing with some kind of record: a component for $\mathbf{G}$, a component for $\mathbf{H}$, and perhaps additional components that do not matter, typically, if our design $\mathfrak{D}$ belongs to $\mathbf{G} \,\&\, \mathbf{H} \,\&\, \mathbf{K}$. Now what is incarnation? This is the part of the design relevant to a behaviour. Hence $|\mathfrak{D}|_{\mathbf{G}\&\mathbf{H}}$ only retains the part of $\mathfrak{D}$ relevant to $\mathbf{G}, \mathbf{H}$, whereas $|\mathfrak{D}|_{\mathbf{H}}$ kills information relevant to $\mathbf{G}$. Since the information relevant to each of them is disjoint from that relevant to the others, we get the result.

This mystery is only possible because of the coexistence of two notions within ludics:

— The official notion of a behaviour, for which an object of type $\mathbf{G} \,\&\, \mathbf{H}$ is an object of type $\mathbf{G}$. In this conception, subtyping is inclusion.

— The old-style definition, for which a pair $(a, b) \in A \,\&\, B$ is not of type A: we are in fact dealing with *material* designs. Subtyping is handled through operations that destroy the useless part, that is, compute the incarnation in the supertype: *coercion* maps.

The additive decomposition corresponds to a sort of general 'record style': each ramification I denotes a field, which may be missing. For an incarnated design of $\mathbf{G} \,\&\, \mathbf{H}$, there is one component for each $I \in \P\mathbf{G} \cup \P\mathbf{H}$; the coercion between $\mathbf{G} \,\&\, \mathbf{H}$ and $\mathbf{H}$ corresponds to the removal of those components I that belong to $\P\mathbf{G}$. Such a coercion can be implemented, provided both behaviours have been delocalised at disjoint *loci* $\xi \vdash$ (for $\mathbf{G} \,\&\, \mathbf{H}$) and $\xi' \vdash$ (for $\mathbf{H}$) by a partial fax, induced by the partial delocation $\theta(\xi * I * \sigma) = \xi' * I * \sigma$, for $I \in \P\mathbf{H}$, and that is undefined on $\xi * I * \sigma$ for $I \notin \P\mathbf{H}$. The pseudo-fax of Example 3, p. 310, implements such a coercion, in the case $\P\mathbf{H} = \{\{3, 7\}, \{4, 7\}\}$.

5.4.2. *Incarnation and records* Imagine the following record:

$$\texttt{coord}\colon (3, 4) \qquad \texttt{colour}\colon \textit{green} \qquad \texttt{shape}\colon \textit{circle} \tag{60}$$

The fields `coord,colour,shape` are encoded by means of the biases $2, 3, 8$, respectively: they become negative behaviours included in $(\mathfrak{Ram}_{(\langle\rangle,\{2\})})^{\perp}$, $(\mathfrak{Ram}_{(\langle\rangle,\{3\})})^{\perp}$ and $(\mathfrak{Ram}_{(\langle\rangle,\{8\})})^{\perp}$, respectively – see below for a precise definition. The planar coordinates (m, n) are rendered by $\{2m, 2n + 1\}$, so that $(3, 4)$ becomes[†] $\{6, 9\}$, colours are encoded by numbers, for instance *green* is 8, and the shape *circle* corresponds to the bias 0. Our record can be expressed by means of the negative design in the following example.

Example 19.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash 261}\,{\scriptstyle (261,\emptyset)}}{26 \vdash}\,{\scriptstyle (26,\{\{1\}\})} \qquad \dfrac{\dfrac{}{\vdash 291}\,{\scriptstyle (291,\emptyset)}}{29 \vdash}\,{\scriptstyle (29,\{\{1\}\})}}{\vdash 2}\,{\scriptstyle (2,\{6,9\})} \qquad \dfrac{\dfrac{\dfrac{}{\vdash 381}\,{\scriptstyle (381,\emptyset)}}{38 \vdash}\,{\scriptstyle (38,\{\{1\}\})}}{\vdash 3}\,{\scriptstyle (3,\{8\})} \qquad \dfrac{\dfrac{\dfrac{}{\vdash 801}\,{\scriptstyle (801,\emptyset)}}{80 \vdash}\,{\scriptstyle (80,\{\{1\}\})}}{\vdash 8}\,{\scriptstyle (8,\{0\})}}{\langle\rangle \vdash}\,{\scriptstyle (\langle\rangle,\{\{2\},\{3\},\{8\}\})}$$

— The first (negative) branching lists the fields (questions) $2, 3, 8$.

— The second branchings (positive) yield the answers – the values of the fields.

— The upper layers are just a convenient way to end the design: one proceeds uniformly with $(\xi, \{1\}), (\xi * 1, \emptyset)$ (which corresponds to $\uparrow \mathfrak{One}$, see Equation (40), p. 335). Anticipating the notation of the next chapter, if $\mathbf{1}_{\xi}$ denotes the positive unit 'One' of base $\vdash \xi$, we can define `coord` $=\uparrow\oplus_{m,n}(\downarrow\uparrow\mathbf{1}_{2*2m*1} \otimes \downarrow\uparrow\mathbf{1}_{2*(2n+1)*1})$ `colour` $=\uparrow\oplus_{n} \downarrow\uparrow\mathbf{1}_{3*n*1}$ `shape` $=\uparrow\oplus_{n} \downarrow\uparrow\mathbf{1}_{8*n*1}$

But say that we do not care about shapes, only coordinate and colour matter: our design is also of type `coord` & `colour` anyway. However, since the shape is not used, we could as well replace it with its incarnation, as in the following example.

[†] In order to avoid the symbol *, we have chosen very small numbers, so that we can write 261 instead of $2 * 6 * 1$ *etc.*

Example 20.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash 261}\,{\scriptstyle(261,\emptyset)}}{26 \vdash}\,{\scriptstyle(26,\{\{1\}\})} \qquad \dfrac{\dfrac{}{\vdash 291}\,{\scriptstyle(291,\emptyset)}}{29 \vdash}\,{\scriptstyle(29,\{\{1\}\})}}{\vdash 2}\,{\scriptstyle(2,\{6,9\})} \qquad \dfrac{\dfrac{\dfrac{}{\vdash 381}\,{\scriptstyle(381,\emptyset)}}{38 \vdash}\,{\scriptstyle(38,\{\{1\}\})}}{\vdash 3}\,{\scriptstyle(3,\{8\})}}{\langle\rangle \vdash}\,{\scriptstyle(\langle\rangle,\{\{2\},\{3\}\})}$$

This represents the truncated record

$$\texttt{coord:}\ (3,4); \quad \texttt{colour:}\ \mathit{green} \tag{61}$$

Further, forgetting the colour yields the incarnation in the following example.

Example 21.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash 261}\,{\scriptstyle(261,\emptyset)}}{26 \vdash}\,{\scriptstyle(26,\{\{1\}\})} \qquad \dfrac{\dfrac{}{\vdash 291}\,{\scriptstyle(291,\emptyset)}}{29 \vdash}\,{\scriptstyle(29,\{\{1\}\})}}{\vdash 2}\,{\scriptstyle(2,\{6,9\})}}{\langle\rangle \vdash}\,{\scriptstyle(\langle\rangle,\{\{2\}\})}$$

That is, the record

$$\texttt{coord:}\ (3,4) \tag{62}$$

in type `coord`. On the other hand, keeping only the colour yields the incarnation in the following example.

Example 22.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash 381}\,{\scriptstyle(381,\emptyset)}}{38 \vdash}\,{\scriptstyle(38,\{\{1\}\})}}{\vdash 3}\,{\scriptstyle(3,\{8\})}}{\langle\rangle \vdash}\,{\scriptstyle(\langle\rangle,\{\{3\}\})}$$

That is, the record

$$\texttt{colour:}\ \mathit{green} \tag{63}$$

in type `colour`. Up to incarnation, a record of type `coord` & `colour` is the pair (that is, the disjoint union) of a record of type `coord` and a record of type `colour`.

The next example is a (sort of) fax that takes a record, and, regardless of the other attributes, will set all colours to *black*, encoded by 0. Of course the source must be delocated in ξ' and the target to the disjoint ξ

Example 23.

$$\dfrac{\cdots \quad \dfrac{\begin{array}{c}\vdots\ \mathfrak{Fax}_{\xi' i,\xi i}\\ \ldots\ \xi' i \vdash \xi i \ldots\end{array}}{\vdash \xi', \xi I}\,(\xi',I) \quad (I\neq\{3\})\ldots \quad \dfrac{\dfrac{\cdots \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \xi 301}\,(\xi 301,\emptyset)}{\xi 30 \vdash}\,(\xi 30,\{\{1\}\})}{\vdash \xi 3}\,(\xi 3,\{0\})}{\xi' 3c1 \vdash \xi 3}\,(\xi' 3c1,\{\emptyset\})}{\vdash \xi' 3c, \xi 3}\,(\xi' 3c,\{1\}) \quad \cdots}{\xi' 3 \vdash \xi 3}\,(\xi' 3,\{\{c\};c\in\mathbb{N}\})}{\vdash \xi', \xi 3}\,(\xi',\{3\})}{\xi \vdash \xi'}\,(\xi,\wp_f(\mathscr{N}))$$

Through normalisation, this design replaces any record located at $\xi' \vdash$ with the 'same' record delocated at $\xi \vdash$, and painted black, the colour encoded by $c \in \mathbb{N}$ becoming colour 0, that is, black. The fax-like part recopies all fields distinct from $\{3\}$, so `coord,shape` are recopied, but other fields will be as well, should they be present.

Exercise 7. Formulate a (sort of) fax that just moves the coordinates according to a given function f, and then figure out the 'composition of both faxes', that is, the fax that changes both coordinates and colour.

As for the representation of the planar coordinates, let us mention the alternative solution, which consists of creating two fields: 0 for the x-coordinate and 1 for the y-coordinate. Then the record becomes

$$\texttt{x-coord: } 3 \qquad \texttt{y-coord: } 4 \tag{64}$$

which is rendered in the following example.

Example 24.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash 031}\,(031,\emptyset)}{03 \vdash}\,(03,\{\{1\}\})}{\vdash 0}\,(0,\{3\}) \qquad \dfrac{\dfrac{\dfrac{}{\vdash 141}\,(141,\emptyset)}{14 \vdash}\,(14,\{\{1\}\})}{\vdash 1}\,(1,\{4\})}{\langle\rangle \vdash}\,(\langle\rangle,\{\{0\},\{1\}\})$$

This admits coercions into records containing only `x-coord` or `y-coord`, contrary to our original choice.

These basic remarks are just an invitation to revisit the existing approaches to subtyping in the light of ludics, for example, the *intersection types* developed in Torino (Coppo *et al.* 1981). The connection with *object-oriented* programming in the style of Abadi and Cardelli (1996) is also very exciting...

6. Multiplicatives

We will now turn our attention to multiplicatives, which are the main connectives of logic: their importance is due to the pregnancy of *linear implication* $\multimap$, which is the part of intuitionistic or classical implication that deals with implication – the other part dealing with reuse, as expressed by the founding formula

$$A \Rightarrow B = !A \multimap B \tag{65}$$

The basic notion is that of the tensor product $\mathfrak{A} \circledast \mathfrak{B}$[†] of two positive designs. Indeed, the definition is not that obvious when the ramifications I, J of the first actions of $\mathfrak{A}, \mathfrak{B}$ intersect: there are four protocols (two asymmetric and two symmetric), giving rise to the tensor products $\mathfrak{A} \olessthan \mathfrak{B}$, $\mathfrak{A} \ogreaterthan \mathfrak{B}$ ($=\mathfrak{B} \olessthan \mathfrak{A}$), $\mathfrak{A} \odot \mathfrak{B}$ and $\mathfrak{A} \obar \mathfrak{B}$. Each of these tensors has an adjoint *application*, namely $\mathfrak{F}[\mathfrak{A}], [\mathfrak{A}]\mathfrak{F}, (\mathfrak{F})\mathfrak{A}, \{\mathfrak{F}\}\mathfrak{A}$. As a consequence, the connectives defined from these tensors are associative *etc*. They are related by the inclusions

$$\odot \subset \begin{matrix} \olessthan \\ \\ \ogreaterthan \end{matrix} \subset \obar \tag{66}$$

Under the hypothesis of *mutual independence*, all these tensors receive natural complete ethics. The stronger hypothesis of *being mutually alien* makes the four tensors collapse into a single one, written $\otimes$.

As usual, the base is atomic, either $\vdash \langle\rangle$ or $\langle\rangle \vdash$.

6.1. *Locative multiplicatives*

6.1.1. *Non-commutative adjunctions*

Definition 30. (N.C. tensor product of designs) We define the *tensor product* $\mathfrak{A} \olessthan \mathfrak{B}$ of positive designs $\mathfrak{A}, \mathfrak{B}$:

— If either $\mathfrak{A}, \mathfrak{B}$ is a daimon, then $\mathfrak{A} \olessthan \mathfrak{B} = \mathfrak{Dai}$.

— Otherwise $\mathfrak{A}, \mathfrak{B}$ have first actions $(\langle\rangle, I)$ and $(\langle\rangle, J)$, respectively. Replace in each chronicle of $\mathfrak{A}, \mathfrak{B}$ the first action $(\langle\rangle, I)$ or $(\langle\rangle, J)$ with $(\langle\rangle, I \cup J)$, so as to get $\mathfrak{A}', \mathfrak{B}'$. Then $\mathfrak{A} \olessthan \mathfrak{B}$ is the subset of $\mathfrak{A}' \cup \mathfrak{B}'$ consisting of:

  - All chronicles of $\mathfrak{B}'$.
  - Those chronicles of $\mathfrak{A}'$ whose second action (i, I') is such that $i \notin J$.

In other words, when the ramifications I, J overlap, part of $\mathfrak{A}$ is replaced. Imagine an aeroplane: first ordinary passengers book (design $\mathfrak{A}$), then comes the time of the VIPs (design $\mathfrak{B}$) – if there is a conflict for one seat (bias $k \in I \cap J$), the VIP gets it[‡].

Remark 11. The tensor product of designs has been defined in the case of an atomic base; but the definition can immediately be extended to the general case of two positive bases $\vdash \xi, \Lambda$, $\vdash \xi, \Lambda'$ such that $\vdash \xi, \Lambda, \Lambda'$ is a pitchfork. The same will be true for the two commutative tensors...

[†] We use the notation $\circledast$ for one of the many tensor products that arise naturally.

[‡] Fortunately, modern companies have heard of category theory and spiritual principles: they use delocation to minimise such conflicts.

Theorem 13. (N.C. adjunctions) Let $\mathfrak{F}, \mathfrak{A}, \mathfrak{B}$ be designs, with $\mathfrak{F}$ (think of a function) negative, $\mathfrak{A}, \mathfrak{B}$ (think of arguments) positive. Then there exist *unique* negative designs $\mathfrak{F}[\mathfrak{A}]$ (not depending on $\mathfrak{B}$), $[\mathfrak{B}]\mathfrak{F}$ (not depending on $\mathfrak{A}$) such that

$$\ll \mathfrak{F} \mid \mathfrak{A} \olessthan \mathfrak{B} \gg \;=\; \ll \mathfrak{F}[\mathfrak{A}] \mid \mathfrak{B} \gg \;=\; \ll \mathfrak{A} \mid [\mathfrak{B}]\mathfrak{F} \gg \tag{67}$$

Proof. Unicity is a consequence of the Separation Theorem (Theorem 2, p. 328), so let us concentrate on existence.

First observe that $\mathfrak{F}[\mathfrak{Dai}] = [\mathfrak{Dai}]\mathfrak{F} = \mathfrak{Dai}^-$ is enough to satisfy the equations if one of $\mathfrak{A}, \mathfrak{B}$ is a daimon. So we are left with the construction of $\mathfrak{F}[\mathfrak{A}], [\mathfrak{B}]\mathfrak{F}$ in the case $\mathfrak{A}, \mathfrak{B}$ have first actions, $(\langle\rangle, I)$ and $(\langle\rangle, J)$. For $i \in I$ (respectively, $j \in J$) let $\mathfrak{A}_i$ (respectively, $\mathfrak{B}_j$) be the subdesign of $\mathfrak{A}$ (respectively, $\mathfrak{B}$) whose base is $i \vdash$ (respectively, $j \vdash$). For $I \in \wp_f(\mathbb{N})$ let $\mathfrak{F}_I$ be the subdesign of $\mathfrak{F}$ whose base is $\vdash I$ (with the convention that $\mathfrak{F}_I = \mathfrak{Fid}$ when the premise $\vdash I$ is missing). $\mathfrak{F}[\mathfrak{A}]$ and $[\mathfrak{B}]\mathfrak{F}$ will be defined by means of their subdesigns $\mathfrak{F}[\mathfrak{A}]_K$ and $[\mathfrak{B}]\mathfrak{F}_K$ of bases $\vdash K$ for each ramification K, and again some of these subdesigns may be partial, that is, missing.

$\mathfrak{F}[\mathfrak{A}]_K$**:** Form a cut-net between $\mathfrak{F}_{I \cup K}$ and the $\mathfrak{A}_i$ in $I - K$; its base is $\vdash K$ and its normal form, partial or total, is $\mathfrak{F}[\mathfrak{A}]_K$. The equation $\ll \mathfrak{F} \mid \mathfrak{A} \olessthan \mathfrak{B} \gg = \ll \mathfrak{F}[\mathfrak{A}] \mid \mathfrak{B} \gg$ follows from the construction of $\mathfrak{F}[\mathfrak{A}]_J$ and the definition of normalisation.

$[\mathfrak{B}]\mathfrak{F}_K$**:** Form a cut-net between $\mathfrak{F}_{J \cup K}$ and the $\mathfrak{B}_j$ in J; its base is $\vdash K - J$, and its normal form, partial of total, defines $[\mathfrak{B}]\mathfrak{F}_K$, up to the detail that the base must be changed to $\vdash K$, which is just a matter of replacing in every chronicle the initial action $(\langle\rangle, K - J)$ with $(\langle\rangle, K)$. The equation $\ll \mathfrak{F} \mid \mathfrak{A} \olessthan \mathfrak{B} \gg = \ll \mathfrak{A} \mid [\mathfrak{B}]\mathfrak{F} \gg$ follows from the construction of $[\mathfrak{B}]\mathfrak{F}_I$ and the definition of normalisation. □

Remark 12. As in λ-calculus, we have a type-free notion of application of a function to an argument. However, there are important differences:

- The argument can be applied to the *left* or to the *right*, yielding *different* results.
- The function must be *negative* and the argument *positive*, yielding a *negative* output.
- Application is a *total* operation: $\mathfrak{F}[\mathfrak{A}]$ is always defined (since negative).

Exercise 8. Show that the tensor product[†] cannot be extended to the partial design $\mathfrak{Fid}$ and still admit adjoints.

Proposition 9. The operation $\olessthan$ is associative and its neutral element is $\mathfrak{One}$, see (40) p. 335.

Corollary 9.1. $\mathfrak{F} \perp \mathfrak{A}$ iff the chronicle $\langle(\langle\rangle, \emptyset)\rangle$ belongs to $\mathfrak{F}[\mathfrak{A}]$ (equivalently to $[\mathfrak{A}]\mathfrak{F}$).

Proof. $\ll \mathfrak{F} \mid \mathfrak{A} \olessthan \mathfrak{One} \gg = \ll \mathfrak{F}[\mathfrak{A}] \mid \mathfrak{One} \gg$, *etc.* □

Corollary 9.2. $\mathfrak{F}[\mathfrak{A}][\mathfrak{B}] = \mathfrak{F}[\mathfrak{A} \olessthan \mathfrak{B}]$, $[\mathfrak{A}][\mathfrak{B}]\mathfrak{F} = [\mathfrak{A} \olessthan \mathfrak{B}]\mathfrak{F}$, $[\mathfrak{B}](\mathfrak{F}[\mathfrak{A}]) = ([\mathfrak{B}]\mathfrak{F})[\mathfrak{A}]$.

In particular, the notation $[\mathfrak{B}]\mathfrak{F}[\mathfrak{A}]$ is not ambiguous.

[†] This counter example is due to François Maurel.

https://doi.org/10.1017/S096012950100336X

Proof. For instance, taking the last equation:

$$\begin{aligned} \ll [\mathfrak{B}](\mathfrak{F}[\mathfrak{A}]) \mid \mathfrak{C} \gg \quad &= \quad \ll \mathfrak{F}[\mathfrak{A}] \mid \mathfrak{C} \olessthan \mathfrak{B} \gg \\ &= \quad \ll \mathfrak{F} \mid \mathfrak{A} \olessthan (\mathfrak{C} \olessthan \mathfrak{B}) \gg \\ &= \quad \ll \mathfrak{F} \mid (\mathfrak{A} \olessthan \mathfrak{C}) \olessthan \mathfrak{B} \gg \\ &= \quad \ll [\mathfrak{B}]\mathfrak{F} \mid \mathfrak{A} \olessthan \mathfrak{C} \gg \\ &= \quad \ll ([\mathfrak{B}]\mathfrak{F})[\mathfrak{A}] \mid \mathfrak{C} \gg \end{aligned}$$

□

6.1.2. *The commutative adjunction*

Definition 31. (C. tensor product of designs) Let $\mathfrak{A}, \mathfrak{B}$ be positive designs. We define the *tensor product* $\mathfrak{A} \odot \mathfrak{B}$:

— If one of $\mathfrak{A}, \mathfrak{B}$ is a daimon, then $\mathfrak{A} \odot \mathfrak{B} = \mathfrak{Dai}$.

— Otherwise $\mathfrak{A}, \mathfrak{B}$ have respective first actions $(\langle\rangle, I)$ and $(\langle\rangle, J)$. If $I \cap J \neq \emptyset$, then $\mathfrak{A} \odot \mathfrak{B} = \mathfrak{Dai}$. Otherwise, replace in each chronicle of $\mathfrak{A}, \mathfrak{B}$ the first action $(\langle\rangle, I)$ or $(\langle\rangle, J)$ with $(\langle\rangle, I \cup J)$ so as to get $\mathfrak{A}', \mathfrak{B}'$. $\mathfrak{A} \odot \mathfrak{B} = \mathfrak{A}' \cup \mathfrak{B}'$.

In other words, in case of conflict, the flight is simply cancelled!

Theorem 14. (C. adjunction) Let $\mathfrak{F}, \mathfrak{A}, \mathfrak{B}$ be designs, $\mathfrak{F}$ negative, $\mathfrak{A}, \mathfrak{B}$ positive. Then there exists a *unique* negative design $(\mathfrak{F})\mathfrak{A}$ (not depending on $\mathfrak{B}$), such that

$$\ll \mathfrak{F} \mid \mathfrak{A} \odot \mathfrak{B} \gg = \ll (\mathfrak{F})\mathfrak{A} \mid \mathfrak{B} \gg \tag{68}$$

Proof. The proof is close to the proof of Theorem 13, p. 349. It amounts to constructing adequate $(\mathfrak{F})\mathfrak{A}_K$:

$I \cap K = \emptyset$: Let $(\mathfrak{F})\mathfrak{A}_K = \mathfrak{F}[\mathfrak{A}]_K = [\mathfrak{A}]\mathfrak{F}_K$.
$I \cap K \neq \emptyset$: Let $(\mathfrak{F})\mathfrak{A}_K = \mathfrak{Dai}$. □

Proposition 10. The operation $\odot$ is commutative, associative and its neutral element is $\mathfrak{One}$.

Corollary 10.1. $\mathfrak{F} \perp \mathfrak{A}$ iff the the chronicle $\langle(\langle\rangle, \emptyset)\rangle$ belongs to $(\mathfrak{F})\mathfrak{A}$.

Corollary 10.2.

$$((\mathfrak{F})\mathfrak{A})\mathfrak{B} = ((\mathfrak{F})\mathfrak{B})\mathfrak{A} = (\mathfrak{F})\mathfrak{A} \odot \mathfrak{B} = (\mathfrak{F})\mathfrak{B} \odot \mathfrak{A} \tag{69}$$

Remark 13. In ludics, the application of a function to an argument (commutative case)[†] is independent of the order. This is because everybody gets a location. In the usual spiritual logic, $f(a)$ means that a has been delocated so as to be put in front of f, and $f(a)(b), f(b)(a)$ are definitely distinct (that is, not isomorphic). Here, the arguments carry their own locations, so there is no possible mismatch. Returning to our aeroplane, the usual way is, say, to fill the rows from `1A` to `13D`[‡]. Then application of passengers to the aeroplane depends on the order of registration, but this is rational, that is, spiritual.

[†] The notation $(\mathfrak{F})\mathfrak{A}$ is in fact Krivine's notation for application in λ-calculus (Krivine 1990).
[‡] If such a number can be found in a plane...

From the locative viewpoint, every passenger comes with a definite seat and their order of appearance is irrelevant provided we adopt the commutative protocol.

6.1.3. *Non-commutative multiplicatives*

Definition 32. ($\oslash, \ltimes$) If $\mathbf{G}, \mathbf{H}$ are positive behaviours, one defines

$$\mathbf{G} \oslash \mathbf{H} = \{\mathfrak{A} \oslash \mathfrak{B}; \mathfrak{A} \in \mathbf{G}, \mathfrak{B} \in \mathbf{H}\}^{\perp\perp} \tag{70}$$

If $\mathbf{G}, \mathbf{H}$ are negative behaviours, one defines

$$\mathbf{G} \ltimes \mathbf{H} = \{\mathfrak{A} \oslash \mathfrak{B}; \mathfrak{A} \in \mathbf{G}^{\perp}, \mathfrak{B} \in \mathbf{H}^{\perp}\}^{\perp} \tag{71}$$

Proposition 11. Let $\mathbf{G}, \mathbf{H}$ be negative behaviours. Then

$$\mathfrak{F} \in \mathbf{G} \ltimes \mathbf{H} \Leftrightarrow \forall \mathfrak{A}(\mathfrak{A} \in \mathbf{G}^{\perp} \Rightarrow \mathfrak{F}[\mathfrak{A}] \in \mathbf{H}) \tag{72}$$

$$\mathfrak{F} \in \mathbf{G} \ltimes \mathbf{H} \Leftrightarrow \forall \mathfrak{B}(\mathfrak{B} \in \mathbf{H}^{\perp} \Rightarrow [\mathfrak{B}]\mathfrak{F} \in \mathbf{G}) \tag{73}$$

Proof. The proof is immediate. □

This is the ‘logical relation’ style of definition. This property (I mean *both* sides, one would not suffice!) is responsible for the associativity and distributivity of the connective.

Definition 33. (Boots) The negative design $\mathfrak{Boots}$ is defined as

$$\dfrac{\dfrac{}{\vdash}\,\maltese}{\langle\rangle \vdash}\,{\scriptstyle(\langle\rangle,\{\emptyset\})} \tag{74}$$

The negative behaviour $\perp$ is defined by $\perp = \mathfrak{Boots}^{\perp\perp}$. The positive behaviour $\mathbf{1}$ is defined by $\mathbf{1} = \mathfrak{One}^{\perp\perp}$.

Observe that $\mathbf{1} = \{\mathfrak{Dai}, \mathfrak{One}\}$ and that $|\perp| = \{\mathfrak{Boots}\}$.

Theorem 15. (Associativity, distributivity $\oslash, \ltimes$) The connective $\oslash$ is (strictly) associative, with neutral element $\mathbf{1}$ and absorber $\mathbf{0}$; it distributes over the locative ‘union’ $\uplus$.

The connective $\ltimes$ is (strictly) associative, with neutral element $\perp$ and absorber $\top$; it distributes over the intersection $\Cap$.

Proof. By duality it is enough to look at the negative side, that is, the connective $\ltimes$. $\mathfrak{F} \in \mathbf{G} \ltimes (\mathbf{G}' \ltimes \mathbf{G}'')$ iff for all $\mathfrak{A} \in \mathbf{G}^{\perp}$, $\mathfrak{A}'' \in \mathbf{G}''^{\perp}$, $[\mathfrak{A}''](\mathfrak{F}[\mathfrak{A}]) \in \mathbf{G}'$; $\mathfrak{F} \in (\mathbf{G} \ltimes \mathbf{G}') \ltimes \mathbf{G}''$ iff for all $\mathfrak{A} \in \mathbf{G}^{\perp}$, $\mathfrak{A}'' \in \mathbf{G}''^{\perp}$, $([\mathfrak{A}'']\mathfrak{F})[\mathfrak{A}] \in \mathbf{G}'$: associativity easily follows from Corollary 9.2, p. 349. We have $\mathfrak{F} \in \mathbf{G} \ltimes \bigcap \mathbf{H}_k$ iff for all $\mathfrak{A} \in \mathbf{G}^{\perp}$, $\mathfrak{F}[\mathfrak{A}] \in \bigcap \mathbf{H}_k$, *etc.*, so one gets distributivity to the right. Left distributivity makes use of the other adjunction $[\mathfrak{A}]\mathfrak{F}$. □

Remark 14. Indeed, associativity means that in Proposition 11, p. 351, one can replace $\mathbf{G}^{\perp}$ with any ethics $\mathbf{E}$ such that $\mathbf{E}^{\perp} = \mathbf{G}$ and still get

$$\mathfrak{F} \in \mathbf{G} \ltimes \mathbf{H} \Leftrightarrow \forall \mathfrak{A}(\mathfrak{A} \in \mathbf{E} \Rightarrow \mathfrak{F}[\mathfrak{A}] \in \mathbf{H}) \tag{75}$$

A similar remark can be made for the other multiplicatives.

6.1.4. *Commutative multiplicatives*

Definition 34. ($\odot, \bowtie$) If $\mathbf{G}, \mathbf{H}$ are positive behaviours, one defines

$$\mathbf{G} \odot \mathbf{H} = \{\mathfrak{A} \odot \mathfrak{B}; \mathfrak{A} \in \mathbf{G}, \mathfrak{B} \in \mathbf{H}\}^{\perp\perp} \tag{76}$$

If $\mathbf{G}, \mathbf{H}$ are negative behaviours, one defines

$$\mathbf{G} \bowtie \mathbf{H} = \{\mathfrak{A} \odot \mathfrak{B}; \mathfrak{A} \in \mathbf{G}^{\perp}, \mathfrak{B} \in \mathbf{H}^{\perp}\}^{\perp} \tag{77}$$

Proposition 12. Let $\mathbf{G}, \mathbf{H}$ be negative behaviours. Then

$$\mathfrak{F} \in \mathbf{G} \bowtie \mathbf{H} \Leftrightarrow \forall \mathfrak{A}(\mathfrak{A} \in \mathbf{G}^{\perp} \Rightarrow (\mathfrak{F})\mathfrak{A} \in \mathbf{H}) \tag{78}$$

$$\mathfrak{F} \in \mathbf{G} \bowtie \mathbf{H} \Leftrightarrow \forall \mathfrak{B}(\mathfrak{B} \in \mathbf{H}^{\perp} \Rightarrow (\mathfrak{F})\mathfrak{B} \in \mathbf{G}) \tag{79}$$

Proof. The proof is immediate. □

Theorem 16. (Associativity, distributivity $\odot, \bowtie$) The connective $\odot$ is (strictly) commutative, associative, with neutral element $\mathbf{1}$ and absorber $\mathbf{0}$; it distributes over the locative 'union' $\uplus$.

The connective $\bowtie$ is (strictly) commutative, associative, with neutral element $\perp$ and absorber $\top$; it distributes over the intersection $\cap\!\!\!\!+$.

Proof. The proof is similar to Theorem 15, p. 351. □

6.1.5. *The connectives* $\obar$, ∞ There is in fact yet another commutative multiplicative, $\obar$. It is based on the idea that in the case of a conflict for the same seat, the seat is given to a skunk[†] (the design $\mathfrak{Sk}$ above bias $k \in I \cap J$), but the flight is not cancelled: it is better to fly in the company of a skunk than not to fly at all.

Definition 35. (The other C. tensor) Let $\mathfrak{A}, \mathfrak{B}$ be positive designs. We define the *tensor product* $\mathfrak{A} \obar \mathfrak{B}$:

— If either $\mathfrak{A}, \mathfrak{B}$ is a daimon, then $\mathfrak{A} \obar \mathfrak{B} = \mathfrak{Dai}$.

— Otherwise $\mathfrak{A}, \mathfrak{B}$ have first actions $(\langle\rangle, I)$ and $(\langle\rangle, J)$, respectively. Replace in each chronicle of $\mathfrak{A}, \mathfrak{B}$ the first action $(\langle\rangle, I)$ or $(\langle\rangle, J)$ with $(\langle\rangle, I \cup J)$ so as to get $\mathfrak{A}', \mathfrak{B}'$. Then $\mathfrak{A} \obar \mathfrak{B}$ is the subset of $\mathfrak{A}' \cup \mathfrak{B}'$ consisting of:

  - Those chronicles of $\mathfrak{A}'$ whose second action (i, I') is such that $i \notin J$.
  - Those chronicles of $\mathfrak{B}'$ whose second action (j, J') is such that $j \notin I$.

$\obar$ and its dual ∞ are defined as for the previous multiplicatives.

Definition 36. ($\obar$, ∞) If $\mathbf{G}, \mathbf{H}$ are positive behaviours, one defines

$$\mathbf{G} \obar \mathbf{H} = \{\mathfrak{A} \obar \mathfrak{B}; \mathfrak{A} \in \mathbf{G}, \mathfrak{B} \in \mathbf{H}\}^{\perp\perp} \tag{80}$$

If $\mathbf{G}, \mathbf{H}$ are negative behaviours, one defines

$$\mathbf{G} \infty \mathbf{H} = \{\mathfrak{A} \obar \mathfrak{B}; \mathfrak{A} \in \mathbf{G}^{\perp}, \mathfrak{B} \in \mathbf{H}^{\perp}\}^{\perp} \tag{81}$$

[†] This means that the seat is given to anybody; up to incarnation it belongs to the skunk.

They enjoy the same style of properties: typically, there exists another commutative adjunction $\{.\}.$ corresponding to $\varobar$.

Proposition 13. Let $\mathbf{G}, \mathbf{H}$ be negative behaviours. Then

$$\mathfrak{F} \in \mathbf{G} \infty \mathbf{H} \Leftrightarrow \forall \mathfrak{A}(\mathfrak{A} \in \mathbf{G}^{\perp} \Rightarrow \{\mathfrak{F}\}\mathfrak{A} \in \mathbf{H}) \tag{82}$$

$$\mathfrak{F} \in \mathbf{G} \infty \mathbf{H} \Leftrightarrow \forall \mathfrak{B}(\mathfrak{B} \in \mathbf{H}^{\perp} \Rightarrow \{\mathfrak{F}\}\mathfrak{B} \in \mathbf{G}) \tag{83}$$

Proof. $\{\mathfrak{F}\}\mathfrak{A}$ is defined as in the proof of Theorems 13, p. 349, and 14, p. 350, by means of the $\{\mathfrak{F}\}\mathfrak{A}_K$: form a cut-net between $\mathfrak{F}_{I \cup K}$, the $\mathfrak{A}_i$ in $I - K$, and the $\mathfrak{S}\mathfrak{k}_k$ for $k \in I \cap K$; its base is $\vdash K - I$, and its normal form, partial or total, defines $\{\mathfrak{F}\}\mathfrak{A}_K$, up to the detail that the base must be changed to $\vdash K$, which is just a matter of replacing in every chronicle the action $(\langle\rangle, K - I)$ with $(\langle\rangle, K)$. □

Theorem 17. (Associativity, distributivity $\varobar$, ∞) The connective $\varobar$ is (strictly) commutative, associative, with neutral element **1** and absorber **0**; it distributes over the locative 'union' $\uplus$.

The connective ∞ is (strictly) commutative, associative, with neutral element $\perp$ and absorber $\top$; it distributes over the intersection $\mathbin{\cap\mkern-13mu-}$.

6.1.6. *The inclusions*

Theorem 18. (Inclusions)

$$\mathbf{G} \odot \mathbf{H} \subset \mathbf{G} \olessthan \mathbf{H} \subset \mathbf{G} \varobar \mathbf{H} \tag{84}$$

$$\mathbf{G} \infty \mathbf{H} \subset \mathbf{G} \ltimes \mathbf{H} \subset \mathbf{G} \bowtie \mathbf{H} \tag{85}$$

Proof. $\mathfrak{A} \varobar \mathfrak{B} \preceq \mathfrak{A} \olessthan \mathfrak{B} \preceq \mathfrak{A} \odot \mathfrak{B}$, and hence $(\mathfrak{A} \varobar \mathfrak{B})^{\perp} \subset (\mathfrak{A} \olessthan \mathfrak{B})^{\perp} \subset (\mathfrak{A} \odot \mathfrak{B})^{\perp}$, from which $(\mathbf{G} \varobar \mathbf{H})^{\perp} \subset (\mathbf{G} \olessthan \mathbf{H})^{\perp} \subset (\mathbf{G} \odot \mathbf{H})^{\perp}$, *etc.* □

The commutative product $\odot$ is therefore a subtype of the non-commutative products, which in turn are subtypes of $\varobar$.

6.1.7. *Multiplicatives and directory*

Proposition 14.

$$\P(\mathbf{G} \circledast \mathbf{H}) = \P\mathbf{G} \mathbin{\cup\mkern-13mu\times} \P\mathbf{H} \tag{86}$$

for $\circledast = \olessthan, \varobar$

$$\P(\mathbf{G} \odot \mathbf{H}) = \{I \cup J; I \in \P\mathbf{G}, J \in \P\mathbf{H}, I \cap J = \emptyset.\} \tag{87}$$

Proof. The proof is obvious. □

6.2. *Completeness properties*

6.2.1. *The projection lemma* What follows is called a lemma, for it is used in so many places. This is in fact one of the deepest results of the theory, the essential key to completeness.

Let $\mathbb{X} \subset \mathbb{N}$ be a reservoir. Any positive design $\mathfrak{A}$ with first action $(\langle\rangle, K)$ can be written uniquely as a tensor product $\mathfrak{D} \otimes \mathfrak{B}$[†] of a design $\mathfrak{D}$ starting with $(\langle\rangle, K \cap \mathbb{X})$ and a design $\mathfrak{B}$ starting with $(\langle\rangle, K - \mathbb{X})$.

Definition 37. (Projection) The design $\mathfrak{D}$ just introduced is called the *projection* of $\mathfrak{A}$ on $\mathbb{X}$ and written $\mathfrak{A} \upharpoonright \mathbb{X}$; we also define $\mathfrak{Dai} \upharpoonright \mathbb{X} = \mathfrak{Dai}$. If $\mathbf{E}$ is an ethics, $\{\mathfrak{A} \upharpoonright \mathbb{X}; \mathfrak{A} \in \mathbf{E}\}$ is called the *projection* of $\mathbf{E}$ on $\mathbb{X}$ and written $\mathbf{E} \upharpoonright \mathbb{X}$.

Theorem 19. (Projection) Assume $\mathbf{E}$ *connected*. Then projection commutes with biorthogonal:

$$\mathbf{E}^{\perp\perp} \upharpoonright \mathbb{X} = (\mathbf{E} \upharpoonright \mathbb{X})^{\perp\perp}.$$

Proof. Let $(\langle\rangle, K)$ be the first action of $\mathbf{E}$. Without loss of generality, we assume that $\mathbb{X} = I \subset K$. We must prove two inclusions:

$\mathbf{E}^{\perp\perp} \upharpoonright I \subset (\mathbf{E} \upharpoonright I)^{\perp\perp}$**:** Let $\mathfrak{F} \in (\mathbf{E} \upharpoonright I)^{\perp}$. Assuming $\mathfrak{F}$ material, we can replace its unique first negative action $(\langle\rangle, I)$ with $(\langle\rangle, K)$, so as to get $\mathfrak{F}'$ (in other words, we are 'using weakening'). If $\mathfrak{D} = \mathfrak{A} \upharpoonright I$, it is immediate that $\ll \mathfrak{A} \mid \mathfrak{F}' \gg = \ll \mathfrak{D} \mid \mathfrak{F} \gg$, and by letting $\mathfrak{A}$ range over $\mathbf{E}$ we conclude that $\mathfrak{F}' \in \mathbf{E}^{\perp}$. Now apply the same equation again with the weaker hypothesis $\mathfrak{A} \in \mathbf{E}^{\perp\perp}$, and conclude that $\mathfrak{D} \perp \mathfrak{F}$, that is, that $\mathfrak{D} \in (\mathbf{E} \upharpoonright I)^{\perp\perp}$.

$(\mathbf{E} \upharpoonright I)^{\perp\perp} \subset \mathbf{E}^{\perp\perp} \upharpoonright I$**:** Let $\mathfrak{F}' \in \mathbf{E}^{\perp}$, and let $\mathfrak{F} = (\mathfrak{F}')\mathfrak{B}$, where $\mathfrak{B} = \mathfrak{Ram}_{(\langle\rangle, K-I)}$ – see Example 14, p. 330. If $\mathfrak{A} \in \mathbf{E}$ and $\mathfrak{D} = \mathfrak{A} \upharpoonright I$, then $\mathfrak{A} \preceq \mathfrak{D} \odot \mathfrak{B}$, so $\mathfrak{F}' \perp \mathfrak{D} \odot \mathfrak{B}$, which yields, by adjunction, $\mathfrak{F} \perp \mathfrak{D}$, and we conclude that $\mathfrak{F} \in (\mathbf{E} \upharpoonright I)^{\perp}$. Now, if $\mathfrak{D} \in (\mathbf{E} \upharpoonright I)^{\perp\perp}$, we can now conclude that $\mathfrak{D} \odot \mathfrak{B} \perp \mathfrak{F}'$, hence $\mathfrak{D} \odot \mathfrak{B} \in \mathbf{E}^{\perp\perp}$, and since $(\mathfrak{D} \odot \mathfrak{B}) \upharpoonright I = \mathfrak{D}$, we conclude that $\mathfrak{D} \in \mathbf{E}^{\perp\perp} \upharpoonright I$. □

6.2.2. *Independence* Independence is a weak form of spirituality, which entails completeness.

Definition 38. (Independence) Two behaviours $\mathbf{G}, \mathbf{H}$ of the same polarity are said to be *independent* when the following holds: if $I, I' \in \P\mathbf{G}$ and $J, J' \in \P\mathbf{H}$ are such that $I \cup J = I' \cup J'$, then $I = I', J = J'$.

Let $\circledast$ be any of the tensors so far defined; all definitions are of the form

$$\mathbf{G} \circledast \mathbf{H} = \{\mathfrak{A} \circledast \mathfrak{B}; A \in \mathbf{G}, \mathfrak{B} \in \mathbf{H}\}^{\perp\perp} \tag{88}$$

So let $\mathbf{G} \copyright \mathbf{H} = \{\mathfrak{A} \circledast \mathfrak{B}; A \in \mathbf{G}, \mathfrak{B} \in \mathbf{H}\}$.

Theorem 20. (Independence property) If $\mathbf{G}, \mathbf{H}$ are positive and independent, then $\mathbf{G} \copyright \mathbf{H}$ is a complete ethics for $\mathbf{H} = \mathbf{G} \circledast \mathbf{H}$.

Proof. Write $\mathbf{G} \copyright \mathbf{H} = \bigcup_K (\bigcup_{I \cup J = K} \mathbf{G}_I \copyright \mathbf{H}_J)$. By the disjunction property, Theorem 11, p. 343, we are reduced to showing that $\bigcup_{I \cup J = K} \mathbf{G}_I \copyright \mathbf{H}_J$ is a behaviour. But the independence property precisely tells us that this union reduces to the simple form

[†] Anticipating our later notation, we simply use $\otimes$ when our four tensors collapse.

$\mathbf{G}_I \textcircled{c} \mathbf{H}_J$, that is, the theorem reduces to the connected case, so let us assume $\mathbf{G}, \mathbf{H}$ connected.

$\circledast = \odot$**:** If I, J are not disjoint, then $\mathbf{G} \odot \mathbf{H} = \mathbf{0}$. If I, J are disjoint, and if $\mathfrak{A} = \mathfrak{D} \odot \mathfrak{B} \in \mathbf{G} \odot \mathbf{H}$, then $\mathfrak{D} \in \mathbf{G} = \mathbf{G}^{\perp\perp}$ and $\mathfrak{B} \in \mathbf{H} = \mathbf{H}^{\perp\perp}$ by the Projection Lemma (Theorem 19, p. 354), so $\mathfrak{A} = \mathfrak{D} \odot \mathfrak{B} \in \mathbf{G} \textcircled{c} \mathbf{H}$.

$\circledast = \olessthan$**:** $\mathbf{G} \textcircled{c} \mathbf{H} = (\mathbf{G} \upharpoonright K - J) \textcircled{c} \mathbf{H}$. By the Projection Lemma (Theorem 19, p. 354 – in fact the second half of the proof), $\mathbf{G} \upharpoonright K - J$ is a behaviour, and we are reduced to the case I, J disjoint, which has already been treated.

$\circledast = \obar$**:** We easily obtain

$$\mathbf{G} \obar \mathbf{H} = (\mathbf{G} \upharpoonright I - J) \textcircled{c} \top^{+}_{I \cap J} \textcircled{c} (\mathbf{H} \upharpoonright J - I) \tag{89}$$

The central part is a positive skunk, whose only material inhabitant is $\mathfrak{Sk}_{(\langle\rangle, I \cap J)}$. From this we get $|\mathbf{G} \obar \mathbf{H}| = \mathbf{G} \textcircled{c} \mathbf{H}$ (in the previous cases, the incarnation was not needed: we directly got $\mathbf{G} \circledast \mathbf{H} = \mathbf{G} \textcircled{c} \mathbf{H}$). □

6.2.3. *The usual multiplicatives*

Definition 39. Two behaviours $\mathbf{G}, \mathbf{H}$ of the same polarity are *alien* when their reservoirs do not intersect, that is, when $\S\mathbf{G} \cap \S\mathbf{H} = \emptyset$.

Alienation is an important locative hypothesis, since:

— It makes the four multiplicatives collapse into a single one, $\otimes$.

— It ensures independence, and the completeness property attached to it.

Moreover, the connective $\otimes$ and its dual $\parr$[†], which are partial, get total variants, which only satisfy the properties up to isomorphism.

6.2.4. *The logical constant* **1** However, something must be noted: alienation does not imply disjunction, that is, $\S\mathbf{G} \cap \S\mathbf{H} = \emptyset$ does not imply $\P\mathbf{G} \cap \P\mathbf{H} = \emptyset$, because of the empty ramification. Hence the behaviour **1** will be problematic, since it is self-alien, but not self-disjoint. In particular when – in the style of Subsection 5.2.2, p. 342 – we use the delocations φ, ψ, **1** remains unchanged, for $\varphi(\emptyset) = \emptyset$. In other words, **1** cannot be treated like a usual formula, since $\varphi(\mathbf{1}) \oplus \psi(\mathbf{1})$ is not defined, and we are not at all accustomed to partial connectives!

To sum up, linear logic has no real multiplicative units! Indeed, this is not surprising since this is just the fact that one cannot define proof-nets for neutral elements without 'tying' the weakenings (Girard 1996). But tying the weakenings is the same as locating the neutrals, which cannot be located, *if they are quite neutral.* Old linear logic created neutral elements on the basis of approximate categorical isomorphisms, but these isomorphisms are wrong.

[†] As usual, $\mathbf{G} \multimap \mathbf{H}$ is short for $\mathbf{G}^{\perp} \parr \mathbf{H}$.

https://doi.org/10.1017/S096012950100336X

Of course booleans can no longer be defined by $\mathbf{1} \oplus \mathbf{1}$, but there is the possibility of close variants like $\varphi(\mathbf{U}) \oplus \psi(\mathbf{U})$, where $\mathbf{U}$ is the biorthogonal of the design:

$$\dfrac{\dfrac{\dfrac{}{\vdash 00}\,{\scriptstyle (00,\emptyset)}}{0 \vdash}\,{\scriptstyle (0,\{\{0\}\})}}{\vdash \langle\rangle}\,{\scriptstyle (\langle\rangle,\{0\})} \tag{90}$$

That is, $\mathbf{U} = \downarrow\uparrow\mathbf{1}$.

6.2.5. *The fax* We introduce, in addition to φ, ψ (Example 17, p. 337), two other delocations, φ', ψ':

$$\varphi'(\langle\rangle) = \psi'(\langle\rangle) = 2 \tag{91}$$

$$\varphi'(i * \sigma) = 2 * 3i * \sigma \tag{92}$$

$$\psi'(i * \sigma) = 2 * (3i + 1) * \sigma \tag{93}$$

Then the design

$$\dfrac{\ldots \qquad \dfrac{\begin{array}{c}\vdots\ \mathfrak{Fax}_{2*3i,3i} \\ \ldots 2 * 3i \vdash 3i \ldots\end{array}}{\vdash 3I, 2}\,{\scriptstyle (2,3I)} \qquad \ldots}{\langle\rangle \vdash}\,{\scriptstyle (\langle\rangle,\{3I \cup \{2\}; I \in \wp_f(\mathbb{N})\})} \tag{94}$$

which is a minor variant of the fax, belongs to the behaviour $\varphi(\mathbf{G}) \multimap \uparrow\varphi'(\mathbf{G})$ for any positive behaviour $\mathbf{G}$ of base $\vdash \langle\rangle$. In the case $\P\mathbf{G} = \{\{3,7\},\{4,7\}\}$, the incarnation of our design in $\varphi(\mathbf{G}) \multimap \uparrow\varphi'(\mathbf{G})$ is a design built from the pseudo-fax of Example 3, p. 310: one restricts to the values $\{3,7\},\{4,7\}$ of I.

Exercise 9. Construct inhabitants of the behaviours $\psi(\mathbf{G}) \multimap \uparrow\psi'(\mathbf{G})$, $\varphi(\mathbf{G}) \multimap \uparrow\psi'(\mathbf{G})$, $\psi(\mathbf{G}) \multimap \uparrow\varphi'(\mathbf{G})$.

6.3. *Sequents of behaviours*

6.3.1. *Definition and basic properties*

Definition 40. (Sequents of behaviours) Let $\Xi \vdash \Lambda$ be a pitchfork and let $\boldsymbol{\Xi}, \boldsymbol{\Lambda}$ be positive behaviours $\mathbf{G}_\sigma$ of respective bases $\vdash \sigma$ for $\sigma \in \Xi, \Lambda$. Then one defines the behaviour $\boldsymbol{\Xi} \vdash \boldsymbol{\Lambda}$ of base $\Xi \vdash \Lambda$ to be the orthogonal of the set of families $(\mathfrak{E}_\sigma)$ of designs $\mathfrak{E}_\sigma \in \mathbf{G}_\sigma$ for $\sigma \in \Xi$ (respectively, $\mathfrak{E}_\sigma \in \mathbf{G}_\sigma{}^\perp$ for $\sigma \in \Lambda$).

Theorem 21.

— The sequent of behaviours $\vdash$ is equal to $\mathbf{0} = \{\mathfrak{Dai}\}$, the only behaviour of base $\vdash$.
— The sequent of behaviours $\vdash \mathbf{G}$ is equal to $\mathbf{G}$.
— The sequent of behaviours $\mathbf{G} \vdash$ is equal to $\mathbf{G}^\perp$.
— $\mathfrak{D} \in \mathbf{G} \vdash \boldsymbol{\Lambda}$ iff for all $\mathfrak{E} \in \mathbf{G}$ $\quad [\![\mathfrak{D}, \mathfrak{E}]\!] \in \vdash \boldsymbol{\Lambda}$.
— $\mathfrak{D} \in \boldsymbol{\Xi} \vdash \mathbf{G}, \boldsymbol{\Lambda}$ iff for all $\mathfrak{E} \in \mathbf{G}^\perp$ $\quad [\![\mathfrak{D}, \mathfrak{E}]\!] \in \boldsymbol{\Xi} \vdash \boldsymbol{\Lambda}$.

6.3.2. *About the fax*

Proposition 15. Let ϕ, ϕ' be the delocations $\phi(\sigma) = \xi * \sigma, \phi'(\sigma) = \xi' * \sigma$ of the locus $\langle\rangle$ into disjoint loci ξ, ξ' and let $\mathbf{G}$ be a positive behaviour of base $\vdash \langle\rangle$. Then $\mathfrak{Fax}_{\xi,\xi'} \in \phi(\mathbf{G}) \vdash \phi'(\mathbf{G})$.

Proof. The proof is almost identical to 6.2.5, p. 356. □

6.3.3. *The category of behaviours* In what follows, ξ, ξ', ξ'' are disjoint loci, and the delocations ϕ, ϕ', ϕ'' are defined by $\phi(\sigma) = \xi * \sigma$ *etc.*

Definition 41. (Category of behaviours) The objects of the category **BV** are the behaviours based on $\vdash \langle\rangle$[†].

If $\mathbf{G}, \mathbf{H}$ are behaviours, a morphism from $\mathbf{G}$ to $\mathbf{H}$ is a *winning*[‡] material design in $\phi(\mathbf{G}) \vdash \phi''(\mathbf{H})$. Then the identity of $\mathbf{G}$ is the incarnation of $\mathfrak{Fax}_{\xi,\xi''}$ with respect to $\phi(\mathbf{G}) \vdash \phi''(\mathbf{G})$.

If $\mathfrak{D}, \mathfrak{E}$ are morphisms from $\mathbf{G}$ to $\mathbf{H}$ and from $\mathbf{H}$ to $\mathbf{K}$, respectively, the composition $\mathfrak{F} = \mathfrak{E} \circ \mathfrak{D}$ is the morphism from $\mathbf{G}$ to $\mathbf{K}$ defined by

$$\mathfrak{F} = |[\![\phi'\phi''^{-1}(\mathfrak{D}), \phi'\phi^{-1}(\mathfrak{E})]\!]| \tag{95}$$

The definition abstracts from the location (the base) by the choice of $\vdash \langle\rangle$. But then there is a technical problem with morphisms: $\langle\rangle \vdash \langle\rangle$ is not a pitchfork! This is why we use the delocations ϕ, ϕ''; in order to compose, an intermediate location is used. When we compose material designs, the result needs not be material: this explains the use of $|.|$ in the definition.

7. Quantifiers

Quantifiers have already been introduced in Chapter 5, p. 339, and written $\bigcap_k, \uplus_k$. In what follows we investigate them with a different spirit, and this is why we will change our notation.

Here we have a delicate point of terminology: one of the discoveries of ludics is that the word 'quantifier' – say universal – corresponds to two completely different approaches that the tradition could hardly separate.

First-order quantification: A big conjunction, which may be uniform in some sense: this is the viewpoint of model-theory, of German style proof-theory, *etc.*

Second-order quantification: An intersection: this is the viewpoint of the forgetful interpretation of system $\mathbb{F}$, and the one we adopt here.

The first-order approach is spiritual (that is, involves a lot of delocations and is up to isomorphism), whereas the second-order approach is locative. The dominant approach to

[†] One should not believe that this is a category of positive behaviours; in reality the object **G** stands for **G** and its negation.

[‡] Winning is defined in Section 9.1, p. 373. If you do not know what winning is about, just ignore it: requiring morphisms to be winning has to do with preservation of... winning, that is, truth.

https://doi.org/10.1017/S096012950100336X

logic being spiritual, the tendency was to try to treat second-order quantification as a poor relation of first-order quantification... to the extent that one hardly understands why the first-order case is complete (or enjoys the subformula property) and the second-order case is incomplete! The word 'quantifier' is definitely not used at all in the same sense in these two occurrences... unfortunately, it is too late to produce neologisms that would distinguish the two uses!

In this monograph, we shall not be concerned with first-order quantification, although the chapter on uniformity introduces the material that should suffice to define a uniform conjunction. We shall only consider the quantifier-as-intersection, which corresponds, in decreasing order of interest to the following cases:

Higher-order quantification: The second-order case and its straightforward generalisations to third-order, fourth-order, *etc.*

Intersection types: Typically a binary intersection, usually carefully kept aside from logic for want of any spiritual (truth values, categorical models, *etc.*) interpretation.

First-order quantification: We can try to mistreat first-order quantification and treat it in our spirit, that is, as an intersection indexed by the domain of interpretation. This plain treason is not worse than the current habit of treating second-order quantification as a big conjunction[†]: it will cause the failure of completeness, but bring prenex forms as a compensation, so it might be worth the attempt to try and see.

To summarise: there are two diverging traditions concerning quantification: one spiritual; one locative – locative before the expression was created. In this section, we investigate the locative quantifier, that is, the quantifier that does not follow the truth tables, that ignores category theory. The question is clear: does this become a mess, or do we get something nice out of it? In fact something wonderful arises from the unexpected shock when locative quantification and spiritual connectives meet: these operations commute – sometimes beyond what seems reasonable, that is, up to the violation of certain classical principles.

These commutations induce *prenex forms*: typically any second-order proposition in $\mathbf{MALL}_2$ (logic without exponentials[‡]) is *equal* to its prenex form, for example, we have $(\forall X P[X]) \otimes (Q \,\&\, \exists Y R[Y])$ is literally the same as $\forall X \exists Y (P[X] \otimes (Q \,\&\, R[Y])$ and $\exists Y \forall X (P[X] \otimes (Q \,\&\, R[Y])$. This is potentially of tremendous interest.

As usual the base is $\vdash \langle\rangle$ or $\langle\rangle \vdash$.

7.1. *Basic definitions*

7.1.1. *Universal quantification*

Definition 42. Let $\mathbf{G}_d$ be a family of behaviours of the *same* polarity, indexed by a set $\mathbb{D}$ of any cardinality. Then we define $\forall d \in \mathbb{D}\, \mathbf{G}_d$ as the intersection $\bigcap_{d \in \mathbb{D}} \mathbf{G}_d$.

[†] Big conjunctions; small brains.

[‡] This fails in the case of exponentials – see Subsection 7.2.4, p. 362.

https://doi.org/10.1017/S096012950100336X

The indexing set can be of any cardinality, typically 0 (thus yielding $\top^{\epsilon}$), 2 (an intersection type), $\aleph_0$ (first-order quantification) $2^{\aleph_0}$ (second-order quantification). The index set is spiritual, that is, is only up to bijection; a more exact (because it is locative), but less convenient notation would be $\forall\mathscr{X}$ where $\mathscr{X}$ is a set of behaviours. When we feel relaxed and nobody from algebraic logic is lurking around, we just write $\forall d$.

7.1.2. *Existential quantification*

Definition 43. Let $\mathbf{G}_d$ be a family of behaviours of the *same* polarity, indexed by a set $\mathbb{D}$ of any cardinality. Then we define $\exists d \in \mathbb{D}\ \mathbf{G}_d = (\bigcup_{d\in\mathbb{D}} \mathbf{G}_d)^{\perp\perp}$.

The life of the unfortunate existentials will be difficult: there is no decent way to give a complete ethics for $\exists d$. When the index set I is something like the set of all behaviours (that is, second-order quantification), Cantor's theorem[†] will annoy you badly.

But existential incompleteness is of a non-enumerative nature: the existential quantifier is the archetype, the paragon of incompleteness, but problems start already when I has two elements, and this is nothing to do with Pr. Dr. Münchhausen and his notorious method of getting out of water. The evidence for incompleteness lies in the politically incorrect prenex forms below, typically $\exists d \forall e(\varphi(\mathbf{A}_d) \multimap {\uparrow}\varphi'(\mathbf{A}_e))$: there is no way to remove the biorthogonal involved in the existential quantifier as soon as $\sharp(\mathbb{D}) \geqslant 2$. If you are not convinced of the incompleteness of $\exists$, observe that for any ethics $\mathbf{E}$ we have:

$$\mathbf{E}^{\perp\perp} = \exists\mathfrak{E} \in \mathbf{E}\ \{\mathfrak{E}'; \mathfrak{E} \preceq \mathfrak{E}'\} \tag{96}$$

In other words, the biorthogonal is nothing but an instance of the existential quantifier, and if incompleteness exists, the existential quantifier must be blamed for it.

7.2. *Shocking equalities and prenex forms*

7.2.1. *The commutation theorem*

Theorem 22. (Commutation) $\forall d$ commutes with all operations, except $\exists$; in particular, ludics admits prenex forms.

Basically, $\forall$ commutes with all complete connectives. The general idea is to use the completeness of the ethics $\mathbf{G}_d$ to replace $\forall d{\mathbf{G}_d}^{\perp\perp}$ with $\forall d\mathbf{G}_d$: everything is almost immediate. The proof is a list of cases of unequal interest. The most important commutations are $\forall/\oplus$ (dually $\exists/\&$) and $\forall/\otimes$ (dually $\exists/\parr$). Indeed the theorem only states the unary commutations, for example:

$$\forall d(\mathbf{G}_d \oplus \mathbf{H}) = (\forall d\mathbf{G}_d) \oplus \mathbf{H} \tag{97}$$

But we in fact get double commutations – so strong that the commutation $\forall/\oplus$ even contradicts classical logic!

[†] It is amusing to see that Gödel's theorem, which was inspired by Cantor's theorem, here returns to the original *matrix*.

The theorem has been phrased in a very charming hand-waving style, but one has to be precise about its contents. I take one example, precisely the commutation $\forall/\oplus$. Here the connective $\oplus$ is the plain spiritual one, namely

$$\mathbf{G} \oplus \mathbf{H} = (\varphi(\mathbf{G}) \cup \psi(\mathbf{H}))^{\perp\perp} \tag{98}$$

where the delocations φ, ψ have been introduced (p. 337). In particular, when we form $\mathbf{G}_d \oplus \mathbf{H}_d$, the delocations at work are independent of d. This is not a trick, this is:

— The most natural thing to do.

— The only reasonable one: note that the default $\mathbb{D}$ is the domain of the second-order quantifier, that is, the set of all behaviours, whose cardinality is likely to be $2^{2^{\aleph_0}}$, whereas there are at most $2^{\aleph_0}$ delocations, and $\aleph_0$ pairwise disjoint delocations.

— Reminiscent of plain realisability: the realiser of a disjunction is of the form $1*$ or $2*$, where the numbers $1, 2$ are fixed delocations, and, by the way, the intuitionistic implication $\forall X(A[X] \vee B[X]) \Rightarrow (\forall X A[X]) \vee (\forall X B[X])$ is realisable.

Technically speaking, the commutation rests on two pillars:

— The disjunction property, Theorem 11, p. 343, which enables us to remove the biorthogonal in (98).

— The fact that the reservoirs $\mathbb{X} = \varphi(\mathbb{N})$ and $\mathbb{Y} = \psi(\mathbb{N})$ are disjoint, hence the two disjuncts are clearly distinguished by the splitting $\mathbb{N} \supset \mathbb{X} \cup \mathbb{Y}$.

For notational simplicity, we shall return to our usual conventions (see Subsection 5.2.2) and do not use the delocations φ, ψ. This means that we assume that, for all $d \in \mathbb{D}$ $\S\mathbf{G}_d \subset \mathbb{X}, \S\mathbf{H}_d \subset \mathbb{Y}$.

7.2.2. *The commutations of* $\forall$ All equations are proved by showing the non-trivial inclusion, that is, the one from left to right.

• $\forall/\downarrow$

$$\forall d \downarrow\mathbf{G}_d = \downarrow\forall d \mathbf{G}_d \tag{99}$$

• $\forall/\uparrow$

$$\forall d \uparrow\mathbf{G}_d = \uparrow\forall d \mathbf{G}_d \tag{100}$$

Proof. The two commutations are immediate. They are essential, since they have to do with the change of polarity. Observe that the $\forall$ on both sides of each equation is of different polarity. □

• $\forall/\oplus$

$$\forall d(\mathbf{G}_d \oplus \mathbf{H}_d) = (\forall d \mathbf{G}_d) \oplus (\forall d \mathbf{H}_d) \tag{101}$$

Proof. If $e \in \mathbb{D}$ and if $\mathfrak{D} \in \forall d(\mathbf{G}_d \oplus \mathbf{H}_d)$ is proper, then $\mathfrak{D} \in \mathbf{G}_e \cup \mathbf{H}_e$, and by the disjunction property belongs to one of $\mathbf{G}_e$, $\mathbf{H}_e$. The same holds for any $e' \in \mathbb{D}$, and the locative hypotheses force $\mathfrak{D}$ to be on the same side of the disjunction. □

Observe how this equation is violently anti-classical: $\forall$ commutes with the multiplicative disjunction $⅋$ in a more polite way!

- $\forall/\otimes$

$$\forall d(\mathbf{G}_d \otimes \mathbf{H}_d) = (\forall d\mathbf{G}_d) \otimes (\forall d\mathbf{H}_d) \tag{102}$$

Proof. If $\mathfrak{D} \in \forall d(\mathbf{G}_d \otimes \mathbf{H}_d)$, then its projection on $\mathbb{X}$ belongs to $\forall d\mathbf{G}_d$ for all d, hence $\forall d\mathbf{G}_d$, *etc.* □

- $\forall/\&$

$$\forall d(\mathbf{G}_d \,\&\, \mathbf{H}_d) = (\forall d\mathbf{G}_d) \,\&\, (\forall d\mathbf{H}_d) \tag{103}$$

- $\forall/\parr$

$$\forall d(\mathbf{G}_d \parr \mathbf{H}) = (\forall d\mathbf{G}_d) \parr \mathbf{H} \tag{104}$$

- $\forall/\forall$

$$\forall d \in \mathbb{D}\, \forall e \in \mathbb{E}\; \mathbf{G}_{d,e} = \forall (d,e) \in \mathbb{D} \times \mathbb{E}\; \mathbf{G}_{d,e} \tag{105}$$

Proof. The last three commutations are no surprise at all; they correspond to standard spiritual commutations: $\forall$ is traditionally negative, hence commutes with $\&, \parr$ and itself. □

7.2.3. *The commutations of* $\exists$

- $\exists/\uparrow$

$$\exists d \uparrow \mathbf{G}_d = \uparrow \exists d\mathbf{G}_d \tag{106}$$

- $\exists/\downarrow$

$$\exists d \downarrow \mathbf{G}_d = \downarrow \exists d\mathbf{G}_d \tag{107}$$

- $\exists/\&$

$$(\exists d\mathbf{G}_d) \,\&\, (\exists d\mathbf{H}_d) = \exists d(\mathbf{G}_d \,\&\, \mathbf{H}_d) \tag{108}$$

- $\exists/\parr$

$$\exists d(\mathbf{G}_d \parr \mathbf{H}_d) = (\exists d\mathbf{G}_d) \parr (\exists d\mathbf{H}_d) \tag{109}$$

- $\exists/\oplus$

$$\exists d(\mathbf{G}_d \oplus \mathbf{H}_d) = (\exists d\mathbf{G}_d) \oplus (\exists d\mathbf{H}_d) \tag{110}$$

- $\exists/\otimes$

$$\exists d(\mathbf{G}_d \otimes \mathbf{H}) = (\exists d\mathbf{G}_d) \otimes \mathbf{H} \tag{111}$$

- $\exists/\exists$

$$\exists d \in \mathbb{D}\, \exists e \in \mathbb{E}\; \mathbf{G}_{d,e} = \exists (d,e) \in \mathbb{D} \times \mathbb{E}\; \mathbf{G}_{d,e} \tag{112}$$

7.2.4. *Miscellaneous* Indeed, a quantifier commutes with everything but a quantifier of the opposite kind. In particular:

$$\forall d \in \mathbb{D}\, \exists e \in \mathbb{E}\, \mathbf{G}_{d,e} = \exists f \in \mathbb{E}^{\mathbb{D}}\, \forall d\, \mathbf{G}_{d,f(d)} \tag{113}$$

is badly wrong, even for $\mathbb{D}$ finite: the reason is that $\exists$ is badly incomplete, so that – in set theoretical terms – we are not quite dealing with $\forall\exists$, but with $\forall\exists^{\perp\perp}$. Our shocking commutations (essentially $\forall/\oplus$ and$\forall/\otimes$) actually come from the completeness of $\oplus$ and $\otimes$.

Among the most unexpected true principles of logic stands

$$\exists d \forall e (\varphi(\mathbf{G}_d) \multimap \uparrow \varphi'(\mathbf{G}_e)) \tag{114}$$

obtained through commutations from the principle

$$(\forall d \mathbf{G}_d \multimap \forall e \uparrow \mathbf{G}_e) \tag{115}$$

Existential quantifiers do not enjoy the existence property, but who cares? The useful existence property deals with numerical quantification – which is not a quantifier in our sense – and which enjoys the existence property.

Unfortunately, it is unlikely that prenex forms persist – at least in this straightforward form – for exponentials. The reason is simple, this would imply the equality $\forall d \neg\neg \mathbf{G}_d = \neg\neg\forall d \mathbf{G}_d$. But surely (we forget shifts, delocations *etc.*) we have $\forall X \neg\neg(X \oplus \neg X)$, and we would obtain $\neg\neg(\forall X\ X \oplus \forall X\ \neg X)$, which is a contradiction.

7.3. *The usual quantifiers*

7.3.1. *First-order quantification* Our notion of a first-order quantifier is very simple, just take an infinite denumerable domain $\mathbb{D}$, and consider *behavioural predicates*, that is, families of behaviours of the same base indexed by $\mathbb{D}$. The most typical such family is equality, a binary predicate.

Definition 44. (Equality) For $d, d' \in \mathbb{D}$, we define the positive behaviour $\boldsymbol{=}_{d,d'}$

— $\boldsymbol{=}_{d,d'} := \mathbf{1}$ if $d = d'$.

— $\boldsymbol{=}_{d,d'} := \mathbf{0}$ if $d \neq d'$.

Our approach to first-order is extremely simple, but contradicts the tradition of predicate calculus, just consider that prenex forms like

$$\exists d \forall e (\varphi(\mathbf{G}_d) \multimap \uparrow \varphi'(\mathbf{G}_e)) \tag{116}$$

are not accepted in the usual first-order logic, not to mention

$$\forall d (\mathbf{G}_d \oplus \mathbf{H}_d) = (\forall d \mathbf{G}_d) \oplus (\forall d \mathbf{H}_d) \tag{117}$$

So there is a mismatch between tradition and ludics.

7.3.2. *Second-order quantification* We only consider second-order propositional quantification. Second-order predicate quantification is already a higher-order quantification and does not deserve an independent reflection.

https://doi.org/10.1017/S096012950100336X

A propositional variable stands for the unknown behaviour. However, certain details must be clarified:

Polarity: Either a variable ranges over all positive behaviours, or it ranges over all negative behaviours. We decide that propositional variables range over positive behaviours: negations of variables will range over negative behaviours, and there is no loss of generality.

Location: In the usual syntax a given variable occurs several times, both positively and negatively, which means that our unknown behaviour has several locations, which is impossible. So we decide that variables range over positive behaviours *of base* $\vdash \langle\rangle$. Various occurrences will be handled through appropriate delocations.

Quantification: Second-order quantifications $\forall\mathbf{G}$ and $\exists\mathbf{G}$ are just intersections indexed by this specific set of behaviours.

For instance, the design of Subsection 6.2.5, p. 356, belongs to $\forall\mathbf{G}(\varphi(\mathbf{G})\multimap\uparrow\varphi'(\mathbf{G}))$.

Observe that nobody forces us to quantify over 'all' behaviours. We could for instance quantify over all 'subtypes' of a given behaviour, that is, form $\forall\mathbf{G}\subset\mathbf{H}_0\ \mathbf{\Phi}[\mathbf{G}]$ *etc.* We could as well restrict to behaviours enjoying certain peculiarities. For instance it can be useful to quantify over those behaviours $\mathbf{G}$ whose directory $\P\mathbf{G}$ does not contain $\emptyset$, which we shall do in Chapters 10– 11.

Second-order quantification differs in spirit from the first-order case: Gödel's theorem shows that incompleteness should be expected as soon as second-order existentials occur. So there is nothing to fix with our prenex forms, as long as we deal with second-order: these prenex forms give rise to implications involving second-order existentials, and the incompleteness of such formulas is expected. The only novelty with respect to Gödel is that we do not need to diagonalise to get an artificial counterexample, we have just natural, immediate, and *useful* counterexamples to completeness.

7.3.3. *Higher-order quantification*

```
(higherorder-my-file 'subsection ''7.3.2'').
```

8. Uniformity

Certain logical connectives (exponentials, spiritual quantification[†]) are *uniform*: this means that the fact that two designs are 'similar' – for instance isomorphic – plays a role in the definition of the connective. Nothing so far introduced is able to cope with uniformity.

What is essentially missing is the possibility of equipping a behaviour with a *partial equivalence relation*, a 'PER'.

Definition 45. A *partial equivalence relation* (PER) on a set X is a binary relation $\cong$ on X that is:

[†] Typically, first order quantifiers in the usual view: they are 'uniform infinite &'.

Symmetric: $x \cong y \Rightarrow y \cong x$.
Transitive: $x \cong y$ and $y \cong z \Rightarrow x \cong z$.

As a consequence, $\cong$ is:

Weakly reflexive: $x \cong y \Rightarrow x \cong x$ and $y \cong y$

The set $\{x; x \cong x\}$ is the *support* of $\cong$.

The new notion 'behaviour + PER' is called a *bihaviour*. There is a 'forgetful functor' from bihaviours to behaviours, which commutes with all connectives so far defined.

According to our general methodological bias, all notions must be introduced interactively: a PER $\cong$ on the behaviour $\mathbf{G}$ should appear as the 'orthogonal' of a PER $\cong^{\perp}$ on $\mathbf{G}^{\perp}$. The orthogonality between $\cong$ and $\cong^{\perp}$ must refer to a PER on the sequent of behaviours $\vdash$ that contains only one element, the daimon $\maltese$, and the theory is going to be trivial, this is the problem with *yes men*... unless we can produce a second element in $\vdash$, which can only be the partial design $\mathfrak{Fid}$.

As soon as we allow $\mathfrak{Fid}$ in this particular case, we must accept that $\cong$ relates *partial* elements of a behaviour $\mathbf{G}$: for this, we must define what is a partial design *with respect to a given behaviour* $\mathbf{G}$. This generalises our original definition of partiality: $\mathfrak{Fid}$ appears as the only design that is quite partial with respect to *all* positive behaviours.

8.1. *Partial designs*

For simplicity of exposition, we assume that the base $\Xi \vdash \Sigma$ is atomic.

Definition 46. (Partial behaviours) Let $\mathbf{E}$ be an ethics; the *partial ethics*[†] associated with $\mathbf{E}$ is the set $\mathbf{E}^p$ of all designs (total or partial in the absolute sense) included in some design of $\mathbf{E}$. The elements of $\mathbf{E}^p$ are the *partial* designs of $\mathbf{E}$; if we want to stress the fact that a partial design belongs to $\mathbf{E}$, we speak of a *total* design.

Typical examples of partial designs are given by $\mathfrak{Fid}$, and by the *slices* of a given total design of $\mathbf{G}$.

Definition 47. We define the equivalence relation $\equiv_{\mathbf{G}}$ on $\mathbf{G}^p$ by

$$\mathfrak{D} \equiv_{\mathbf{G}} \mathfrak{D}' \Leftrightarrow \forall \mathfrak{E} \in \mathbf{G}^{\perp p} \ \ll \mathfrak{D} \mid \mathfrak{E} \gg = \ll \mathfrak{D}' \mid \mathfrak{E} \gg \tag{118}$$

Definition 48. (Incarnation) The *incarnation* $|\mathfrak{D}|_{\mathbf{G}}$ of $\mathfrak{D}$ with respect to $\mathbf{G}$ is the smallest design – with respect to inclusion – $\mathfrak{E} \subset \mathfrak{D}$ such that $\mathfrak{E} \equiv_{\mathbf{G}} \mathfrak{D}$.

The existence of incarnation is a consequence of the Stability Theorem (Theorem 3, p. 331). The next exercise generalises Exercise 5, p. 335.

Exercise 10. If $\mathfrak{D}, \mathfrak{D}' \in \mathbf{G}^p$ are material and distinct, prove the existence of a (total) $\mathfrak{E} \in \mathbf{G}^{\perp}$ such that:

1 At least one of $\mathfrak{D}, \mathfrak{D}'$ is orthogonal to $\mathfrak{E}$.
2 The disputes (partial or total, see Remark 5, p. 326) $[\mathfrak{D} \rightleftharpoons \mathfrak{E}]$ and $[\mathfrak{D}' \rightleftharpoons \mathfrak{E}]$ are distinct.

[†] For a behaviour $\mathbf{G}$, the *partial behaviour* $\mathbf{G}^p$.

Proposition 16. Let $\mathfrak{D}, \mathfrak{D}' \in \mathbf{G}^p$. Then

$$\mathfrak{D} \equiv_{\mathbf{G}} \mathfrak{D}' \;\Leftrightarrow\; |\mathfrak{D}|_{\mathbf{G}} = |\mathfrak{D}'|_{\mathbf{G}} \tag{119}$$

Proof. The condition is obviously sufficient. For the converse, assume that the partial $\mathfrak{D}, \mathfrak{D}' \in \mathbf{G}^p$ are *material* and distinct and that $\mathfrak{D} \equiv_{\mathbf{G}} \mathfrak{D}'$. By Exercise 10 just above, there exists a design $\mathfrak{E}' \in \mathbf{G}^{\perp}$ such that $[\mathfrak{D} \rightleftharpoons \mathfrak{E}'] \neq [\mathfrak{D}' \rightleftharpoons \mathfrak{E}']$. Moreover, $\mathfrak{E}'$ is orthogonal to one of $\mathfrak{D}, \mathfrak{D}'$, since $\mathfrak{D} \equiv_{\mathbf{G}} \mathfrak{D}'$, $\mathfrak{E}'$ is orthogonal to both of $\mathfrak{D}, \mathfrak{D}'$ and the two disputes are total. The two disputes first differ because of the choice of distinct positive actions κ, κ', occurring as $\mathfrak{c} * \kappa \in \mathfrak{D}, \mathfrak{c} * \kappa' \in \mathfrak{D}'$; assume that, say, κ is proper. The normalisation of $\llbracket \mathfrak{D}, \mathfrak{E} \rrbracket$ 'consumes' a chronicle $\mathfrak{c}' * \widetilde{\kappa} \in \mathfrak{E}'$ and let $\mathfrak{E} \subset \mathfrak{E}'$ be obtained from $\mathfrak{E}$ by removing all chronicles $\mathfrak{c}' * \widetilde{\kappa} * \mathfrak{c}''$. Then $\ll \mathfrak{D} \mid \mathfrak{E} \gg$ diverges, whereas $\ll \mathfrak{D}' \mid \mathfrak{E} \gg$ converges, which is a contradiction. □

Corollary 16.1. Let $\mathbf{G}$ be a behaviour and $\mathfrak{D}, \mathfrak{D}' \in \mathbf{G}^p$. Then $\mathfrak{D} \equiv_{\mathbf{G}} \mathfrak{D}'$ iff for all $\mathfrak{E}, \mathfrak{E}' \in \mathbf{G}^{\perp p}$ such that $\mathfrak{E} \equiv_{\mathbf{G}^{\perp}} \mathfrak{E}'$

$$\ll \mathfrak{D} \mid \mathfrak{E} \gg \; = \; \ll \mathfrak{D}' \mid \mathfrak{E}' \gg \tag{120}$$

8.2. *Bihaviours*

8.2.1. *Biethics*

Definition 49. A *biethics* is a pair $(\mathbf{E}, \cong)$ of an ethics $\mathbf{E}$ and a PER $\cong$ on $\mathbf{E}^p$ such that:

Positive base: If $\mathfrak{D} \cong \mathfrak{E}$, then

— If $\mathfrak{E} = \mathfrak{Fid}$, then $\mathfrak{D} = \mathfrak{Fid}$.

— If $\mathfrak{D} = \mathfrak{Dai}$, then $\mathfrak{E} = \mathfrak{Dai}$.

Negative base: $\mathfrak{Sk} \cong \mathfrak{Sk}$, $\mathfrak{Dai}^{-} \cong \mathfrak{Dai}^{-}$.

Definition 50. Let $(\mathbf{E}, \cong)$ be a biethics. Its *orthogonal* is the pair $(\mathbf{E}^{\perp}, \cong^{\perp})$ defined by $\mathfrak{E} \cong^{\perp} \mathfrak{E}'$ iff for all $\mathfrak{D} \cong \mathfrak{D}'$

$$\ll \mathfrak{D} \mid \mathfrak{E} \gg \; = \; \ll \mathfrak{D}' \mid \mathfrak{E}' \gg \tag{121}$$

Proposition 17. If $(\mathbf{E}, \cong)$ is a biethics, then $(\mathbf{E}^{\perp}, \cong^{\perp})$ is a biethics too.

Proof. It is (almost) immediate that $\cong^{\perp}$ is a PER. Moreover, note that, with $\mathfrak{D}$ positive and distinct from $\mathfrak{Dai}, \mathfrak{Fid}$:

$$\ll \mathfrak{Sk} \mid \mathfrak{Fid} \gg \; = \; \ll \mathfrak{Dai}^{-} \mid \mathfrak{Fid} \gg \; = \; \ll \mathfrak{Sk} \mid \mathfrak{D} \gg \; = \mathfrak{Fid} \tag{122}$$

$$\ll \mathfrak{Sk} \mid \mathfrak{Dai} \gg \; = \; \ll \mathfrak{Dai}^{-} \mid \mathfrak{Dai} \gg \; = \; \ll \mathfrak{Dai}^{-} \mid \mathfrak{D} \gg \; = \mathfrak{Dai} \tag{123}$$

From this it follows that:

E positive: $\mathfrak{Sk} \cong^{\perp} \mathfrak{Sk}$ since $\mathfrak{Dai}$ can only be equivalent to itself. Similarly, $\mathfrak{Dai}^{-} \cong^{\perp} \mathfrak{Dai}^{-}$, since $\mathfrak{Fid}$ can only be equivalent to itself.

E negative: $\mathfrak{Fid} \cong^{\perp} \mathfrak{D}$ and $\mathfrak{Fid} \cong^{\perp} \mathfrak{Dai}$ are impossible because of $\mathfrak{Dai}^{-} \cong \mathfrak{Dai}^{-}$; $\mathfrak{Dai} \cong^{\perp} \mathfrak{D}$ is impossible because of $\mathfrak{Sk} \cong \mathfrak{Sk}$. By the way, observe that $\mathfrak{Fid} \cong^{\perp} \mathfrak{Fid}$ and $\mathfrak{Dai} \cong^{\perp} \mathfrak{Dai}$. □

8.2.2. *Biincarnation*

Definition 51. (Bihaviours) A *bihaviour* is a biethics $(\mathbf{G}, \cong)$ equal to its biorthogonal.

Example 25. By Corollary 16.1, p. 365, $(\mathbf{G}, \equiv_{\mathbf{G}})$ is a bihaviour when $\mathbf{G}$ is a behaviour, which shows that the new notion encompasses the old one. In particular, the constants $\mathbf{0}, \mathbf{1}, \bot, \top$ can be seen as bihaviours. Besides ordinary behaviours, bihaviours enable one to speak of the 'symmetric sum', see Subsection 8.3.4, the 'symmetric tensor product', see Subsection 8.4.4, see also Subsection 8.2.3 below.

Proposition 18. Let $(\mathbf{G}, \cong)$ be a bihaviour. Then for all $\mathfrak{D}, \mathfrak{D}' \in \mathbf{G}^p$:

$$\mathfrak{D} \cong \mathfrak{D}' \Leftrightarrow |\mathfrak{D}| \cong |\mathfrak{D}'| \tag{124}$$

Proof. The proof is left as an easy exercise. □

Definition 52. (Biincarnation) If $(\mathbf{G}, \cong)$ is a bihaviour and $\mathfrak{D} \cong \mathfrak{D} \in \mathbf{G}^p$, its *biincarnation* $\|\mathfrak{D}\|$ is the smallest design $\mathfrak{E} \in \mathbf{G}^p$ such that $\mathfrak{E} \subset \mathfrak{D}$ and $\mathfrak{D} \cong \mathfrak{E}$. A design equal to its biincarnation is said to be *bimaterial* or *biincarnated*.

Remark 15. The existence of the biincarnation is an immediate application of the Stability Theorem (Theorem 3, p. 331). Moreover, since $\mathfrak{D} \cong |\mathfrak{D}|$, we get

$$\|\mathfrak{D}\| \subset |\mathfrak{D}| \subset \mathfrak{D} \tag{125}$$

Also, if $\|\mathfrak{D}\| \subset \mathfrak{E} \subset \mathfrak{D}$, then $\|\mathfrak{E}\| = \|\mathfrak{D}\|$.

8.2.3. *The singleton bihaviour* The next example shows that the principal behaviour $\mathfrak{D}^{\bot\bot}$ can be equipped with a PER such that $\mathfrak{D}$ is the only total design equivalent to itself. The property plays a crucial role in the Uniformity Lemma (Theorem 30, p. 387).

Proposition 19. Let $\mathfrak{D}$ be a design of base $\vdash \langle\rangle$. Then one can form a bihaviour $(\mathfrak{D}^{\bot\bot}, \cong)$ such that:

1 If $\mathfrak{E} \in \mathfrak{D}^{\bot\bot}$ is such that $\mathfrak{E} \cong \mathfrak{E}$, then $|\mathfrak{E}| = \mathfrak{D}$.
2 $\mathfrak{D}$ is bimaterial in $(\mathfrak{D}^{\bot\bot}, \cong)$.

Proof. Define $\cong$ as the biorthogonal of the relation: $\mathfrak{E} \simeq \mathfrak{E}'$ iff $\mathfrak{E} = \mathfrak{E}' \subset \mathfrak{D}$. Observe that, since $\simeq$ is coarser than $\equiv_{\mathfrak{D}^{\bot\bot}}$, we have $\simeq^{\bot}$ is finer than $\equiv_{\mathfrak{D}^{\bot}}$, and, in particular, $\mathfrak{F} \simeq^{\bot} \mathfrak{F}$ for all $\mathfrak{F} \in \mathfrak{D}^{\bot}$.

1 Assume that $\mathfrak{E} \in \mathfrak{D}^{\bot\bot}$, that is, that $\mathfrak{D} \preceq \mathfrak{E}$, with $|\mathfrak{E}| \neq \mathfrak{D}$. Then there exists a chronicle $\mathfrak{c} * \kappa \in \mathfrak{D}$, ending with a proper positive action κ, such that $\mathfrak{c} * \maltese \in \mathfrak{E}$. Then the anti-design $\mathfrak{F} = \mathfrak{Opp}_c$ (Definition 19, p. 328), is such that $[\![\mathfrak{E}, \mathfrak{F}]\!] = \mathfrak{Dai}$, $[\![\mathfrak{D}, \mathfrak{F}]\!] = \mathfrak{Fid}$. Moreover, $\mathfrak{Opp}_c \subset \mathfrak{Opp}_{c*\kappa} \in \mathfrak{D}^{\bot}$, which shows that $\mathfrak{F} \in \mathfrak{D}^{\bot p}$. Then $\mathfrak{F} \simeq^{\bot} \mathfrak{Sk}$, since $[\![\mathfrak{D}', \mathfrak{F}]\!] = [\![\mathfrak{D}', \mathfrak{Sk}]\!] = \mathfrak{Fid}$ for all $\mathfrak{D}' \subset \mathfrak{D}$. Since $[\![\mathfrak{E}, \mathfrak{F}]\!] = \mathfrak{Dai}$ and $[\![\mathfrak{E}, \mathfrak{Sk}]\!] = \mathfrak{Fid}$, we conclude that $\mathfrak{E} \not\cong \mathfrak{E}$.
2 Assume that $\mathfrak{E} \subsetneq \mathfrak{D}$. Then there is an anti-design $\mathfrak{F} \in \mathfrak{D}^{\bot} - \mathfrak{E}^{\bot}$. We have $\mathfrak{F} \simeq^{\bot} \mathfrak{F}$, but $[\![\mathfrak{D}, \mathfrak{F}]\!] = \mathfrak{Dai}$, $[\![\mathfrak{E}, \mathfrak{F}]\!] = \mathfrak{Fid}$, and hence $\mathfrak{D} \not\cong \mathfrak{E}$. □

https://doi.org/10.1017/S096012950100336X

8.3. *Additives*

In what follows, the base is atomic, $\vdash \langle\rangle$ or $\langle\rangle \vdash$.

8.3.1. *Directory of a bihaviour*

Definition 53. If $(\mathbf{G}, \cong)$ is negative, a directory $\mathscr{N}$ is *saturated* with respect to$(\mathbf{G}, \cong)$ when $\mathfrak{Dir}_{\mathscr{N}}$ is bimaterial in $(\mathbf{G}, \cong)$.

Theorem 23. (Bidirectory) If $\mathbf{G}$ is negative, there exists a PER $\sim_{\mathbf{G}}$[†] on $\P\mathbf{G}$ such that the directories that are saturated with respect to $(\mathbf{G}, \cong)$ are exactly the sets $\mathscr{N}$ that can be written as unions $\bigcup \mathscr{N}_k$ of equivalence classes *modulo* $\sim_{\mathbf{G}}$.

Proof. If $(\mathbf{G}, \cong)$ is negative, say that two ramifications I, I' are *related* when there exist partial designs $\mathfrak{E}, \mathfrak{E}' \in \mathbf{G}^{\perp p}$ whose first actions are $(\langle\rangle, I), (\langle\rangle, I')$, respectively, and such that $\mathfrak{E} \cong^{\perp} \mathfrak{E}'$. Define the PER $\sim_{\mathbf{G}}$ on $\P\mathbf{G}$ as the transitive closure of the relation 'to be related'. The remainder of the proof rests heavily upon the remark that, when the positive $\mathfrak{E}$ starts with $(\langle\rangle, I)$, we have $[\![\mathfrak{Dir}_{\mathscr{N}}, \mathfrak{E}]\!]$ converges iff $I \in \mathscr{N}$.

— If $\mathfrak{Dir}_{\mathscr{N}} \not\cong \mathfrak{Dir}_{\mathscr{N}}$, then there are equivalent $\mathfrak{E}, \mathfrak{E}' \in \mathbf{G}^{\perp p}$ such that $\ll \mathfrak{Dir}_{\mathscr{N}} \mid \mathfrak{E} \gg \neq \ll \mathfrak{Dir}_{\mathscr{N}} \mid \mathfrak{E}' \gg$, and it is immediate that $\mathfrak{E}, \mathfrak{E}'$ are distinct from $\mathfrak{Dai}, \mathfrak{Fid}$. If $(\langle\rangle, I)$ and $(\langle\rangle, I')$ are the first actions of $\mathfrak{E}$ and $\mathfrak{E}'$, respectively, then $I \in \mathscr{N}$ iff $I' \notin \mathscr{N}$. If $\mathfrak{Dir}_{\mathscr{N}} \cong \mathfrak{Dir}_{\mathscr{N}}$, but $\mathfrak{Dir}_{\mathscr{N}}$ is not bimaterial, then there exist $I \in \mathscr{N}$ such that $\mathfrak{Dir}_{\mathscr{N}} \cong \mathfrak{Dir}_{\mathscr{N}-\{I\}}$. Then no $\mathfrak{E} \in \mathbf{G}^{\perp p}$ starting with $(\langle\rangle, I)$ can be related to itself.

— If $\mathfrak{Dir}_{\mathscr{N}}$ is bimaterial, then it is easily shown to be closed under the relation 'to be related', and, moreover, if $I \in \mathscr{N}$ is related to nothing, then $\mathfrak{Dir}_{\mathscr{N}} \cong \mathfrak{Dir}_{\mathscr{N}-\{I\}}$. □

Corollary 19.1. Assume that $\mathbf{G}$ is negative. Then the support $\mathscr{G}$ of $\sim_{\mathbf{G}}$ is characterised by the equation

$$\|\mathfrak{Dai}^{-}\| = \mathfrak{Dir}_{\mathscr{G}} \tag{126}$$

8.3.2. *Locative additives*

As in Chapter 5, p. 339, we can define general locative additives. The novelty is the definition of the relation $\cong$.

— $(\bigcap_k \mathbf{G}_k, \cong_{\bigcap_k \mathbf{G}_k})$ is defined by $\mathfrak{D} \cong_{\bigcap_k \mathbf{G}_k} \mathfrak{E} \Leftrightarrow \forall k\ \mathfrak{D} \cong_{\mathbf{G}_k} \mathfrak{E}$.

— The ethics $(\bigcup_k \mathbf{G}_k, \cong)$ is defined by $\cong$ is the transitive closure of the union of the $\cong_{\mathbf{G}_k}$, and $(\bigcup\!\!\!\!\!\!*_k \mathbf{G}_k, \cong_{\bigcup\!\!\!\!\!\!*_k \mathbf{G}_k})$ is its biorthogonal.

The exact analogues of Propositions 5, p. 340, and 6, p. 341, hold for our connectives.

Only part of the results of Subsection 5.1.5, p. 341, remain: $\sim_{\Cap_k \mathbf{G}_k}$ is the transitive closure of the union $\bigcup_k \sim_{\mathbf{G}_k}$ (the same for $\sim_{\biguplus_k \mathbf{G}_k}$). But there is no way to compute $\sim_{\Cap_k \mathbf{G}_k}$ from the $\sim_{\mathbf{G}_k}$: typically, I may be related to itself with respect to the positive $(\mathbf{G}, \cong)$ because of some designs $\mathfrak{D} \cong \mathfrak{D}$, and related to itself with respect to $(\mathbf{G}, \cong')$ because of some designs $\mathfrak{D}' \cong' \mathfrak{D}'$, but the intersection of the set of the $\mathfrak{D}$ and the set of the $\mathfrak{D}'$ may be empty.

[†] The correct notation would be $\sim_{(\mathbf{G},\cong)}$.

8.3.3. *The additive decomposition* As in Chapter 5, p. 339, the symbols $\oplus, \&$ are restricted to disjoint directories. A – say positive – bihaviour $\mathbf{G}$ is *connected* when it is distinct from $\mathbf{0}$ and cannot be written as a non-trivial $\oplus$ of two bihaviours. Indeed, connected bihaviours fall into two cases:

— Those bihaviours $(\mathbf{G}, \cong)$ such that $\P\mathbf{G} \neq \emptyset$ and $I \sim_{\mathbf{G}} J$ for all $I, J \in \P\mathbf{G}$.

— Those bihaviours $(\mathbf{G}, \cong)$ such that $\P\mathbf{G}$ is a singleton $\{I\}$ and $I \not\sim_{\mathbf{G}} I$.

Then any positive bihaviour can be written in a unique way as a $\oplus$ of connected bihaviours.

Example 26. Let us give some example of directories with their PER:

1 The basic case of a behaviour – seen as a bihaviour – corresponds to equality: $I \sim_{\mathbf{G}} J$ iff $I = J \in \P\mathbf{G}$. The additive decomposition of a behaviour as a bihaviour coincides with the familiar one.

2 If the negative $\mathbf{G}$ is such that $\mathfrak{D} \cong_{\mathbf{G}} \mathfrak{D}'$ for all $\mathfrak{D}, \mathfrak{D}' \in \mathbf{G}$, then $\|\mathfrak{Dai}^-\| = \mathfrak{Sk}$ and the PER $\sim_{\mathbf{G}}$ has empty support.

3 In Subsection 8.3.4 below, assume that $\mathbf{G}$ is a plain behaviour. Then $\sim_{\mathbf{H}}$ is the total equivalence relation on $\varphi(\P\mathbf{G}) \cup \psi(\P\mathbf{G})$ whose classes are the sets $\{\varphi(I), \psi(I)\}$.

4 In Subsection 8.4.4, p. 369, assume that $\mathbf{G}$ is a plain behaviour. Then $\sim_{\mathbf{H}}$ is the partial equivalence relation on $\varphi(\P\mathbf{G}) \mathbin{\cup\!\!\!\!\times} \psi(\P\mathbf{G})$ whose classes are the singletons $\{\varphi(I) \cup \psi(I)\}$.

8.3.4. *An example: Symmetric sum* Our basic example should be the basic instance of spiritual quantification. Let $(\mathbf{G}, \cong_{\mathbf{G}})$ be a positive bihaviour, then define $(\mathbf{H}, \cong_{\mathbf{H}})$ by $\mathbf{H} = \varphi(\mathbf{G}) \cup \psi(\mathbf{G})$, the PER being defined by:

$$\varphi(\mathfrak{D}) \cong_{\mathbf{H}} \varphi(\mathfrak{E}) \Leftrightarrow \mathfrak{D} \cong_{\mathbf{G}} \mathfrak{E} \tag{127}$$

$$\psi(\mathfrak{D}) \cong_{\mathbf{H}} \psi(\mathfrak{E}) \Leftrightarrow \mathfrak{D} \cong_{\mathbf{G}} \mathfrak{E} \tag{128}$$

$$\varphi(\mathfrak{D}) \cong_{\mathbf{H}} \psi(\mathfrak{E}) \Leftrightarrow \mathfrak{D} \cong_{\mathbf{G}} \mathfrak{E} \tag{129}$$

$$\psi(\mathfrak{D}) \cong_{\mathbf{H}} \varphi(\mathfrak{E}) \Leftrightarrow \mathfrak{D} \cong_{\mathbf{G}} \mathfrak{E} \tag{130}$$

The material designs $\mathfrak{A} \in (\mathbf{H}, \cong_{\mathbf{H}})$ such that $\mathfrak{A} \cong_{\mathbf{H}} \mathfrak{A}$ are exactly the designs of the form $\varphi(\mathfrak{D})$ or $\psi(\mathfrak{D})$, with $\mathfrak{D} \cong_{\mathbf{G}} \mathfrak{D}$.

The orthogonal $(\mathbf{H}^{\perp}, \cong_{\mathbf{H}^{\perp}})$ can be directly defined by:

— $\mathbf{H}^{\perp} = \varphi(\mathbf{G}^{\perp}) \cap \psi(\mathbf{G}^{\perp})$ – by the way, remember that $\varphi(\mathbf{G}^{\perp})$ is in fact the double orthogonal of the direct image of $\mathbf{G}^{\perp}$ under φ.

— It is enough to define the relation $\cong_{\mathbf{H}^{\perp}}$ between the material designs of $\mathbf{H}^{\perp}$: they are of the form $\varphi(\mathfrak{D}) \cup \psi(\mathfrak{E})$ (mystery of incarnation, Theorem 10, p. 343):

$$\varphi(\mathfrak{D}) \cup \psi(\mathfrak{E}) \cong_{\mathbf{H}^{\perp}} \varphi(\mathfrak{D}') \cup \psi(\mathfrak{E}') \Leftrightarrow \mathfrak{D} \cong_{\mathbf{G}^{\perp}} \mathfrak{D}' \cong_{\mathbf{G}^{\perp}} \mathfrak{E} \cong_{\mathbf{G}^{\perp}} \mathfrak{E}' \tag{131}$$

The material designs $\mathfrak{A} \in (\mathbf{H}^{\perp}, \cong_{\mathbf{H}^{\perp}})$ such that $\mathfrak{A} \cong_{\mathbf{H}^{\perp}} \mathfrak{A}$ are exactly the designs of the form $\varphi(\mathfrak{D}) \cup \psi(\mathfrak{D})$, with $\mathfrak{D} \cong_{\mathbf{G}^{\perp}} \mathfrak{D}$.

8.4. *Multiplicatives*

In what follows, the base is atomic, $\vdash \langle\rangle$ or $\langle\rangle \vdash$.

8.4.1. *Locative multiplicatives* Multiplicatives are better handled in the negative case. For any of our connectives $\bowtie, \ltimes, \rtimes, \infty$, one must define a PER. We shall concentrate on $\bowtie$ and define the bihaviour $(\mathbf{G} \bowtie \mathbf{H}, \simeq_{\mathbf{G}\bowtie\mathbf{H}})$.

Definition 54. If $(\mathbf{G}, \simeq_{\mathbf{G}}), (\mathbf{H}, \simeq_{\mathbf{H}})$ are negative, we define $\mathfrak{F} \simeq_{\mathbf{G}\bowtie\mathbf{H}} \mathfrak{G}$ by

$$\forall \mathfrak{A}, \mathfrak{B}\ (\mathfrak{A} \simeq_{\mathbf{G}^\perp} \mathfrak{B} \Rightarrow (\mathfrak{F})\mathfrak{A} \simeq_{\mathbf{H}} (\mathfrak{G})\mathfrak{B}\ ) \tag{132}$$

Using $((\mathfrak{F})\mathfrak{A})\mathfrak{A}' = ((\mathfrak{F})\mathfrak{A}')\mathfrak{A}\ldots$ (Equation (69), p.350) the right-hand side of the implication can be rewritten:

$$\forall \mathfrak{A}', \mathfrak{B}'\ (\mathfrak{A}' \simeq_{\mathbf{H}^\perp} \mathfrak{B}' \Rightarrow (\mathfrak{F})\mathfrak{A}' \simeq_{\mathbf{G}} (\mathfrak{G})\mathfrak{B}'\ ) \tag{133}$$

From this it is easy to prove the analogue of Theorem 16, p. 352. The same holds for the other multiplicatives.

8.4.2. *Completeness properties* Completeness can only be stated under the hypothesis of *alienation*, that is, $\S\mathbf{G} \cap \S\mathbf{H} = \emptyset$. In this case we introduce the symbols $\otimes, \parr$.

Proposition 20. If $\mathbf{G}, \mathbf{H}$ are alien and positive, then

$$\mathfrak{D} \otimes \mathfrak{D}' \simeq_{\mathbf{G}\otimes\mathbf{H}} \mathfrak{E} \otimes \mathfrak{E}' \Leftrightarrow \mathfrak{D} \simeq_{\mathbf{G}} \mathfrak{D}' \text{ and } \mathfrak{E} \simeq_{\mathbf{H}} \mathfrak{E}' \tag{134}$$

Proof. The implication $\mathfrak{D} \simeq_{\mathbf{G}} \mathfrak{D}'$ and $\mathfrak{E} \simeq_{\mathbf{H}} \mathfrak{E}' \Rightarrow \mathfrak{D} \otimes \mathfrak{D}' \simeq_{\mathbf{G}\otimes\mathbf{H}} \mathfrak{E} \otimes \mathfrak{E}'$ is immediate. The converse can be proved by the 'weakening' technique used in Theorem 19, p. 354. □

8.4.3. *Multiplicatives and directory* We shall compute the directory of the tensor product of alien behaviours.

Proposition 21. If $\mathbf{G}, \mathbf{H}$ are positive and alien, and if $I, I' \in \P\mathbf{G}, I', J' \in \P\mathbf{H}$, then

$$I \cup J \sim_{\mathbf{G}\otimes\mathbf{H}} I' \cup J' \quad \Leftrightarrow \quad I \sim_{\mathbf{G}} I' \text{ and } J \sim_{\mathbf{H}} J' \tag{135}$$

Proof. The statement is an easy consequence of Proposition 20. □

8.4.4. *An example: Symmetric tensor product* We present here a small, but important part of the exponential[†], that is, the *symmetric* tensor product. Let $(\mathbf{G}, \simeq_{\mathbf{G}})$ be a positive bihaviour. Then define $(\mathbf{H}, \simeq_{\mathbf{H}})$ by $\mathbf{H} = \varphi(\mathbf{G}) \otimes \psi(\mathbf{G})$ and

$$\varphi(\mathfrak{D}) \otimes \psi(\mathfrak{E}) \simeq_{\mathbf{H}} \varphi(\mathfrak{D}') \otimes \psi(\mathfrak{E}') \Leftrightarrow \mathfrak{D} \simeq_{\mathbf{G}} \mathfrak{D}' \simeq_{\mathbf{G}} \mathfrak{E} \simeq_{\mathbf{G}} \mathfrak{E}' \tag{136}$$

Although it is only a toy example, the next theorem is important: the same machinery could work *mutatis mutandis* in the case of exponentials.

Lemma 24.1. Let θ be the delocation

$$\theta(3i * \sigma) = (3i+1) * \sigma \quad ; \quad \theta((3i+1) * \sigma) = 3i * \sigma \tag{137}$$

[†] At least this is the case for our present approach to exponentials.

https://doi.org/10.1017/S096012950100336X

Then:

1 $\mathfrak{A} \simeq_{\mathbf{H}} \mathfrak{A}$ and $\mathfrak{F} \simeq_{\mathbf{H}^\perp} \mathfrak{F} \;\Rightarrow\; \ll \theta(\mathfrak{A}) \mid \mathfrak{F} \gg \,=\, \ll \mathfrak{A} \mid \mathfrak{F} \gg$.
2 $\mathfrak{F} \simeq_{\mathbf{H}^\perp} \mathfrak{G} \;\Rightarrow\; \theta(\mathfrak{F}) \simeq_{\mathbf{H}^\perp} \mathfrak{G}$.
3 $\mathfrak{A} \simeq_{\mathbf{H}^{\perp\perp}} \mathfrak{A}$ and $\mathfrak{F} \simeq_{\mathbf{H}^\perp} \mathfrak{F} \;\Rightarrow\; \ll \theta(\mathfrak{A}) \mid \mathfrak{F} \gg \,=\, \ll \mathfrak{A} \mid \mathfrak{F} \gg$.
4 $\mathfrak{A} \simeq_{\mathbf{H}^{\perp\perp}} \mathfrak{B} \;\Rightarrow\; \theta(\mathfrak{A}) \simeq_{\mathbf{H}^{\perp\perp}} \mathfrak{B}$.

Proof. The proof is an easy sequence of verifications. □

Theorem 24. (Uniformity) $(\mathbf{H}, \simeq_{\mathbf{H}})$ is a bihaviour.

Proof. Since $\simeq_{\varphi(\mathbf{G})\otimes\psi(\mathbf{G})}$ is finer than $\simeq_{\mathbf{H}}$, it is still finer than $\simeq_{\mathbf{H}^{\perp\perp}}$. If

$$\varphi(\mathfrak{D}) \otimes \psi(\mathfrak{E}) \simeq_{\mathbf{H}^{\perp\perp}} \varphi(\mathfrak{D}') \otimes \psi(\mathfrak{E}')$$

then $\mathfrak{D} \simeq_{\mathbf{G}} \mathfrak{D}'$, $\mathfrak{E} \simeq_{\mathbf{G}} \mathfrak{E}'$. By the lemma, we also get $\varphi(\mathfrak{E}) \otimes \psi(\mathfrak{D}) \simeq_{\mathbf{H}^{\perp\perp}} \varphi(\mathfrak{D}') \otimes \psi(\mathfrak{E}')$, from which we conclude that $\mathfrak{E} \simeq_{\mathbf{G}} \mathfrak{D}'$. Hence the four designs $\mathfrak{D}, \mathfrak{D}', \mathfrak{E}, \mathfrak{E}'$ are equivalent with respect to $\simeq_{\mathbf{G}}$. The two PERs $\simeq_{\mathbf{H}^{\perp\perp}}$ and $\simeq_{\mathbf{H}}$ are therefore equal. □

The material designs in $(\mathbf{H}, \simeq_{\mathbf{H}})$ such that $\mathfrak{A} \simeq_{\mathbf{H}} \mathfrak{A}$ are exactly the designs $\varphi(\mathfrak{D}) \otimes \psi(\mathfrak{D})$, with $\mathfrak{D} \simeq_{\mathbf{G}} \mathfrak{D}$.

8.5. *Quantifiers*

The prenex forms of Section 7.2, p. 359, still hold (I have not checked the details).

8.6. *Sequents of bihaviours*

Definition 40, p. 356, is easily extended to bihaviours: if $\Xi \vdash \Lambda$ is a pitchfork and $\boldsymbol{\Xi}, \boldsymbol{\Lambda}$ are positive bihaviours $(\mathbf{G}_\sigma, \simeq_\sigma)$ of the respective bases $\vdash \sigma$ for $\sigma \in \Xi, \Lambda$, then one must take the orthogonal of the relation $(\mathfrak{E}_\sigma) \simeq (\mathfrak{E}'_\sigma)$ defined by $\mathfrak{E}_\sigma \simeq_\sigma \mathfrak{E}'_\sigma$ for all $\sigma \in \Xi$, $\mathfrak{E}_\sigma \simeq^\perp_\sigma \mathfrak{E}'_\sigma$ for $\sigma \in \Lambda$:

$$\mathfrak{D} \simeq \mathfrak{D}' \;\Leftrightarrow\; \forall(\mathfrak{E}_\sigma)\forall(\mathfrak{E}'_\sigma)\,(\,(\mathfrak{E}_\sigma) \simeq (\mathfrak{E}'_\sigma) \;\Rightarrow\; [\![\mathfrak{D}, (\mathfrak{E}_\sigma)]\!] = [\![\mathfrak{D}', (\mathfrak{E}'_\sigma)]\!]\,) \tag{138}$$

It is possible to prove the exact analogues of the results of Theorem 21, p. 356, as in the following theorem.

Theorem 25.

— The sequent of bihaviours $\vdash$ is reduced to $(\mathbf{0}, =)$.
— The sequent of bihaviours $\vdash (\mathbf{G}, \simeq)$ is equal to $(\mathbf{G}, \simeq)$.
— The sequent of bihaviours $(\mathbf{G}, \simeq) \vdash$ is equal to $(\mathbf{G}^\perp, \simeq^\perp)$.
— When $\boldsymbol{\Xi} = \mathbf{G}$, Equation (138) can be replaced with:

$$\mathfrak{D} \simeq_{\mathbf{G}\vdash\boldsymbol{\Lambda}} \mathfrak{D}' \;\Leftrightarrow\; \forall\mathfrak{E}\forall\mathfrak{E}'\,(\,\mathfrak{E} \simeq_{\mathbf{G}} \mathfrak{E}' \;\Rightarrow\; [\![\mathfrak{D}, \mathfrak{E}]\!] \simeq_{\vdash\boldsymbol{\Lambda}} [\![\mathfrak{D}', \mathfrak{E}']\!]\,) \tag{139}$$

— When $\mathbf{G} \in \boldsymbol{\Lambda}$, Equation (138) can be replaced with:

$$\mathfrak{D} \simeq_{\boldsymbol{\Xi}\vdash\mathbf{G},\boldsymbol{\Lambda}'} \mathfrak{D}' \;\Leftrightarrow\; \forall\mathfrak{E}\forall\mathfrak{E}'\,(\,\mathfrak{E} \simeq_{\mathbf{G}^\perp} \mathfrak{E}' \;\Rightarrow\; [\![\mathfrak{D}, \mathfrak{E}]\!] \simeq_{\boldsymbol{\Xi}\vdash\boldsymbol{\Lambda}'} [\![\mathfrak{D}', \mathfrak{E}']\!]\,) \tag{140}$$

It is useful to find the analogue of the PER $\sim$ in such a case. For this we need to define $\P\Xi \vdash \mathbf{\Lambda}$, and $\sim_{\Xi\vdash\mathbf{\Lambda}}$. There are two cases:

Negative base: If $\xi \in \Xi$, then $\P\Xi \vdash \mathbf{\Lambda} = \P\mathbf{G}_\xi$ and $\sim_{\Xi\vdash\mathbf{\Lambda}} = \sim_{\mathbf{G}_\xi}$.

Positive base: If $\Xi = \emptyset$, then $\P\Xi \vdash \mathbf{\Lambda}$ is the disjoint union of the $\P\mathbf{G}_\sigma$ ($\sigma \in \Lambda$). Moreover, $(\sigma, I) \sim_{\Xi\vdash\mathbf{\Lambda}} (\tau, J)$ iff $\sigma = \tau$ and $I \sim_{\mathbf{G}_\sigma} I$.

The negative case comes from a straightforward generalisation of Proposition 19.1, p. 367. The positive case is a consequence of the important property given in the following proposition.

Proposition 22. Assume that $\mathfrak{D} \cong_{\vdash\mathbf{\Lambda}} \mathfrak{E}$ and $\mathfrak{D}, \mathfrak{E}$ have first actions $(\sigma, I), (\tau, J)$, respectively; then $\sigma = \tau$ and $I \sim_{\mathbf{G}_\sigma} J$.

Proof. Consider, for instance, $\vdash \mathbf{G}, \mathbf{H}$. Then $\mathfrak{D} \cong_{\vdash\mathbf{\Lambda}} \mathfrak{E}$ implies that

$$[\![\mathfrak{D}, \mathfrak{Dir}_{\mathscr{M}}, \mathfrak{Dir}_{\mathscr{N}}]\!] = [\![\mathfrak{E}, \mathfrak{Dir}_{\mathscr{M}}, \mathfrak{Dir}_{\mathscr{N}}]\!]$$

for all *saturated* $\mathfrak{Dir}_{\mathscr{M}}$ (with respect to $\mathbf{G}^\perp$) and $\mathfrak{Dir}_{\mathscr{N}}$ (with respect to $\mathbf{H}^\perp$). The PER $\sim_{\vdash\mathbf{G},\mathbf{H}}$ on the disjoint union $\P\mathbf{G} + \P\mathbf{H}$ is therefore equal to the disjoint union $\sim_{\mathbf{G}} + \sim_{\mathbf{H}}$. □

8.7. *Bihaviours as games*

We know, see Subsection 4.2.3, p. 335, that a behaviour induces a game: the design $\mathfrak{D}$ (or rather its incarnation) is a *strategy* when all disputes (plays) generated from counter-strategies $\mathfrak{E} \in \mathbf{G}^\perp$ converge. But what about bihaviours? It is still possible to view them as sorts of games: instead of playing a single design $\mathfrak{D}$, our would-be strategy will consist of *four* designs on the pro-base, $(\mathfrak{D}_1 \subset \mathfrak{D}'_1, \mathfrak{D}_2 \subset \mathfrak{D}'_2)$, what we call a 4-*design*. This construction is only of conceptual interest and will legitimate Definition 56, p. 373 of uniformity, whose effect is precisely to reduce a 4-design to a familiar design.

We first define orthogonality of 4-designs.

Definition 55. The 4-designs $(\mathfrak{D}_1 \subset \mathfrak{D}'_1, \mathfrak{D}_2 \subset \mathfrak{D}'_2)$ and $(\mathfrak{E}_1 \subset \mathfrak{E}'_1, \mathfrak{E}_2 \subset \mathfrak{E}'_2)$ are *orthogonal* when:

$$\mathfrak{D}'_1 \perp \mathfrak{E}'_1 \quad ; \quad \mathfrak{D}'_2 \perp \mathfrak{E}'_2 \tag{141}$$

and

$$\ll \mathfrak{D}_1 \mid \mathfrak{E}_1 \gg \; = \; \ll \mathfrak{D}_2 \mid \mathfrak{E}_2 \gg \tag{142}$$

Remark 16. Is this definition *streamlike*? Equalities of the form '$\ll \mathfrak{D} \mid \mathfrak{E} \gg = \ldots$' are definitely not streamlike. But imagine that $\mathfrak{D}$ is given to us as an intersection $\mathfrak{D}' \cap \mathfrak{X}$, with $\mathfrak{D}' \in \mathbf{G}$ and $\mathfrak{X}$ is a set of chronicles closed under restriction[†]. If $\mathfrak{E} = \mathfrak{E}' \cap \mathfrak{Y}$, with $\mathfrak{E}' \in \mathbf{G}^\perp \ldots$, then, since we know that $\mathfrak{D}' \perp \mathfrak{E}'$, the value of $\ll \mathfrak{D} \mid \mathfrak{E} \gg$ can actually be

[†] And such that both $\mathfrak{X}$ and its complement are streamlike.

computed: first wait until $\{\mathfrak{D}', \mathfrak{E}'\}$ converges, then check whether or not the normalisation takes place inside the space delimited by $\mathfrak{X}, \mathfrak{Y}$[†].

To each bihaviour $(\mathbf{G}, \cong)$, one can now associate the set $\widehat{(\mathbf{G}, \cong)}$ of all the 4-designs $(\mathfrak{D}_1 \subset \mathfrak{D}'_1, \mathfrak{D}_2 \subset \mathfrak{D}'_2)$ such that

1 $\mathfrak{D}'_1, \mathfrak{D}'_2 \in \mathbf{G}$.

2 $\mathfrak{D}_1 \cong \mathfrak{D}_2$.

Proposition 23.

1 $\mathbf{G}$ can be recovered from $\widehat{(\mathbf{G}, \cong)}$.

2 $\widehat{(\mathbf{G}^{\perp}, \cong^{\perp})} = \widehat{(\mathbf{G}, \cong)}^{\perp}$.

Proof.

1 Observe that $\emptyset \cong \emptyset$ (remember that $\emptyset$ is written $\mathfrak{Fid}$ or $\mathfrak{Sk}$, depending on the polarity) so that:

$$\mathfrak{D} \in \mathbf{G} \Leftrightarrow (\emptyset \subset \mathfrak{D}, \emptyset \subset \mathfrak{D}) \in \widehat{(\mathbf{G}, \cong)} \tag{143}$$

2 This is almost immediate. □

In Section 9.1.1 we shall be concerned with *winning* conditions. One of these conditions is uniformity: from the viewpoint of `Proponent`, whose strategy is $(\mathfrak{D}_1 \subset \mathfrak{D}'_1, \mathfrak{D}_2 \subset \mathfrak{D}'_2)$, the four disputes $[\mathfrak{D}_1 \rightleftharpoons \mathfrak{E}_1], [\mathfrak{D}_2 \rightleftharpoons \mathfrak{E}_2], [\mathfrak{D}'_1 \rightleftharpoons \mathfrak{E}'_1]$, and $[\mathfrak{D}'_2 \rightleftharpoons \mathfrak{E}'2]$ should be the same, which means that the blame can be put on `Opponent` when they differ, for example, when $[\![\mathfrak{D}_1, \mathfrak{E}_1]\!]$ diverges: the first time any two of these disputes differ is due to a different action (or absence of action) of `Opponent`. If we consider counter-4-designs of the form $(\emptyset \subset \mathfrak{E}', \emptyset \subset \mathfrak{E}')$, this forces – using Exercise 5, p. 335 – $|\mathfrak{D}'_1| = |\mathfrak{D}'_2|$. Similarly, using counter-4-designs of the form $(\mathfrak{E} \subset \mathfrak{E}', \mathfrak{E} \subset \mathfrak{E}')$, we get $\|\mathfrak{D}_1\| = \|\mathfrak{D}_2\|$. Without loss of generality, we can assume $\mathfrak{D}'_1, \mathfrak{D}'_2$ material and $\mathfrak{D}_1, \mathfrak{D}_2$ bimaterial, and we get $\mathfrak{D}_1 = \mathfrak{D}_2 = \mathfrak{D}$, $\mathfrak{D}'_1 = \mathfrak{D}'_2 = \mathfrak{D}'$.

Now take a general counter-4-design $(\mathfrak{E}_1 \subset \mathfrak{E}'_1, \mathfrak{E}_2 \subset \mathfrak{E}'_2)$, and assume that $[\![\mathfrak{D}', \mathfrak{E}_1]\!]$ converges. Then $[\![\mathfrak{D}', \mathfrak{E}'_1]\!]$ converges *with the same dispute*, and the dispute $[\mathfrak{D} \rightleftharpoons \mathfrak{E}_1]$ must correspond to a convergent computation, since, when $[\mathfrak{D} \rightleftharpoons \mathfrak{E}_1] \neq [\mathfrak{D}' \rightleftharpoons \mathfrak{E}'_1] = [\mathfrak{D}' \rightleftharpoons \mathfrak{E}_1]$, the blame cannot be put on `Opponent`. Then $[\![\mathfrak{D}, \mathfrak{E}_2]\!]$ converges and $[\![\mathfrak{D}', \mathfrak{E}_2]\!]$ does as well. This shows that $\mathfrak{D}' \cong \mathfrak{D}'$ and that $\mathfrak{D} = \|\mathfrak{D}'\|$.

We have therefore shown that, up to incarnation and biincarnation, a uniform 4-design can be written $(\|\mathfrak{D}\| \subset \mathfrak{D}, \|\mathfrak{D}\| \subset \mathfrak{D})$ for some incarnated $\mathfrak{D}$ such that $\mathfrak{D} \cong \mathfrak{D}$: eventually a *uniform* 4-design reduces to a single design.

9. Truth and completeness

The traditional meaning of completeness relates truth and provability: this is *external completeness*. Our task is therefore to define truth and then to prove its adequacy with

[†] Indeed the 'dispute' generated by $\{\mathfrak{D}, \mathfrak{E}\}$ is an initial segment of the dispute generated by $\{\mathfrak{D}', \mathfrak{E}'\}$; if this segment is proper, the normalisation diverges.

respect to a formal system. In the following chapters we shall establish full soundness and completeness of $\mathbf{MALL}_2$, that is, second-order multiplicative-additive linear logic[†] with respect to ludics. In this chapter we content ourselves with the definition and basic properties of truth.

9.1. *To lose; to win*

9.1.1. *The losers* In this subsection we shall devise *necessary* conditions for a 4-design $(\mathfrak{D}_1 \subset \mathfrak{D}'_1, \mathfrak{D}_2 \subset \mathfrak{D}'_2)$ to be winning against a 4-anti-design $(\mathfrak{E}_1 \subset \mathfrak{E}'_1, \mathfrak{E}_2 \subset \mathfrak{E}'_2)$. For this we shall fantasise some ideal properties of disputes; these conditions will usually be violated, and the trespasser (in fact the initial trespasser) will *lose*.

Uniformity: In a perfect world, the 4 cut-nets $\{\mathfrak{D}_i, \mathfrak{E}_i\}, \{\mathfrak{D}'_i, \mathfrak{E}'_i\} (i = 1, 2)$ should converge, *with the same dispute*. Of course, this can hardly be the case, but any time two among these 4 disputes differ, one of the players is to blame for the mismatch: if the disputes differ because of `Proponent` (the owner of $(\mathfrak{D}_1 \subset \mathfrak{D}'_1, \mathfrak{D}_2 \subset \mathfrak{D}'_2)$), then `Proponent` loses. If `Proponent` never loses for this reason, we have just established (Subsection 8.7) that:

$$|\mathfrak{D}'_1| = |\mathfrak{D}'_2|, \|\mathfrak{D}_1\| = \|\mathfrak{D}_2\| = \|\mathfrak{D}'_1\| = \|\mathfrak{D}'_2\| \tag{144}$$

that is, up to (bi-)incarnation, one can assume that the 4-design is of the form $(\mathfrak{D} \subset \mathfrak{D}, \mathfrak{D} \subset \mathfrak{D})$. This is our first condition, *uniformity*: the four components should be equal. Observe that if we form a net of uniform 4-designs, the normal form, provided it exists, is uniform too.

But *basta* with 4-designs! If I want to win (and not only to prevent `Opponent` from winning), I must play 'uniformly', that is, I must select the same design $\mathfrak{D}$ four times, and, of course, $\mathfrak{D}$ must satisfy a requirement:

$$(\mathfrak{D} \subset \mathfrak{D}, \mathfrak{D} \subset \mathfrak{D}) \in \widehat{(\mathbf{G}, \cong)} \tag{145}$$

which reduces to the following definition.

Definition 56. (Uniformity) $\mathfrak{D} \in \mathbf{G}$ is *uniform* when $\mathfrak{D} \cong \mathfrak{D}$.

This is the first necessary condition to be winning. Like all winning properties, it is a property of single designs. From now on we can forget 4-designs...

Obstination: In a perfect world, all disputes should be infinite... but precisely, this is never the case. One of the two players is to blame for that, that is, the first one who fails to deliver a positive action (when normalisation diverges) or the first one who aborts by using $\maltese$ loses. If I want to win, I play *obstinately*, that is, I never cause termination. If $\mathfrak{D}$ is material, note that a daimon in $\mathfrak{D}$ *must* be consumed in the normalisation of some $[\![\mathfrak{D}, \mathfrak{E}]\!]$. This requirement gives us the following definition.

Definition 57. (Obstination) $\mathfrak{D} \in \mathbf{G}$ is *stubborn* when it does not use the daimon $\maltese$.

[†] Without the multiplicative constants $1, \perp$.

Observe that if we form a net of stubborn designs, the normal form (provided it exists) is stubborn too.

Parsimony: In this ideal world, there should be no waste. In particular, all created loci should eventually be used as a focus. Observe that this principle is hardly tenable, unless one uses many actions $(\xi, \emptyset)$ that consume foci without replacing them. Again the problem is not to fulfil the requirement, but *to put the blame on `Opponent`*, the *principle of the barbichette*. In other words, 'OK, I didn't focus on ξ, but I was waiting until `Opponent` focused on ξ'', which he never did', says `Proponent`, but he may only be pretending... This dialogue occurs during a dispute $[\mathfrak{D} \rightleftharpoons \mathfrak{E}] = \mathfrak{S} \cup \mathfrak{T}$. `Proponent`'s claim is that he can extend $\mathfrak{S}$ into $\mathfrak{S}'$ in such a way that an action with focus ξ is performed in $\mathfrak{S}'$; the standard way to do so is to extend a maximal chronicle $\mathfrak{c} \in \mathfrak{S}$ into $\mathfrak{c} * (\xi', I') * (\xi, I) \in \mathfrak{S}'$. But this might be impossible, typically when the pretender has unfortunately destroyed all available loci, by toying too much with the ramification $\emptyset$... The claim of `Proponent` is credible if he is able to dispatch the unfocused loci without loss; such a dispatching is called... a dessin, and 'no loss' means *exact* rules.

Definition 58. (Exactness) With the notation of Subsection 2.2.2, p. 308, a rule is said to be *exact* when:

Demon: No restriction.

Positive rule: $\Lambda = \bigcup_i \Lambda_i$.

Negative rule: $\Lambda_I = \Lambda$ for all $I \in \mathcal{N}$.

A design is *exact* when it admits an *exact* dessin, that is, a design-dessin in which all rules are exact.

The typical example of an inexact design is the following (notice the use of the ramification $\emptyset$):

$$\dfrac{\dfrac{\overline{\vdash \langle 0,0\rangle, \langle 0,1\rangle}\;{\scriptstyle(\langle 0,0\rangle,\emptyset)}}{\langle 0\rangle \vdash}\;{\scriptstyle(\langle 0\rangle,\{0,1\})}}{\vdash \langle\rangle}\;{\scriptstyle(\langle\rangle,\{0\})} \tag{146}$$

If you are caught with this guy, do not say that you planned to focus on $\langle 0, 1\rangle$... To sum up, the last necessary winning condition is parsimony.

Definition 59. (Parsimony) $\mathfrak{D} \in \mathbf{G}$ is *parsimonious* when for all $\mathfrak{E} \in \mathbf{G}^{\perp}$ the slice $\mathfrak{D}_{\mathfrak{E}} \subset \mathfrak{D}$ that has been consumed during the normalisation (and defined by $[\mathfrak{D} \rightleftharpoons \mathfrak{E}] = \mathfrak{D}_{\mathfrak{E}} \cup \mathfrak{E}_{\mathfrak{D}}$) is exact.

The primal meaning of parsimony is the forbidding of the positive action $(\xi, \emptyset)$ on a non-atomic base $\vdash \xi, \Lambda$. But this condition is really warped:

1 There is no simple rephrasing in terms of designs-desseins, not even in terms of designs-dessins: the requirement that certain slices of $\mathfrak{D}$ are exact is not enough to ensure the exactness of the whole $\mathfrak{D}$, see Subsection 11.2.1. For this reason, the Full Completeness Theorem (Theorem 32, p. 392) for $\mathbf{MALL}_2$, is slightly below the methodological

standards of this monograph: it is proved under the assumption of exactness. I do not think that anybody has the right to replace the interactive *parsimony* with the non-interactive *exactness* in the definition 60 of winning, and this poses a small, but fruitful problem.

2 If you understand it interactively, the standard way to be parsimonious is never to use positive actions $(\xi, \emptyset)$. But what hypocrisy! The idea is to make the number of available foci decrease, but if you never use $\emptyset$, you are relying on `Opponent`...

Of all the notions of ludics, parsimony is the one that has endeavoured the most transformations. It is not completely unlikely that this condition will be dropped eventually, which of course would mean a replacement of linear logic by an affine version. This would be a way to escape the problem due to the mismatch between *parsimonious* and *exact*, but there might be more exciting solutions.

9.1.2. *Winning* Uniform, stubborn and parsimonious: the ideal citizen, surely not a *loser*...

Definition 60. (Winning) A design $\mathfrak{D} \in (\mathbf{G}, \cong)$ is *winning* when it is uniform, stubborn and parsimonious. A design is *losing* when it is not winning.

Theorem 26. (Winning) Winning is preserved by normalisation. For instance if $\mathfrak{D}, \mathfrak{E}$ are winning designs in $\vdash (\mathbf{G}, \cong)$ and $(\mathbf{G}, \cong) \vdash (\mathbf{H}, \cong')$, then $[\![\mathfrak{D}, \mathfrak{E}]\!]$ is a winning design in $\vdash (\mathbf{H}, \cong')$.

Proof. The proof is immediate. □

Example 27. If $(\mathbf{G}, \cong)$ is a bihaviour, then the fax $\mathfrak{Fax}_{\xi,\xi'} \in \varphi(\mathbf{G}) \vdash \varphi'(\mathbf{G})$ is a winning design.

Proposition 24. If $\mathfrak{D}$ is winning, it remains winning in any super bihaviour; its incarnation is winning too.

Proof. Inclusion of bihaviours is defined in the obvious way (so that $\mathfrak{D} \cong_{\mathbf{G}} \mathfrak{D} \Rightarrow \mathfrak{D} \cong_{\mathbf{G}'} \mathfrak{D}$). The closure of winning under incarnation is obvious too, for example, we have $\mathfrak{D} \cong \mathfrak{D} \Rightarrow |\mathfrak{D}| \cong |\mathfrak{D}|$, see Proposition 18, p. 366. □

9.1.3. *Truth and falsity*

Definition 61. (Truth) A bihaviour $\mathbf{G}$ is *true* when it contains a winning design, and *false* when $\mathbf{G}^{\perp}$ contains a winning design.

Corollary 26.1. $\mathbf{G}, \mathbf{G}^{\perp}$ cannot both be true – and hence they cannot both be false.

Proof. If $\mathfrak{D}, \mathfrak{E}$ are winning in $\mathbf{G}, \mathbf{G}^{\perp}$, respectively, then the normal form $[\![\mathfrak{D}, \mathfrak{E}]\!] = \mathfrak{Dai}$ is winning, in particular, stubborn... □

Remark 17.

— Most bihaviours will be neither true nor false. Obstination is the only exclusive

https://doi.org/10.1017/S096012950100336X

condition, that is, two orthogonal designs cannot both be stubborn: this condition is responsible for a traditional, but overestimated, aspect of logic, namely *consistency*.

— The incarnation of a winning design remains winning. As a consequence of Proposition 24, p. 375, truth is preserved by super-typing, falsity is preserved by subtyping.

— More generally: if $\mathfrak{D}$ is a morphism (see Definition 41, p. 357) from $\mathbf{G}$ to $\mathbf{H}$ and $\mathbf{G}$ is true, then $\mathbf{H}$ is true; if $\mathbf{H}$ is false, then $\mathbf{G}$ is false. This is why we requested morphisms to be winning.

9.2. *Logical Interference*

9.2.1. *Is classical logic 'stronger'?* There is a spiritual prejudice that says 'you cannot do better than classical logic'. If a formula A is provable, then it remains provable when its connectives are interpreted 'classically', that is, all conjunctions are read as $\wedge$, all disjunctions as $\vee$, and exponentials as the identity. The reason is that linear and intuitionistic connectives are distinguished by a specific maintenance of structural rules[†], whereas classical logic is more liberal. To put it crudely, when you do not know if a thing is true, first try classically and then refine – if it works. This is why illiterate people sometimes say that intuitionistic and linear logic are 'weaker' than classical logic.

However, this was before locative phenomenons were disclosed. To see that something new occurs, observe that $\forall d(A_d \oplus B_d)$ is equal to $(\forall d A_d) \oplus (\forall d B_d)$, contradicting the classical rules for disjunction: additive (and intuitionistic) disjunction enjoy strong non-classical principles. So the question is: we have a notion of truth, which does not follow classical logic; can we find examples in which truth behaves contradictorily to truth tables?

In what follows, we consider the locative conjunction $\odot$, and produce aberrant truth tables. This shows the existence of a *logical interference* due to the sharing of *loci*.

9.2.2. *True* $\odot$ *True* $=$ *False*

— For $i = 0, 1, 2$ define positive designs $\mathfrak{C}_i$ with basis $\vdash \xi$

$$\dfrac{\dfrac{}{\xi i \vdash}\,{\scriptstyle(\xi i,\emptyset)}}{\vdash \xi}\,{\scriptstyle(\xi,\{i\})} \tag{147}$$

$\mathfrak{C}_0$ is *winning*, and hence $\mathbf{C}_0 = \mathfrak{C}_{o}{}^{\perp\perp}$ is *true*, but $\mathbf{C}_0 \odot \mathbf{C}_0 = \mathbf{0}$ is *false*.

9.2.3. *False* $\odot$ *False* $=$ *True* The second example is a variation on the Gustave function.

— For $i = 0, 1, 2$ consider the positive designs $\mathfrak{D}_i$ of basis $\vdash \xi$:

$$\dfrac{\dfrac{\dfrac{}{\vdash \xi ii}\,\maltese}{\xi i \vdash}\,{\scriptstyle(\xi i,\{\{i\}\})}}{\vdash \xi}\,{\scriptstyle(\xi,\{i\})} \tag{148}$$

[†] One of the main discoveries of linear logic was that the usual intuitionistic restriction 'One formula to the right' is in reality the forbidding of contraction to the right.

— Then $\mathbf{D} = \{\mathfrak{D}_0, \mathfrak{D}_1, \mathfrak{D}_2\}^{\perp\perp}$ is *false*, as well as $\mathbf{E} = \{\mathfrak{C}_0 \odot \mathfrak{D}_1, \mathfrak{C}_1 \odot \mathfrak{D}_2, \mathfrak{C}_2 \odot \mathfrak{D}_0\}^{\perp\perp}$: typically, $\mathfrak{C}_0 \odot \mathfrak{D}_1$ equals

$$\dfrac{\dfrac{}{\xi 0 \vdash}\,{\scriptstyle(\xi 0,\emptyset)} \qquad \dfrac{\dfrac{}{\vdash \xi 11}\,\maltese}{\xi 1 \vdash}\,{\scriptstyle(\xi 1,\{\{1\}\})}}{\vdash \xi}\,{\scriptstyle(\xi,\{0,1\})} \tag{149}$$

and the design

$$\dfrac{\dfrac{\dfrac{}{\xi 11 \vdash \xi 0}\,{\scriptstyle(\xi 11,\emptyset)}}{\vdash \xi 0, \xi 1}\,{\scriptstyle(\xi 1,\{1\})} \quad \dfrac{\dfrac{}{\xi 22 \vdash \xi 1}\,{\scriptstyle(\xi 22,\emptyset)}}{\vdash \xi 1, \xi 2}\,{\scriptstyle(\xi 2,\{2\})} \quad \dfrac{\dfrac{}{\xi 00 \vdash \xi 2}\,{\scriptstyle(\xi 00,\emptyset)}}{\vdash \xi 2, \xi 0}\,{\scriptstyle(\xi 0,\{0\})}}{\xi \vdash}\,{\scriptstyle(\xi,I)} \tag{150}$$

with $I = \{\{0,1\},\{1,2\},\{2,0\}\}$, is a *winning* design in $\mathbf{E}^{\perp}$, and hence $\mathbf{E}$ is false.

— $\mathbf{D} \odot \mathbf{E}$ is *true*: this behaviour is the biorthogonal of the three designs

$$\mathfrak{C}_0 \odot \mathfrak{D}_1 \odot \mathfrak{D}_2, \mathfrak{C}_1 \odot \mathfrak{D}_2 \odot \mathfrak{D}_0, \mathfrak{C}_2 \odot \mathfrak{D}_0 \odot \mathfrak{D}_1.$$

Their intersection

$$\dfrac{\dfrac{}{\xi 0 \vdash}\,{\scriptstyle(\xi 0,\emptyset)} \quad \dfrac{}{\xi 1 \vdash}\,{\scriptstyle(\xi 1,\emptyset)} \quad \dfrac{}{\xi 2 \vdash}\,{\scriptstyle(\xi 2,\emptyset)}}{\vdash \xi}\,{\scriptstyle(\xi,\{0,1,2\})} \tag{151}$$

is in $\mathbf{D} \odot \mathbf{E}$, and this design is *winning*, so $\mathbf{D} \odot \mathbf{E}$ is *true*.

10. Soundness of $\mathbf{MALL}_2$

In this chapter we discuss the technical issues of completeness and soundness in their *full* external acceptance. The syntax for $\mathbf{MALL}_2$ is introduced and full soundness is established.

10.1. *Full completeness and soundness*

10.1.1. *The restricted formulation* The schizophrenic tradition of the XX$^{\text{th}}$ century used to present logic through the mutual adequacy of reality (syntax) and thought (semantics) (sorry, reality (semantics) *vs.* thought (syntax)):

Soundness: If the formula A is *provable*, then A is *true* in all models.
Completeness: If the formula A is true in all models, then A is provable.

The formula A is supposed to be first-order. Indeed A is not closed, since it contains predicate or propositional parameters. If we remark that, say, $A[P] \Rightarrow B[P,Q]$ is provable iff $\forall X \forall Y (A[X] \Rightarrow B[X,Y])$ is provable, whereas the same $A[P] \Rightarrow B[P,Q]$ is true in all models iff $\forall X \forall Y (A[X] \Rightarrow B[X,Y])$ is true, we get a more satisfactory formulation:

Soundness: If the closed formula A is *provable*, then A is *true*.
Completeness: If the closed $\mathbf{\Pi}^1$ formula A is true, then A is provable.

Observe that soundness holds for all closed A, whereas completeness must be restricted to $\mathbf{\Pi}^1$ formulas, that is, formulas in which second-order quantifiers are universal. The second-order translation of arithmetic, using the Dedekind definition of natural numbers, shows that $\mathbf{\Pi}^1$ roughly corresponds to what we call $\mathbf{\Sigma}^0_1$, whereas the dual class $\mathbf{\Sigma}^1$ corresponds to $\mathbf{\Pi}^0_1$, a class of formulas known since Gödel's first incompleteness theorem (1931) to contain undecidable sentences: the restriction to $\mathbf{\Pi}^1$ is therefore strictly necessary.

10.1.2. *The full formulation* The Curry–Howard isomorphism, denotational semantics, all the categorical tradition, up to geometry of interaction and ludics, do not interpret formulas, but their proofs. For instance, ludics will associate to any closed formula A a bihaviour $\mathbf{A}$ and to any proof π of A a design $\boldsymbol{\pi} \in \mathbf{A}$. Moreover, certain designs are distinguished – this is the winning conditions of Definition 60, p. 375 – and we obtain the formulation (*full* completeness and soundness):

Soundness: If π is a proof of the closed formula A, then $\boldsymbol{\pi}$ is winning and $\boldsymbol{\pi} \in \mathbf{A}$.

Completeness: If A is a closed $\mathbf{\Pi}^1$ formula, if $\mathfrak{D} \in \mathbf{A}$ is winning and *material*, then $\mathfrak{D} = \boldsymbol{\pi}$ for a certain cut-free proof π of A.

The restriction to cut-free proofs and material designs is necessary: for instance, we have $\mathbf{A} \mathbin{\&} \mathbf{B} \subset \mathbf{A}$, but a proof of $A \mathbin{\&} B$ is not a proof of A. In general $\boldsymbol{\pi}$ is not material, think of a proof of $\exists X X$, which comes from a proof of some positive formula A: the incarnation $|\boldsymbol{\pi}|_{\exists \mathbf{X}\mathbf{X}}$ is equal to $\mathfrak{S}\mathfrak{k}^+$... However, if the formula A is $\mathbf{\Pi}^1$, then $\boldsymbol{\pi}$ is material: this matches the completeness theorem. By the way, there is a puzzling problem related to incarnation: it is immediate that normalisation does not preserve incarnation, that is, a net of incarnated designs may normalise into a non-incarnated design. However, incarnation is preserved in the case of proofs of $\mathbf{\Pi}^1$ formulas... is there a direct explanation of this fact?

If we remember that truth is the existence of a winning design in $\mathbf{A}$, our full formulation admits a forgetful formulation, which is nothing but the restricted formulation of Subsection 10.1.1, p. 377. In particular, the restriction of full completeness to $\mathbf{\Pi}^1$ formulas is strictly necessary... by Gödel's theorem.

10.1.3. *The results* Our aim is to prove full soundness and completeness for linear logic without exponentials. The task will be carried out, with certain limitations.

- We must remove the constant 1: this is because one cannot delocate the empty ramification (there are problems with $\mathbf{1} \oplus \mathbf{1}$, see Subsection 6.2.4, p. 355). For the same reason, our second-order quantification will be restricted to positive bihaviours $(\mathbf{G}, \cong)$ such that $\emptyset \notin \P\mathbf{G}$; if we introduce the notation $\wp_*(\mathbb{N})$ for $\wp_f(\mathbb{N}) - \{\emptyset\}$, variables range over bihaviours $\mathbf{X}$ such that $\P\mathbf{X} \subset \wp_*(\mathbb{N})$.
- We cannot consider first-order quantification, since the existence of prenex forms contradicts completeness: the conflict between completeness and prenex forms cannot be solved in two lines. If one absolutely needs completeness, one should devise another interpretation... which is beyond the limits of this monograph.
- There is a novel connective, namely the shift $\updownarrow$.

https://doi.org/10.1017/S096012950100336X

10.2. *The syntax of* $\boldsymbol{MALL_2}$

10.2.1. *Propositions* The syntax for $\mathbf{MALL}_2$ that we consider is limited to *positive* propositions:

$$P = X, Y \ldots ; 0; \downarrow P^{\perp}; P \oplus P; P \otimes P; \exists X P$$

Negative propositions are implicitly handled through the left-hand side of sequents.

10.2.2. *Sequents*

Definition 62. (Sequents) A *sequent* is an expression $\Gamma \vdash \Delta; \Sigma$ where Γ, Δ, Σ are finite multisets of propositions, with the following *stoup* constraint:

If the stoup Σ *is non-empty, it contains exactly one formula and* Γ *consists only of propositional variables.*

10.2.3. *The calculus* The sequent calculus of $\mathbf{MALL}_2$ is basically a variation on the focusing calculi of Andreoli (Andreoli and Pareschi 1991); the only original feature is the treatment of the propositional variables.

- Cut

$$\frac{\Gamma \vdash \Delta; P \quad \Gamma', P \vdash \Delta'; \Sigma}{\Gamma, \Gamma' \vdash \Delta, \Delta'; \Sigma}\ {;Cut} \qquad\qquad \frac{\Gamma \vdash \Delta, P; \quad \Gamma', P \vdash \Delta';}{\Gamma, \Gamma' \vdash \Delta, \Delta';}\ {Cut;}$$

- Atom (Identity axiom)

$$\frac{}{X \vdash ; X}\ {X \vdash X}$$

- Focalisation

$$\frac{\Gamma \vdash \Delta; P}{\Gamma \vdash \Delta, P;}\ {Foc}$$

- Shift

$$\frac{\Gamma, P \vdash \Delta;}{\Gamma \vdash \Delta; \downarrow P^{\perp}}\ {\vdash\downarrow} \qquad\qquad \frac{\Gamma \vdash \Delta, P;}{\Gamma, \downarrow P^{\perp} \vdash \Delta;}\ {\downarrow\vdash}$$

- Tensor

$$\frac{\Gamma \vdash \Delta; P \quad \Gamma' \vdash \Delta'; Q}{\Gamma, \Gamma' \vdash \Delta, \Delta'; P \otimes Q}\ {\vdash\otimes} \qquad\qquad \frac{\Gamma, P, Q \vdash \Delta;}{\Gamma, P \otimes Q \vdash \Delta;}\ {\otimes\vdash}$$

- Zero

$$\frac{}{\Gamma, 0 \vdash \Delta;}\ {0\vdash}$$

https://doi.org/10.1017/S096012950100336X

- Plus

$$\frac{\Gamma \vdash \Delta; P}{\Gamma \vdash \Delta; P \oplus Q}\vdash l\oplus \qquad\qquad \frac{\Gamma, P \vdash \Delta; \quad \Gamma, Q \vdash \Delta;}{\Gamma, P \oplus Q \vdash \Delta;}\oplus\vdash$$

$$\frac{\Gamma \vdash \Delta; Q}{\Gamma \vdash \Delta; P \oplus Q}\vdash r\oplus$$

- Existence

$$\frac{\Gamma \vdash \Delta; P[Q/X]}{\Gamma \vdash \Delta; \exists X P}\vdash\exists \qquad\qquad \frac{\Gamma, P \vdash \Delta;}{\Gamma, \exists X P \vdash \Delta;}\exists\vdash$$ [†]

The stoup is used for the positive rules: when a proposition enters[‡] the stoup (rule *Foc*) one must systematically work on its subformulas, up to the moment one reaches shifts, which enable one to empty the stoup. The stoup can also be emptied through an atom rule $X \vdash X$, which plays the role of an *identity axiom.*

Exercise 11. For any formula A exhibit a proof Id_A of $A \vdash A$. That is, the η-*expansion* of the identity.

10.2.4. *Cut-elimination*

Theorem 27. The calculus $\mathbf{MALL}_2$ enjoys cut-elimination.

Proof. This is almost obvious to anybody familiar with linear logic and system $\mathbb{F}$ (Girard 1971). Of course, this is old-style syntax, and one must be pedantic about occurrences, especially repetitions of variables on the left... I suppose if you have reached this point, this will be nothing to you.

There is only one problem: when normalising a cut on an existential formula, we must substitute a proposition P for the variable X in the premise $\Gamma, X \vdash \Delta;$ of the left existence rule. The problem is that – when we substitute P for X in the proof π of $\Gamma, X \vdash \Delta;$ – the 'proof' π' obtained contains sequents of the form $\Lambda, P, \ldots, P \vdash \Pi; \Sigma$ that need not satisfy the stoup constraint. This is fixed by η-expansion. Rather than giving the full – boring and space-consuming – solution, I will take a typical P, namely $Y \oplus (\downarrow Q^{\perp} \otimes \downarrow R^{\perp})$ that concentrates all difficulties. Moreover, I assume that the conclusion of π is $\Gamma, X \vdash \Delta;$ with X not free in Γ and occurring only positively in Δ: the general case is obtained by iteration.

— I first replace X with P everywhere. The conclusion of the 'proof' π' becomes $\Gamma, Y \oplus (\downarrow Q^{\perp} \otimes \downarrow R^{\perp}) \vdash \Delta';$

[†] X is not free in Γ, Δ.

[‡] Bottom up, that is, starting from the conclusion.

https://doi.org/10.1017/S096012950100336X

— Consider the ‘proof’ π''

$$\dfrac{\begin{array}{c}\vdots\,\pi'_1\\ \Gamma, Y \vdash \Delta';\end{array} \qquad \dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\,\pi'_2\\ \Gamma \vdash Q, R, \Delta';\end{array}}{\Gamma, \downarrow Q^{\perp} \vdash R, \Delta';}\,{\scriptstyle \downarrow\vdash}}{\Gamma, \downarrow Q^{\perp}, \downarrow R^{\perp} \vdash \Delta';}\,{\scriptstyle \downarrow\vdash}}{\Gamma, \downarrow Q^{\perp} \otimes \downarrow R^{\perp} \vdash \Delta';}\,{\scriptstyle \otimes\vdash}}{\Gamma, Y \oplus (\downarrow Q^{\perp} \otimes \downarrow R^{\perp}) \vdash \Delta';}\,{\scriptstyle \oplus\vdash} \tag{152}$$

where π'_1 (respectively, π'_2) is obtained by replacing in π' each sequent $\Lambda, P \vdash \Pi; \Sigma$ with $\Lambda, Y \vdash \Pi; \Sigma$ (respectively, $\Lambda \vdash Q, R, \Pi; \Sigma$).

— π'' is still incorrect, since it contains ‘identity axioms’ of the form $Y \vdash ; P$ and of the form $\vdash Q, R; P$ coming from axioms $X \vdash X$. The former are replaced with:

$$\dfrac{\dfrac{}{Y \vdash ; Y}\,{\scriptstyle Y \vdash Y}}{Y \vdash ; Y \oplus (\downarrow Q^{\perp} \otimes \downarrow R^{\perp})}\,{\scriptstyle \vdash r\oplus} \tag{153}$$

whereas the latter are replaced with:

$$\dfrac{\dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\,Id_Q\\ Q \vdash ; Q\end{array}}{Q \vdash Q;}\,{\scriptstyle Foc}}{\vdash Q; \downarrow Q^{\perp}}\,{\scriptstyle \vdash\downarrow} \qquad \dfrac{\dfrac{\begin{array}{c}\vdots\,Id_R\\ R \vdash ; R\end{array}}{R \vdash R;}\,{\scriptstyle Foc}}{\vdash R; \downarrow R^{\perp}}\,{\scriptstyle \vdash\downarrow}}{\vdash Q, R; \downarrow Q^{\perp} \otimes \downarrow R^{\perp}}\,{\scriptstyle \vdash\otimes}}{\vdash Q, R; Y \oplus (\downarrow Q^{\perp} \otimes \downarrow R^{\perp})}\,{\scriptstyle \vdash l\oplus} \tag{154}$$

where Id_Q, Id_R come from Exercise 11, p. 380. □

Remark 18. The calculus is even Church–Rosser, up to permutations of negative (left) rules and re-dispatchings of irrelevant contexts. We will not pursue this: the real guys are the designs, not their bureaucracy!

10.2.5. *Affine logic* For technical reasons, and also because this might be of independent interest, we also consider *affine* logic. $\mathbf{MAAL}_2$ will be the system in which the following version of the identity axiom is allowed:

- Atom (Affine version)

$$\dfrac{}{\Gamma, X \vdash \Delta; X}\,{\scriptstyle X \vdash X}$$

This calculus normalises too, and is reasonably Church–Rosser, provided the rule indicates which occurrence of X has been used on the left.

10.3. *Locative issues*

10.3.1. *Summary* In order to be able to prove soundness and completeness, we must give a precise location to our formulas, especially propositional variables. But, due to the fact that $\mathbf{MALL}_2$ is spiritual – like all existing logical systems – and is not very exciting, and maybe you can content yourself with a summary:

1 Each atom receives a location σ, so that 0 is now written 0_σ and X is written $\theta(X)$ for an appropriate delocation θ such that $\theta(\langle\rangle) = \sigma$.
2 Furthermore, each proposition receives an alternative *left* delocation, on the base $\langle 1 \rangle$. This alternative location is used on the left of sequents.

Let us give some examples:

- Cut

$$\dfrac{\Gamma \vdash \Delta ; P \quad \Gamma', P^l \vdash \Delta' ; \Sigma}{\Gamma, \Gamma' \vdash \Delta, \Delta' ; \Sigma}\ {;Cut} \qquad\qquad \dfrac{\Gamma \vdash \Delta, P ; \quad \Gamma', P^l \vdash \Delta' ;}{\Gamma, \Gamma' \vdash \Delta, \Delta' ;}\ {Cut;}$$

In this example we see the matching between P (right) and its left variant P^l.

- Atom (Identity axiom)

$$\dfrac{}{\theta^a(X) \vdash ; \theta'(X)}\ \theta \vdash \theta'$$

In this example we see the identity axiom between $\theta(X)$ (with left location $\theta(X)^l$, also written $\theta^a(X)$). The name of the rule is unambiguous.

- Shift

$$\dfrac{\Gamma, P^l \vdash \Delta ;}{\Gamma \vdash \Delta ; {\downarrow} P^\perp}\ {\vdash\downarrow} \qquad\qquad \dfrac{\Gamma \vdash \Delta, P ;}{\Gamma, ({\downarrow} P^\perp)^l \vdash \Delta ;}\ {\downarrow\vdash}$$

This example is just to remind you not to forget the '^{l}' on the left.

- Zero

$$\dfrac{}{\Gamma, 0_{\langle 1 \rangle} \vdash \Delta ;}\ 0 \vdash$$

This one is just to remind you that, if we are pedantic, we must indicate the location of 0… But on the left only one location is possible, namely $\langle 1 \rangle$.

10.3.2. *Relative locations* According to the general 'philosophy' of ludics, there is nothing like distinct occurrences of a single formula, only distinct isomorphic formulas. This is why we shall now define the *relative location* $\theta_{A;B}$ of a subformula (old style: occurrence) B of A: if A is located in ξ, R will be located in $\theta(\xi)$.

— $\theta_{A;A} = \theta_{\exists XA;A}$ is the identity.
— If C is a subformula of B, which in turn is a subformula of A, then $\theta_{A;C} = \theta_{A;B} \circ \theta_{B;C}$.
— $\theta_{P \otimes Q;P} = \theta_{P \oplus Q;P} = \varphi$.
— $\theta_{P \otimes Q;Q} = \theta_{P \oplus Q;Q} = \psi$.
— $\theta_{\downarrow P^\perp;P} = s$, where s is the *shift* $s(\sigma) = 0 * \sigma$.

https://doi.org/10.1017/S096012950100336X

10.3.3. *Absolute locations* We should now give absolute locations to our formulas (indeed occurrences). Of course we could do it for the full $\mathbf{MALL}_2$, but this is of no interest. The only formulas that matter are the ones that are used in a given proof π, namely[†]:

1 the formulas of the conclusion;
2 the cut-formulas;
3 their subformulas.

Let U be the tensor product of the finitely many formulas occurring in conclusion and cuts: for technical reasons, it is better to replace U with $\downarrow U^{\perp}$. Then any formula P of our proof is a well-defined subformula-as-occurrence of $\downarrow U^{\perp}$ and has therefore a relative location θ: we locate P in $\theta(\langle\rangle)$. A proposition, together with its relative (hence absolute) location, is called a *right* proposition.

10.3.4. *Left locations* Unfortunately we are still not finished with locative bureaucracy. The problem comes from the fact that we allow several formulas on the left in our sequent calculus, in contrast to what happens in ludics. Typically, the left tensor will introduce two formulas on the left, with the same location $0 * \sigma$ (but with *almost disjoint*, see p. 337, relative locations $s \circ \theta' \circ \varphi, s \circ \theta' \circ \psi$). Worse, if we proceed upwards with a left shift rule, one of these formulas will migrate to the right and will be located in $(s \circ \theta' \circ \varphi \circ s)(\langle\rangle) = 0 * \sigma * 0$ or in $(s \circ \theta' \circ \psi \circ s)(\langle\rangle) = 0 * \sigma * 1$, which are both subloci of $0 * \sigma$.

This is why we introduce *left formulas*: if A has the location $\sigma = \theta(\langle\rangle)$, then A^l is the 'same' formula, delocated by means of the delocation ϖ_σ from σ to $\langle 1\rangle$; the finitely many delocations ϖ_σ are taken almost disjoint. If $\sigma \neq \sigma'$, then

$$\varpi_\sigma(\sigma * i * \tau) \neq \varpi_{\sigma'}(\sigma' * i' * \tau') \tag{155}$$

The absolute location $\langle 1\rangle$ of A^l is disjoint from the right locations, which (and this is the point about $\downarrow U^{\perp}$) are of the form $0 * \ldots$ The *relative location* of A^l is the delocation $\varpi_\sigma \circ \theta$. A proposition, together with its left relative location, is called a *left* proposition.

10.3.5. *Variables* Eventually, what is important is the treatment of variables, so let us make some notational conventions:

— We use X, Y, Z for *plain* variables, not for what is usually called 'occurrence'.
— Each 'occurrence' of X is distinguished by a relative location θ. We use the notation $\theta(X)$ to speak of this occurrence.
— On the left, the 'same' occurrence of X becomes

$$\theta(X)^l = \theta^a(X) = \varpi_\sigma(\theta(X)) \tag{156}$$

10.3.6. *Sequents* We are now dealing with an extremely pedantic syntax (propositions have a precise location, occurrences of variables are written $\theta(X)\ldots$), so it may be interesting to revisit our sequent calculus.

[†] In the case of completeness, the proof π is not given, but we know its conclusion and we are looking for a cut-free proof, hence the discussion applies.

https://doi.org/10.1017/S096012950100336X

Definition 63. (Sequents) A *sequent* is an expression $\Gamma \vdash \Delta;\Sigma$ where Γ, Δ, Σ are finite sets of propositions such that

— Γ consists of left propositions. Γ can naturally be written as a union Γ^a, Γ^n, where Γ^a only consist of left atoms, and Γ^n of the non-atoms.

— Δ, Σ consists of right propositions.

— The propositions in Δ, Σ have pairwise disjoint locations.

— If the *stoup* Σ is non-empty, then Γ^n is empty, and Σ contains exactly one proposition.

10.4. *Soundness*

10.4.1. *Interpretation of propositions and sequents* To every proposition A one associates a bihaviour $\mathbf{A}$: all symbols already have their interpretation. We must only clarify the status of variables: X stands for the unknown bihaviour. Hence we shall interpret X by means of a *value* $\mathbf{X}$, that is, a positive bihaviour of base $\vdash \langle\rangle$ such that $\P\mathbf{X} \subset \wp_*(\mathbb{N}) = \wp_f(\mathbb{N}) - \{\emptyset\}$. The atom $\theta(X)$ will be interpreted by $\theta(\mathbf{X})$; left propositions are interpreted in the same way (only the location changes) – if P has the location σ, then $\mathbf{P}^l = \varpi_\sigma(\mathbf{P})$, and in particular, $\mathbf{X}^a = \theta^a(\mathbf{X})$. The interpretation $\mathbf{\Gamma} \vdash \mathbf{\Delta};\mathbf{\Sigma}$ of the sequent $\Gamma \vdash \Delta;\Sigma$ is the behaviour $\bigotimes \mathbf{\Gamma} \vdash \mathbf{\Delta}, \mathbf{\Sigma}$: for the moment, we do not distinguish between $\Gamma \vdash \Delta;\Sigma$ and $\Gamma \vdash \Delta, \Sigma;$.

10.4.2. *Interpretation of proofs* The interpretation of a proof π of $\Gamma \vdash \Delta;\Sigma$ is a design $\boldsymbol{\pi} \in \mathbf{\Gamma} \vdash \mathbf{\Delta}, \mathbf{\Sigma}$. This design does not depend on the values given to variables. The construction splits into two cases.

• Rules with no stoup

We assume that the premises π', π'' have been interpreted by $\boldsymbol{\pi}', \boldsymbol{\pi}''$, and we define $\boldsymbol{\pi}$:

Rule $;Cut$**:** Basically, this is handled by composition. Assume that P is located in σ. Then $\boldsymbol{\pi}$ is defined by

$$[\![\boldsymbol{\pi}, \mathfrak{D} \otimes \mathfrak{D}', (\mathfrak{E}_\tau), (\mathfrak{E}'_{\tau'})]\!] = [\![\boldsymbol{\pi}'', \mathfrak{D}', (\mathfrak{E}'_{\tau'}), \varpi_\sigma([\![\boldsymbol{\pi}', \mathfrak{D}, (\mathfrak{E}_\tau)]\!])]\!] \tag{157}$$

for all $\mathfrak{D}, (\mathfrak{E}_\tau)$ whose bases are those of $\Gamma, \Delta^\perp$, respectively, and $\mathfrak{D}', (\mathfrak{E}'_{\tau'})$ whose bases are those of $\Gamma', \Delta'^\perp$, respectively.

Rule $\downarrow\ \vdash$**:** Assume that the loci of $\downarrow P^\perp, P$ are $\sigma, \sigma * i$, respectively. Then $\boldsymbol{\pi}$ is defined by:

$$[\![\boldsymbol{\pi}, \mathfrak{E} \otimes \varpi_\sigma(\downarrow\mathfrak{F})]\!] = [\![\boldsymbol{\pi}, \mathfrak{E}, \mathfrak{F}]\!] \tag{158}$$

for all $\mathfrak{E}, \mathfrak{F}$ of bases $\langle 1 \rangle \vdash$ and $\vdash \sigma * i$, respectively, such that $\mathfrak{E} \otimes \varpi_\sigma(\downarrow\mathfrak{F})$ is proper. Equation (158) is completed with

$$[\![\boldsymbol{\pi}, \mathfrak{E}']\!] = \mathfrak{Fid} \tag{159}$$

for all proper $\mathfrak{E}'$ not of the form $\mathfrak{E} \otimes \varpi_\sigma(\downarrow\mathfrak{F})$.

Rule $0\ \vdash$**:** $\boldsymbol{\pi} = \mathfrak{Sk}$.

Rule $\otimes\ \vdash$**:** $\boldsymbol{\pi} = \boldsymbol{\pi}'$.

Rule $\oplus\ \vdash$**:** $\boldsymbol{\pi} = \boldsymbol{\pi}' \cup \boldsymbol{\pi}''$.

Rule $\exists\ \vdash$**:** $\boldsymbol{\pi} = \boldsymbol{\pi}'$.

• Rules with a stoup

There is an additional constraint linked with the stoup: we must also give a *value* to each atom[†] $\theta^a(X)$ occurring in Γ^a, that is, some *proper* design in $\theta^a(\mathbf{X})$; observe that two distinct 'occurrences' of the same atom X^a need not receive the same value, and, since $\varnothing \notin \P\mathbf{X}$, that $\mathfrak{One}$ is not a value. The data 'values for variables + values for the atoms' is called a *valuation* $\mathfrak{V}$, which, by abuse, we identify with the part concerning the atoms. Consistent with this abuse, we use the notation $\bigotimes \mathfrak{V}$ to denote the tensor product of the values of the atoms. We define $\boldsymbol{\pi}_{\mathfrak{V}}$ by

$$[\![\boldsymbol{\pi}_{\mathfrak{V}}, \mathfrak{D}]\!] = [\![\boldsymbol{\pi}, \bigotimes \mathfrak{V} \otimes \mathfrak{D}]\!] \tag{160}$$

for all $\mathfrak{D} \in \boldsymbol{\Gamma}^a$.

The 'stoup constraint' is that $\boldsymbol{\pi}_{\mathfrak{V}}$ is proper with a first action focusing on the locus of the stoup for any valuation of the variables and atoms.

Rule $;$*Cut***:** This is exactly as *Cut*;.

Rule *Foc***:** $\boldsymbol{\pi} = \boldsymbol{\pi}'$.

Rule $\vdash\downarrow$**:** Assume that the loci of $\downarrow P^{\perp}, P$ are $\sigma, \sigma * i$, respectively. Then $\boldsymbol{\pi}$ is defined by:

$$\boldsymbol{\pi}_{\mathfrak{V}} = \downarrow {\varpi_\sigma}^{-1}(\boldsymbol{\pi}'_{\mathfrak{V}}) \tag{161}$$

There is an abuse of notation in the formula (161), since one should indicate delocations for all the loci in Δ: we implicitly take the identity maps.

Rule $\theta \vdash \theta'$**:** $\boldsymbol{\pi}$ is a pseudo-fax (that is, a delocating fax, see Example 18, p. 338) corresponding to θ^a, θ, in which the basic ramification has been reduced to $\theta^a(\wp_*(\mathbb{N}))$ to ensure incarnation.

$$[\![\boldsymbol{\pi}, \theta^a(\mathfrak{D})]\!] = \theta'(\mathfrak{D}) \tag{162}$$

for all positive $\mathfrak{D}$ of base $\vdash \langle\rangle$ distinct from $\mathfrak{One}$; $[\![\boldsymbol{\pi}, \theta^a(\mathfrak{One})]\!] = \mathfrak{Fid}$.

Rule $\otimes$**:** $\boldsymbol{\pi}$ is defined by

$$\boldsymbol{\pi}_{\mathfrak{V}} = \boldsymbol{\pi}'_{\mathfrak{V}'} \otimes \boldsymbol{\pi}''_{\mathfrak{V}''} \tag{163}$$

For the meaning of $\otimes$ when the base is not atomic, see Remark 11, p. 348; $\mathfrak{V}', \mathfrak{V}''$ correspond to the splitting Γ^a, Γ'^a.

Rules $\oplus$**:** $\boldsymbol{\pi} = \boldsymbol{\pi}'$ in the case of $\vdash l\oplus$, $\boldsymbol{\pi} = \boldsymbol{\pi}''$ in the case of $\vdash r\oplus$.

Rule $\vdash \exists$**:** $\boldsymbol{\pi} = \boldsymbol{\pi}'$. This rule needs not preserve incarnation, see discussion *supra.*

10.4.3. *Soundness*

Theorem 28. (Soundness) To every proof π of a closed sequent $\Gamma \vdash \Delta; \Sigma$ in $\mathbf{MALL}_2$ one associates a design $\boldsymbol{\pi} \in \boldsymbol{\Gamma} \vdash \boldsymbol{\Delta}; \boldsymbol{\Sigma}$. The associated design is winning; also, it is material[‡] when the sequent is $\boldsymbol{\Pi}^1$, that is, when Δ, Σ is $\boldsymbol{\Pi}^1$ and Γ is $\boldsymbol{\Sigma}^1$. Moreover, the interpretation is invariant under cut-elimination.

[†] We have so far given values to the variables, not to their left occurrences!

[‡] With respect to the behaviour $\bigcap_{\mathbf{X},\mathbf{Y},\ldots} \boldsymbol{\Gamma} \vdash \boldsymbol{\Delta}; \boldsymbol{\Sigma}$.

Proof. Either we write twenty pages of nonsense or we just say that everything so far written makes this result obvious. In fact, the design $\boldsymbol{\pi}$ is *exact*: this is just the stupid remark that $\boldsymbol{\pi}$ can be written as a *dessin* with 'no weakening'. □

10.4.4. *Soundness for* $\boldsymbol{MAAL_2}$ Soundness holds as well for $\mathbf{MAAL}_2$. But we can no longer guarantee that $\boldsymbol{\pi}$ is parsimonious. The axiom $\Gamma^a, \theta^a(X) \vdash \Delta; \theta'(X)$ is interpreted by a *weak fax* $\mathfrak{F}$ defined by:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ \ldots\sigma * \theta'(i) \vdash 1 * \theta^a(i)\ldots\end{array}}{\ldots \vdash \sigma, 1 * \theta^a(I), 1 * \rho^a(J), \ldots}\,{\scriptstyle(\sigma,\theta'(I))}}{1 \vdash \Upsilon, \sigma}\,{\scriptstyle(1,\mathscr{N})} \tag{164}$$

where:

— $\langle 1\rangle, \Upsilon, \sigma$ are the locations of $\Gamma^a, \Delta, \theta'(X)$, respectively.

— The directory $\mathscr{N}$ consists in all ramifications $\theta^a(I) \cup \rho^a(J) \cup \ldots$, where $\rho^a\ldots$ are the respective *relative* locations of the atoms in Γ^a, and $I, J\ldots \in \wp_*(\mathbb{N})$.

This design $\mathfrak{F}$ is such that:

$$[\![\mathfrak{F}, \bigotimes \mathfrak{B} \otimes \theta^a(\mathfrak{D}), (\mathfrak{E}_\sigma)]\!] = \theta(\mathfrak{D}) \tag{165}$$

for any valuation $\mathfrak{B}, \theta(\mathfrak{D})^a$ of the atoms and any $(\mathfrak{E}_\sigma)$ of respective bases the bases of $\Delta^\perp$; in fact $\mathfrak{F}$ is the only incarnated design enjoying (165).

11. Full completeness of MALL$_2$

Our task is now to prove full completeness for $\mathbf{MALL}_2$. We start with $\mathbf{MAAL}_2$ and prove completeness with respect to uniform and stubborn designs. Next we plug in parsimony, and try to prove completeness with respect to winning, but fail... so we content ourselves with full completeness of $\mathbf{MALL}_2$ with respect to uniform, stubborn and *exact* designs.

11.1. *Full completeness for* $\boldsymbol{MAAL_2}$

In this section, we shall use the expressions 'winning' to mean 'uniform and stubborn'. In the next section we shall plug in parsimony – not quite: exactness! – to get the result for $\mathbf{MALL}_2$.

11.1.1. *The theorem*

Theorem 29. (Completeness) Let $\Gamma \vdash \Delta;$ be a closed Π^1 sequent and let $\mathfrak{D} \in \boldsymbol{\Gamma} \vdash \boldsymbol{\Delta};$ be a material winning design. Then there is a proof π of $\Gamma \vdash \Delta;$ such that $\mathfrak{D} = \boldsymbol{\pi}$.

Proof. We need an *induction loading* corresponding to the case of a general sequent: $\Gamma \vdash \Delta; \Sigma$ being Π^1 (this means that Γ and Δ, Σ are made of Σ^1 and Π^1 propositions, respectively), we define the expression $\mathfrak{D} \in \boldsymbol{\Gamma} \vdash \boldsymbol{\Delta}; \boldsymbol{\Sigma}$:

— $\mathfrak{D} \in \mathbf{\Gamma} \vdash \mathbf{\Delta};\mathbf{\Sigma}$ means that $\mathfrak{D} \in \bigotimes\mathbf{\Gamma} \vdash \mathbf{\Delta},\mathbf{\Sigma}$ for any choice of values $\mathbf{X},\mathbf{Y},\mathbf{Z}\ldots$ for the variables.

— When the stoup Σ is non-empty, $\Gamma = \Gamma^a$ consists of atoms. If $\mathfrak{B}$ is any valuation, then $\mathfrak{D}(\mathfrak{B}) = [\![\mathfrak{D},\mathfrak{B}]\!]$ belongs to $\vdash \mathbf{\Delta},\mathbf{\Sigma}$ – the additional *stoup constraint* is that the first action of $\mathfrak{D}(\mathfrak{B})$ focuses on the locus of Σ.

The proof is by induction on the *size* of the sequent $\Gamma \vdash \Delta;\Sigma$, that is, its total number of connectives and atoms. Concretely, it consists of (given the winning $\mathfrak{D} \in \mathbf{\Gamma} \vdash \mathbf{\Delta};\mathbf{\Sigma}$), producing a final syntactical rule and *ad hoc* premises for the rule (that is, designs belonging to the interpretations of the premises of the rule), and applying the induction hypothesis to the premises. The essential ingredient of the proof is *uniformity*. We say that:

$\mathfrak{D} \in \mathbf{\Gamma} \vdash \mathbf{\Delta};\mathbf{\Sigma}$ is uniform when $\mathfrak{D} \cong_{\mathbf{\Gamma}\vdash\mathbf{\Delta},\mathbf{\Sigma}} \mathfrak{D}$ for any choice $\mathbf{X},\mathbf{Y},\ldots$ of values for the variables, see Section 8.6, p. 370.

The proof will consist of several steps, some of them being independent theorems; the proof occupies the remainder of this section. □

11.1.2. *The uniformity lemma*

Theorem 30. (Uniformity lemma) Assume that $\mathfrak{D} \in \mathbf{\Gamma}^a \vdash \mathbf{\Delta},\mathbf{P};$ is uniform[†], and that for some valuation $\mathbf{X},\mathbf{Y},\ldots,\mathfrak{B}$ of the free variables and the atoms, $\mathfrak{D}(\mathfrak{B})$ starts with a proper rule focusing on the locus ξ of P. Then, for any other valuations $\mathbf{X}',\mathbf{Y}',\ldots,\mathfrak{B}'$ and $\mathbf{X}',\mathbf{Y}',\ldots,\mathfrak{B}''$:

1 The respective first actions of $\mathfrak{D}(\mathfrak{B}')$, $\mathfrak{D}(\mathfrak{B}'')$ are of the form (ξ, I') and (ξ, I'').

2 If $\mathfrak{B}' \cong \mathfrak{B}''$, then $I' \sim_{\mathbf{P}} I''$.

Proof.

1 Change the valuation of the variables: $X, Y, \ldots$ are now all interpreted by $(\mathsf{T}^*, \preceq_{\mathsf{T}^*})$, the greatest positive value, which consists of all designs distinct from $\mathfrak{One}$, and $\mathfrak{D} \cong_{\mathsf{T}^*} \mathfrak{E}$ for all *proper* designs $\mathfrak{D},\mathfrak{E}$. Then $\mathfrak{B}$ is pointwise $\cong_{\mathsf{T}^*}$-comparable to $\mathfrak{B}'$, from which it follows that $\mathfrak{D}(\mathfrak{B}) \cong_{\vdash\mathbf{\Delta},\mathbf{P}} \mathfrak{D}(\mathfrak{B}')$. We apply Proposition 22, p. 371, to conclude.

2 This is immediate from Proposition 22. □

The importance of this lemma is immense. It will be used in the treatment of all right logical rules. We will start with the most important application.

Corollary 30.1. A stubborn uniform design in $\mathfrak{D} \in \mathbf{\Gamma}^a, \vdash \mathbf{\Delta};$ can be obtained by a focalisation rule[‡].

Proof. Select true bihaviours as values for the variables, for example, T^*, and stubborn elements as values for the atoms. Then $\mathfrak{D}_{\mathfrak{B}}$ is stubborn, hence distinct from $\mathfrak{Dai}$. It has therefore a first proper action that focuses on the locus of some $P \in \Delta\ldots$ □

[†] Observe that $\mathbf{P}$ may depend on the value of the variables, think of $P = X$.

[‡] It is applied to the same $\mathfrak{D}$, since focalisation does not change the interpretation.

11.1.3. *The polymorphic lemma* This is by far the most important result, which enables one to reconstruct the 'identity' axioms.

Theorem 31. (Polymorphic lemma) Let $\mathfrak{F}$ be a uniform design in $\mathbf{\Gamma}^a \vdash \mathbf{\Delta}; \theta(\mathbf{X})$; then one can find an atom $\rho^a(X) \in \Gamma^a$ such that for any valuation $\mathfrak{B}, \rho^{\mathfrak{a}}(\mathfrak{D})$ of the atoms and any $(\mathfrak{E}_\sigma)$ of respective bases the bases of $\Delta^\perp$:

$$[\![\mathfrak{F}, \bigotimes \mathfrak{B} \otimes \rho^a(\mathfrak{D}), (\mathfrak{E}_\sigma)]\!] = \theta(\mathfrak{D}) \tag{166}$$

In other words, $\mathfrak{F}$ is a weak fax.

We begin with a lemma.

Lemma 31.1. Assume that the equation

$$[\![\mathfrak{F}, \bigotimes \mathfrak{B} \otimes \rho^a(\mathfrak{D}), (\mathfrak{E}_\sigma)]\!] \supset \theta(\mathfrak{D}) \tag{167}$$

holds for any valuation $\mathfrak{B}, \rho^{\mathfrak{a}}(\mathfrak{D})$ of the atoms and any $(\mathfrak{E}_\sigma)$ of respective bases the bases of $\Delta^\perp$. Then equality, that is, Equation (166) holds.

Proof. Assume that the inclusion (167) is strict for a certain choice $\mathfrak{B}, \rho^{\mathfrak{a}}(\mathfrak{D}), (\mathfrak{E}_\sigma)$ of arguments. Hence

$$[\![\mathfrak{F}, \bigotimes \mathfrak{B} \otimes \rho^a(\mathfrak{D}), (\mathfrak{E}_\sigma)]\!] = \theta(\mathfrak{D}') \tag{168}$$

for some $\mathfrak{D}' \supsetneq \mathfrak{D}$. Choose $\mathfrak{D}'' \supsetneq \mathfrak{D}$ such that $\mathfrak{D}' \cup \mathfrak{D}''$ *is not* a design. Then

$$[\![\mathfrak{F}, \bigotimes \mathfrak{B} \otimes \rho^a(\mathfrak{D}''), (\mathfrak{E}_\sigma)]\!] \supset \theta(\mathfrak{D}' \cup \mathfrak{D}'') \tag{169}$$

which is a contradiction. □

We can now establish Theorem 31.

Proof. We start with some simplifying hypotheses:

Δ empty: Assume that the property holds in the case of an empty Δ. Then we can apply it to $[\![\mathfrak{F}, (\mathfrak{S}\mathfrak{k}_\sigma)]\!]$, and we get

$$[\![\mathfrak{F}, \bigotimes \mathfrak{B} \otimes \rho^a(\mathfrak{D}), (\mathfrak{S}\mathfrak{k}_\sigma)]\!] = \theta(\mathfrak{D}) \tag{170}$$

Hence, since $\mathfrak{E}_\sigma \supset \mathfrak{S}\mathfrak{k}_\sigma$, we conclude that

$$[\![\mathfrak{F}, \bigotimes \mathfrak{B} \otimes \rho^a(\mathfrak{D}), (\mathfrak{E}_\sigma)]\!] \supset \theta(\mathfrak{D}) \tag{171}$$

Lemma 31.1, p. 388, forces the equality (166).

One variable: Assume that the result holds when the only variable is X. If several variables 'occur' as atoms, we can set them all equal to X and apply the result in this case: if the occurrence of $\rho^a(X)$ is obtained by substituting the variable X for $Y \neq X$ in $\rho^a(Y)$, then

$$\mathfrak{D} \in \rho^a(\mathbf{Y}) \Rightarrow \theta(\mathfrak{D}) \in \theta(\mathbf{X}) \tag{172}$$

for all values $\mathbf{X}, \mathbf{Y}$ and $\mathfrak{D} \in \mathbf{Y}$, which is absurd.

Two left atoms: We are therefore left with the case of several occurrences of the variable X on the left. For notational simplicity, we restrict to the binary case $\Gamma^a = \rho(X), \rho'(X)$. The general case would be as follows:

https://doi.org/10.1017/S096012950100336X

Zero atom: This is trivially impossible – $\bigcap_{\mathbf{G}} \theta(\mathbf{G}) = \mathbf{0}$...

One atom: Just add a dummy second variable...

Three or more atoms: There are more cases to consider – first $I \neq I' \neq I''$, then $I = I' \neq I''$, then $I = I' = I''$...

We therefore assume that $\mathfrak{F} \in \rho(\mathbf{X}), \rho'(\mathbf{X}) \vdash ; \theta(\mathbf{X})$ for all values $\mathbf{X}$ and is uniform. Then we must prove that either

$$[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D}')]\!] = \theta(\mathfrak{D}) \tag{173}$$

for all $\mathfrak{D}, \mathfrak{D}'$ or

$$[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D}')]\!] = \theta(\mathfrak{D}') \tag{174}$$

for all $\mathfrak{D}, \mathfrak{D}'$.

We shall establish the result in several steps; in what follows, $\mathbf{D}, \mathbf{D}'$... stand for the 'singleton bihaviours' constructed from $\mathfrak{D}^{\perp\perp}$, $\mathfrak{D}'^{\perp\perp}$, ... in Proposition 19, p. 366, and 'proper design' stands for 'proper design distinct from $\mathfrak{One}$'.

1 Let $\mathfrak{D}, \mathfrak{D}'$ be proper, with distinct first actions. If $\mathbf{X} = \mathbf{D} \oplus \mathbf{D}'$, then we have that $[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D}')]\!]$ is a uniform proper element of $\theta(\mathbf{D})$. By Proposition 19, p. 366, these elements are equal to $\mathfrak{D}$ or $\mathfrak{D}'$, up to incarnation. Hence, for instance,

$$[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D}')]\!] \supset \theta(\mathfrak{D}) \tag{175}$$

But Lemma 31.1, p. 388 (or rather its proof) implies equality: take $\mathfrak{D}''$ such that $\mathfrak{D} \cup \mathfrak{D}''$ is not a design. We conclude that either (173) or (174) holds. Let us assume that equation (173), say, holds for one particular choice of proper $\mathfrak{D}, \mathfrak{D}'$, with distinct first actions.

2 Let $\mathfrak{D}''$ be another proper design whose first action is distinct from the first action of $\mathfrak{D}$. Let $\mathbf{G}$ be the bihaviour $(\{\mathfrak{D}', \mathfrak{D}''\}, \simeq)^{\perp\perp}$, where $\simeq$ is such that $\mathfrak{D}' \simeq \mathfrak{D}''$. If $\mathbf{X} = \mathbf{D} \oplus \mathbf{G}$, then, by uniformity:

$$[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D}')]\!] \cong_{\theta(\mathbf{X})} [\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D}'')]\!] \tag{176}$$

which forces (since $\mathfrak{D} \not\simeq \mathfrak{D}''$) equality: Equation (173) holds for *any* choice $\mathfrak{D}'$, as long as the first actions remain distinct.

3 Symmetrically, let $\mathfrak{D}''$ be another proper design whose first action is distinct from the first action of $\mathfrak{D}'$. Let $\mathbf{H}$ be the bihaviour $(\{\mathfrak{D}, \mathfrak{D}''\}, \simeq)^{\perp\perp}$, where $\simeq$ is such that $\mathfrak{D} \simeq \mathfrak{D}''$. If $\mathbf{X} = \mathbf{H} \oplus \mathbf{D}'$, then, by uniformity:

$$[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D}')]\!] \cong_{\theta(\mathbf{X})} [\![\mathfrak{F}, \rho(\mathfrak{D}'') \otimes \rho'(\mathfrak{D}')]\!] \tag{177}$$

which forces (since $\mathfrak{D} \not\simeq \mathfrak{D}'$) equality:

$$[\![\mathfrak{F}, \rho(\mathfrak{D}'') \otimes \rho'(\mathfrak{D}')]\!] = \theta(\mathfrak{D}'') \tag{178}$$

Putting things together, Equation (173) holds for *all* proper $\mathfrak{D}, \mathfrak{D}'$, as long as their first actions differ.

4 Assume that $\mathfrak{D} = \mathfrak{D}'$. If $\mathbf{X} = \mathbf{D}$, then $[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D})]\!]$ is a uniform proper element of $\theta(\mathbf{D})$. By Proposition 19, p. 366, this forces

$$[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D})]\!] = \theta(\mathfrak{D}) \tag{179}$$

(in fact an inclusion $\supset$, which is easily transformed into an equality by Lemma 31.1, p. 388.) As a consequence, if $\mathfrak{Sk}^+$ denotes the smallest design included in $\mathfrak{D}$:

$$[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{Sk}^+)]\!] \subset \theta(\mathfrak{D}) \tag{180}$$

5 Consider the bihaviour $\mathbf{K}$ defined from $\varphi(\mathbf{D}) \oplus \psi(\mathbf{D})$, the 'symmetric sum' of Subsection 8.3.4, p. 368. Then

$$[\![\mathfrak{F}, \rho(\varphi(\mathfrak{D})) \otimes \rho'(\psi(\mathfrak{Sk}^+))]\!] \cong [\![\mathfrak{F}, \rho(\varphi(\mathfrak{D})) \otimes \rho'(\varphi(\mathfrak{Sk}^+))]\!] \tag{181}$$

Hence, using (180) and (173) with $\mathfrak{D}' = \psi(\mathfrak{Sk}^+)$:

$$\theta(\varphi(\mathfrak{D})) \cong [\![\mathfrak{F}, \rho(\varphi(\mathfrak{D})) \otimes \rho'(\varphi(\mathfrak{Sk}^+))]\!] \subset \theta(\varphi(\mathfrak{D})) \tag{182}$$

which by Proposition 19, p. 366, forces the equality:

$$[\![\mathfrak{F}, \rho(\varphi(\mathfrak{D})) \otimes \rho'(\varphi(\mathfrak{Sk}^+))]\!] = \theta(\varphi(\mathfrak{D})) \tag{183}$$

From which we deduce

$$[\![\mathfrak{F}, \rho(\varphi(\mathfrak{D})) \otimes \rho'(\varphi(\mathfrak{D}'))]\!] \supset \theta(\varphi(\mathfrak{D})) \tag{184}$$

for all $\mathfrak{D}'$ with the same first action as $\mathfrak{D}$. We have used φ, ψ to be consistent with the definition of Section 8.3.4, p.368, but the choice of the delocations is arbitrary. In particular, we could have selected ϕ', ψ', such that $\phi'(\mathfrak{D}) = \mathfrak{D}$, $\phi'(\mathfrak{D}') = \mathfrak{D}'$, and we conclude that

$$[\![\mathfrak{F}, \rho(\mathfrak{D}) \otimes \rho'(\mathfrak{D}')]\!] \supset \theta(\mathfrak{D}) \tag{185}$$

which entails equality, again by Lemma 31.1, p. 388.

The Equation (173) has therefore been established in all cases: for proper $\mathfrak{D}, \mathfrak{D}'$ with distinct first actions, and for proper $\mathfrak{D}, \mathfrak{D}'$ with the same first action. □

11.1.4. *Left logical rules* Assume that $\mathfrak{D} \in \mathbf{\Gamma} - \mathbf{P}^l, \mathbf{P}^l \vdash \mathbf{\Delta}$;, then we can find the 'last rule'. The task is to find 'premises' $\mathfrak{D}', \mathfrak{D}'', \ldots$ in the appropriated bihaviours.

Shift: If $P = \downarrow Q^\perp$, and the respective loci of $P, Q^\perp$ are $\sigma, \sigma * i$, then $\mathfrak{D}'$ is defined by

$$[\![\mathfrak{D}', \mathfrak{E}, F]\!] = [\![\mathfrak{D}, \mathfrak{E} \otimes \varpi_\sigma(\downarrow\mathfrak{F})]\!] \tag{186}$$

for all $\mathfrak{E}, \mathfrak{F}$ of bases $\langle 1 \rangle \vdash$ and $\vdash \sigma * i$, respectively, such that $\mathfrak{E} \otimes \varpi_\sigma(\downarrow\mathfrak{F})$ is proper – compare with (158).

Zero: If $P = 0$, we need not produce any premise. Observe that the condition $\mathfrak{D} \in \mathbf{\Gamma} - \mathbf{P}^l, \mathbf{P}^l \vdash \mathbf{\Delta}$; is vacuously satisfied, and that the hypothesis of incarnation forces $\mathfrak{D}$ to be a skunk.

Plus: If $P = Q \oplus R$, then $\mathfrak{D}$ is a disjoint union $\mathfrak{D}' \cup \mathfrak{D}''$, with $\mathfrak{D}' \in \mathbf{\Gamma} - \mathbf{P}^l, \mathbf{Q}^l \vdash \mathbf{\Delta}$; and $\mathfrak{D}'' \in \mathbf{\Gamma} - \mathbf{P}, \mathbf{R}^l \vdash \mathbf{\Delta}$;.

Tensor: If $P = Q \oplus R$, just take $\mathfrak{D}' = \mathfrak{D}$.

Existence: If $P = \exists X Q$, just take $\mathfrak{D}' = \mathfrak{D}$.

In all cases, the premises found are winning. Observe that the choice of $P \in \Gamma$ is not unique: the sequence of left rules is up to permutation. In a calculus with synthetic connectives, such an ambiguity of the proof would not occur.

https://doi.org/10.1017/S096012950100336X

11.1.5. *Right logical rules* Assume that $\mathfrak{D} \in \mathbf{\Gamma}^a \vdash \mathbf{\Delta}; \mathbf{P}$ is given to us as a minimal *dessin*, see p. 315. Then we can find the 'last rule'; we assume that the relative location of P is θ. The task is to find 'premises' $\mathfrak{D}', \mathfrak{D}'', \ldots$ of the appropriate type. We first consider the case $\Gamma^a = \emptyset$.

Atom: Formally, we write $\theta({\varpi_\sigma}^{-1}(\mathfrak{V}))$.

Shift: If $P = \downarrow Q^\perp$, and P, Q are located in $\sigma, \sigma * i$, then $\mathfrak{D} = \downarrow \mathfrak{D}_{\mathrm{I}}$ for an appropriate $\mathfrak{D}_{\mathrm{I}}$. Let $\mathfrak{D}' = \varpi_{\sigma * i}(\mathfrak{D}_{\mathrm{I}})$. Compare this with (161).

Tensor: If $P = Q \otimes R$, find a last *exact* rule in $\mathfrak{D}$[†]. This rule induces a unique splitting between the contexts of Q, R, say Δ', Δ'', and we can write $\mathfrak{D} = \mathfrak{D}' \otimes \mathfrak{D}''$. We 'close' the context by forming $[\![\mathfrak{D}, (\mathfrak{E})_\sigma]\!]$ with appropriate $(\mathfrak{E})_\sigma \in \mathbf{\Delta}^\perp$. Then we use the Completeness Theorem (Theorem 20, p. 354) to conclude that $[\![\mathfrak{D}', (\mathfrak{E})_\sigma]\!] \in \mathbf{Q}$, $[\![\mathfrak{D}'', (\mathfrak{E})_\sigma]\!] \in \mathbf{R}$. Letting $(\mathfrak{E})_\sigma \in \mathbf{\Delta}^\perp$ vary, we conclude that $\mathfrak{D}' \in \vdash \mathbf{\Delta}', \mathbf{Q}$ and $\mathfrak{D}'' \in \vdash \mathbf{\Delta}'', \mathbf{R}$.

Zero: This does not apply, since $\mathbf{0}$ contains no proper design.

Plus: If $P = Q \oplus R$, the ramification of the first action 'belongs' to P or Q and the disjunction is exclusive. Depending on the case, either $\mathfrak{D}' = \mathfrak{D} \in \mathbf{Q}$ or $\mathfrak{D}'' = \mathfrak{D} \in \mathbf{R}$, the disjunction being exclusive: we use the Completeness Theorem (Theorem 11, p. 343) together with a back and forth argument imitated from the tensor case.

Existence: This does not apply, since P is $\Pi^1 \ldots$ fortunately, since there is no *internal* completeness theorem in that case.

Let us now consider the general case. If $P = \theta(X)$, we can apply the results of Subsection 11.1.3, p. 388, to conclude: the material $\mathfrak{D}$ is equal to the design (164), p. 386, that is, it comes from an *affine* identity axiom.

Otherwise, starting with with $\mathfrak{D} \in \mathbf{\Gamma}^a \vdash \mathbf{\Delta}; \mathbf{P}$, we choose values for the variables together with a valuation $\mathfrak{V}$ for the atoms: then $\mathfrak{D}_{\mathfrak{V}} = [\![\mathfrak{D}, \bigotimes \mathfrak{V}]\!] \in \mathbf{\Delta} \vdash \mathbf{P}$. Then we can write $\mathfrak{D}_{\mathfrak{V}}$ as the result of a logical rule (shift, tensor, plus), applied to one or two premises $\mathfrak{D}', \mathfrak{D}''$. Then we define $\mathfrak{D}', \mathfrak{D}''$ by

$$[\![\mathfrak{D}', \bigotimes \mathfrak{V}]\!] = \mathfrak{D}'_{\mathfrak{V}'} \tag{187}$$

$$[\![\mathfrak{D}'', \bigotimes \mathfrak{V}]\!] = \mathfrak{D}'_{\mathfrak{V}''} \tag{188}$$

where $\mathfrak{V} = \mathfrak{V}', \mathfrak{V}''$, following the splitting of the context. We are done… provided the definition makes sense. This offers no difficulty, except in two cases:

Plus: The choice between left and right might depend on the valuation. Fortunately, the Uniformity Lemma (Theorem 30, p. 387) shows that this is not the case.

Tensor: Assume for simplicity that $\Gamma^a \vdash \Delta$ is of the form $\rho(X) \vdash R$ and that $\mathfrak{D} \in \rho(\mathbf{X}) \vdash \mathbf{R}; \mathbf{P} \otimes \mathbf{Q}$. Then one may define $\mathfrak{D}', \mathfrak{D}''$ by $\mathfrak{D}_{\mathfrak{V}} = \mathfrak{D}'_{\mathfrak{V}} \otimes \mathfrak{D}''_{\mathfrak{V}}$, and apply the induction hypothesis to $\mathfrak{D}' \in \mathfrak{D} \in \rho(\mathbf{X}) \vdash \mathbf{R}; \mathbf{P}$, $\mathfrak{D}'' \in \mathfrak{D} \in \rho(\mathbf{X}) \vdash \mathbf{R}; \mathbf{Q}$, which yields proofs π', π'' of $\rho(X) \vdash R; P$ and $\rho(X) \vdash R; Q$. The problem is that $\rho(X), R$ have been given to both premises.

— Assume that, say, $\mathfrak{D}'$ contains no affine identity $\rho \vdash \theta'$. Then $\mathfrak{D}'_V$ does not depend on the choice of the proper value $\mathfrak{V}$ and when I write $\mathfrak{D}_V = \mathfrak{D}'_V \otimes \mathfrak{D}''_V$, either the

[†] This has nothing to do with *parsimony*, which can only fail in case of positive rules with an empty ramification.

context R is always given to $\mathfrak{D}'_V$ or this never happens and it can always be given to $\mathfrak{D}''_V$.

— If $\mathfrak{D}', \mathfrak{D}''$ both contain affine identities, then $\rho \vdash \theta'$, $\rho \vdash \theta''$, which are interpreted by weak faxes. In $\mathfrak{D}$, above the negative action $(1, \{\rho(0)\})$ stands a pitchfork $\vdash \sigma * 0, \tau, \lambda$; the locus $\sigma * 0$ that is a focus in both weak faxes must be used as a focus in both $\mathfrak{D}', \mathfrak{D}''$, contradicting propagation.

11.1.6. *End of the proof* We have been able, given a winning design in $\mathbf{\Gamma} \vdash \mathbf{\Delta}$, to inductively construct premises, which in turn correspond to winning designs. If the *size* of a sequent is the total number of its symbols, then a premise has a strictly smaller size than the conclusion of the same rule. Hence the process of inductively finding premises eventually stops with a left rule for 0, or an identity axiom, and we have produced the desired proof π.

Remark 19. Every Π^1 proposition can – using prenex forms – be put under the form $\forall X_1 \ldots \forall X_n P$ where P is quantifier-free. Hence, contrary to a superficial impression, the general Π^1 case reduces to the pure propositional case.

11.2. *Full completeness for* $\boldsymbol{MALL_2}$

Here we should return to the true meaning of 'winning', that is, plug in parsimony. Unfortunately, the result fails; in the next theorem, 'winning' means uniform, stubborn and *exact*.

Theorem 32. (Completeness) Let $\Gamma \vdash \Delta$; be a closed Π^1 sequent and let $\mathfrak{D} \in \mathbf{\Gamma} \vdash \mathbf{\Delta}$; be a material winning design. Then there is a proof π of $\Gamma \vdash \Delta$ such that $\mathfrak{D} = \boldsymbol{\pi}$.

This reduces to showing that a proof π of $\mathbf{MAAL}_2$ such that $\boldsymbol{\pi}$ is exact is (modulo some adequate re-dispatching of irrelevant contexts) a proof of the same sequent in $\mathbf{MALL}_2$.

11.2.1. *Parsimonious vs. exact* An exact design is obviously parsimonious, and the intuition is that

$$\textbf{\textit{Parsimonious}} = \textbf{\textit{Exact}}$$

Unfortunately, as the following exercises show, this is wrong.

Exercise 12. Show[†] that the inexact design

$$\dfrac{\dfrac{\dfrac{}{\vdash \sigma}\,{\scriptstyle(\sigma,\emptyset)}}{\xi 0 \vdash \sigma}\,{\scriptstyle(\xi 0,\emptyset)} \qquad \dfrac{\dfrac{}{\vdash \tau}\,{\scriptstyle(\tau,\emptyset)}}{\xi 1 \vdash \tau}\,{\scriptstyle(\xi 1,\emptyset)}}{\vdash \xi, \sigma, \tau, \lambda}\,{\scriptstyle(\xi,\{0,1\})} \tag{189}$$

is parsimonious (with respect to its principal behaviour).

[†] This counter-example is due to Claudia Faggian.

Exercise 13. Show that the inexact design

$$\cfrac{\cfrac{\cfrac{}{\vdash \lambda}{\scriptstyle (\lambda,\emptyset)} \qquad \cfrac{}{\vdash \xi 00,\lambda}{\scriptstyle (\xi 00,\emptyset)}}{\xi 0 \vdash \lambda}{\scriptstyle (\xi 0,\{\emptyset,\{0\}\})} \qquad \cfrac{}{\xi 1 \vdash}{\scriptstyle (\xi 1,\emptyset)}}{\vdash \xi,\lambda}{\scriptstyle (\xi,\{0,1\})} \tag{190}$$

is parsimonious (with respect to its principal behaviour).

The two counter-examples are different in nature:

— (189) exhibits an inexact slice that cannot be consumed during a single normalisation: in order to do so, the counterdesign should allow a ‘simultaneous’ focalisation on $\xi 0, \xi 1$ in the pitchfork $\vdash \xi 0, \xi 1$.

— (190) exhibits an inexact design with two exact slices: only one of them can be consumed during a single normalisation. In order to consume both of them, the counterdesign should be allowed to perform two positive actions with the same focus $\xi 0$.

In order to get a completely satisfactory result, one should therefore try to modify our definitions so as to allow *parallel actions* and reuse of actions, with possible *non-deterministic* aspects, due to superimposition of loci. But that is another story.

11.2.2. *Proof of the theorem* I will hand wave in this final step, since there is little in this proof.

1 First of all we start with a π that is a proof in $\mathbf{MAAL}_2$, and $\boldsymbol{\pi}$ is assumed to be exact, which means that it can be written as a *dessin* $\mathfrak{D}$ with an exact maintenance of contexts.

2 Then, ‘following’ $\mathfrak{D}$, we can reconstruct another proof π', in the same $\mathbf{MAAL}_2$. Here lies the hand waving, but one should not waste paper!

3 One eventually reaches identity axioms, which correspond to some sort of exact faxes. We are reduced to showing that these ‘weak faxes’ indeed correspond to identity axioms of $\mathbf{MALL}_2$, which we do in the next lemma.

Lemma 32.1. The ‘weak fax’ $\mathfrak{F}$ introduced in (164), p.386 is exact iff $\Gamma^a = \Delta = \emptyset$.

Proof. We only check the necessity of the condition. Assume that one of Γ^a, Δ is non-empty. Then the rule $(\sigma, \theta'(I))$ of the display (164) is inexact, which does not mean that there is no alternative *dessin* of the same with an exact maintenance. But then one locus, say ξ, of the missing context is dispatched to one of the premises, say – assuming that $0 \in I$ – the one of index $\theta'(0)$. This yields the alternative premise

$$\sigma * \theta'(i) \vdash 1 * \theta^a(i), \xi \tag{191}$$

still justified by the same fax $\mathfrak{Fax}_{\sigma*\theta'(i),1*\theta^a(i)}$, which harbours the following slice:

$$\cfrac{\cfrac{}{\vdash 1 * \theta^a(i), \xi}{\scriptstyle (1*\theta^a(i),\emptyset)}}{\sigma * \theta'(i) \vdash 1 * \theta^a(i), \xi}{\scriptstyle (\sigma*\theta'(i),\{\emptyset\})} \tag{192}$$

which definitely contains an inexact rule, namely $(1 * \theta^a(i), \emptyset)$. □

11.3. *Other logics?*

Full completeness also applies for non-stubborn designs – independently of parsimony. The following principle must be added

- Daimon

$$\frac{}{\Gamma^a \vdash \Delta;}\maltese$$

In this way we obtain two additional systems (one linear and one affine) that are of extremely good quality. One may object that these two additional systems are inconsistent – but this is only in terms of provability (everything is provable) and not in terms of proofs.

However, we know no way to dispense with uniformity: for instance, the set of stubborn and parsimonious designs in a closed Π^1 bihaviour is not even denumerable.

Exercise 14.

1 Write the two uniform designs in (for all $\mathbf{X}$) $\rho(\mathbf{X}) \otimes \rho'(\mathbf{X}) \vdash; \theta(\mathbf{X}) \otimes \theta'(\mathbf{X})$, called *identity* and *flip*.

2 For any partition of $\wp_*(\mathbb{N}) \times \wp_*(\mathbb{N})$ into two classes, define a (non-uniform) stubborn design that behaves as *identity* or *flip* depending on the pair (I, J). Are these designs parsimonious?

Exercise 15.

1 Write the two uniform designs in (for all $\mathbf{X}$) $\rho(\mathbf{X}) \oplus \rho'(\mathbf{X}) \vdash; \theta(\mathbf{X}) \oplus \theta'(\mathbf{X})$, called *identity* and *flip*.

2 For any partition of $\wp_*(\mathbb{N})$ into two classes, define a (non-uniform) stubborn design that behaves as *identity* or *flip* depending on I. Are these designs parsimonious?

The extreme role of uniformity emphasises the importance of… non-uniform designs, which ‘fill the space’, and other losers. These losers are more important than the boring winners. For me the only interest of full completeness is the disclosure of these losers[†]:

The actual inhabitants of the logical universe.

Appendix A. A pure waste of paper

J'ai composé cette histoire – simple, simple, simple,
Pour mettre en fureur les gens graves – graves, graves, graves,
Et amuser les enfants – petits, petits, petits.
Charles Cros, Le coffret de santal, 1873.

[†] Anyway, the winners have been available symbolically, that is, as formal proofs, for at least a century.

• Abduction

It is fun to read Conan Doyle, but the logical method consisting of deducing the hypotheses from the conclusion hardly works in real life... unless you know the answer in advance. So-called 'abduction' is by far the worst paralogic. It seems that the basic mistake lies in a confusion between the actual sense of rules (upwards) and their formal writing (downwards).

See: **Arsène Lupin, Black Mass, Formal, Formalisable, Kepler, Lemma, Nostradamus, Notations, Numerology, Paralogics, Proof-search, Sense of rules, Sherlock Holmes**.

• Abstraction

Abstraction consists of treating things not as they are, but *as they should be*. Typically, when I replace a Michelin `XH1 TL 86H` with a `XH1 TL 86H`, I am not replacing a thing with its exact copy, I am just matching a specification. The new tyre is different from the one it replaces, but it behaves in the same way, as long as I use it as a tyre; differences, like this small change of colour, do not matter. Observe that the specifications of tyres are sufficiently precise so as to allow the replacement of only one of your two front tyres with another of the same type considered as identical.

This applies to industry, which deals, say, not with food, but with the idea of food, think of McDonald's. Craft deals with objects as they are, no replacement is possible... for better or worse.

Abstraction should not be confused with spiritualism, which is just abstraction from location.

See: **Implicit, Locative logic, Money, Pauperism, Specification, Spiritualism, Tradition**.

• Action

An action plays the role of a logical rule in ludics, especially in desseins: its form is (ϵ, ξ, I), where ϵ is the polarity (usually omitted), ξ is the focus and I the ramification. This basically means something like 'A formula of polarity ϵ located in ξ, has been created by means of a logical rule involving immediate subformulas of polarity $-\epsilon$ located in the $\xi * i$ for $i \in I$'. Observe that neither the rule nor the formulas matter. Each action comes with its opposite, obtained by swapping the polarity.

There is also an improper action, the Daimon.

See: **Daimon, Dessein, Focus, Maul, Ramification, Slice**.

• Adjunction

One of the main properties of designs is associativity of cut-elimination, that is, $\llbracket\llbracket\mathfrak{D}, \mathfrak{E}\rrbracket, \mathfrak{F}\rrbracket = \llbracket\mathfrak{D}, \mathfrak{E}, \mathfrak{F}\rrbracket$. This property induces in turn the existence of two adjoints for each tensor product: for instance $\olessthan$ has two adjoints that necessarily enjoy the associative equation $([\mathfrak{B}]\mathfrak{F})[\mathfrak{A}] = [\mathfrak{B}](\mathfrak{F}[\mathfrak{A}])$, and the cotensor $\mathbf{G} \ltimes \mathbf{H}$ can be seen either as the set of 'functions' from $\mathbf{G}^\perp$ to $\mathbf{H}$ or the set of 'functions' from $\mathbf{H}^\perp$ to $\mathbf{G}$. Without this equivalence between the two adjunctions, one would be forced to require that our functions map $\mathbf{G}^\perp$ into $\mathbf{H}$ *and* $\mathbf{H}^\perp$ into $\mathbf{G}$, and this would be the end of associativity... (and distributivity as well): typically $\mathbf{G} \ltimes (\mathbf{H} \ltimes \mathbf{K}) = (\mathbf{G} \ltimes \mathbf{H}) \ltimes \mathbf{K}$ because we can express these two behaviours as made of, respectively:

— *Functions* from $\mathbf{G}^\perp$ to $\mathbf{H} \ltimes \mathbf{K}$, which turn out to be functions from $\mathbf{G}^\perp \times \mathbf{H}^\perp$ to $\mathbf{K}$, that is, functions from $\mathbf{G}^{\perp} \times \mathbf{H}^{\perp} \times \mathbf{K}^{\perp}$ to $\mathbf{0}$ (the unique behaviour of base $\vdash$, which is the dualiser).

— *Functions* from $\mathbf{K}^{\perp}$ to $\mathbf{G} \ltimes \mathbf{H}$, which turn out to be functions from $\mathbf{H}^{\perp} \times \mathbf{K}^{\perp}$ to $\mathbf{G}$, that is, functions from $\mathbf{G}^{\perp} \times \mathbf{H}^{\perp} \times \mathbf{K}^{\perp}$ to $\mathbf{0}$.

In the first case, we must absolutely take our argument in $\mathbf{G}^{\perp}$, for we do not know how to deal with arguments in $\mathbf{G}^{\perp} \otimes \mathbf{H}^{\perp}$.

The tensor product is therefore built in three steps:

1 First as an ethics $\mathbf{G} \copyright \mathbf{H}$, the set of all tensor products of a design of $\mathbf{G}$ with a design of $\mathbf{H}$.

2 Then its dual, which can be handled by the two adjunctions, and has therefore good properties such as associativity.

3 Then the real tensor product $(\mathbf{G} \copyright \mathbf{H})^{\perp\perp}$, which inherits the good properties of its predual.

See: **Associativity, Closure principle, Commutativity, Distributivity, Dualiser, Tensor product, Non-associative logic**.

• Admissible rule

An admissible rule is a rule of the form ‘If A is provable, then B is provable’, which is a statement strictly weaker than its contextual version ‘If A is provable in context Γ, then B is provable in the same context’, which amounts to the provability of the implication, take $\Gamma = A^{\perp}$. The specificity of ludics is to define logical connectives without context: of course not all inhabitants of the behaviour $\mathbf{A}$ corresponding to A are ‘proofs’, that is, are winning, but, winning or losing, a design $\mathfrak{D} \in \mathbf{A}$ is close to a proof of A. In particular, it is easy, using the ‘methods’ of admissibility, to prove inclusion between behaviours, for instance $\forall d(\mathbf{G}_d \vee \mathbf{H}_d) \subset (\forall d\, \mathbf{G}_d) \vee (\forall d\, \mathbf{H}_d)$, which is, in fact, an equality. In the usual intuitionistic logic, the inclusion is just an admissible rule, in ludics[†] it becomes a true implication. This shows that the usual logic is badly incomplete; I do not know whether we can fix it by means of new principles or if this is a fundamental phenomenon.

See: **Completeness (internal), Harmony, Prenex form, Winning**.

• Affine logic

We understand ‘affine’ to mean the refusal of contraction, since in the absence of contraction implication is handled by means of affine functions, which satisfy $f(a \cup b) = f(a) \cup f(b)$, but perhaps not $f(\varnothing) = \varnothing$. Such a restriction first appeared in a paper by Grishin (Grishin 1982). However, this was hardly more than an isolated remark, and not even connected with the functional view of logic. After the invention of linear logic, everybody took it for granted that there was – nearby linear logic – a well-defined affine logic, and this was quite true, up to the semantics of proofs: but how can we compose a weakening on the left with a weakening on the right? There is a solution, namely to work with an intuitionistic version (weakening only on the left), which has certain qualities, see for instance the work of Asperti (Asperti 1998). But ‘intuitionistic affine’ is a bit weird for a real logic.

[†] Since we have not yet treated exponentials, replace $\vee$ with the linear $\oplus$, which is almost the same.

Ludics changes the problem, since the critical pair of Lafont disappears, due to polarities. Then affine logic gets a reasonable status: the maps are real linear maps, though possibly not parsimonious. The completeness theorem for $\mathbf{MALL}_2$ is indeed proved using an intermediate step, namely full completeness for $\mathbf{MAAL}_2$, its affine version. As to the eventual status of affine logic, I have no definite opinion. The most likely situation is that we eventually end with a calculus in the style of the unified logic **LU** (Girard 1993) in which affine maintenance of context is declared, and for which affine application would appear as a subtype of the usual intuitionistic implication.
See: **Coherent spaces, Contraction, Critical pair, Linear logic, Material implication, Parsimony, Substructural logics, Subtyping, Weakening, Weak logics, Xenoglossy**.

• Algebraic logic
The remark that a logical system can be seen as an algebraic structure is of some interest. It becomes dubious when this algebraic aspect becomes prominent to the point that 'a' logic becomes any schematic formal system enjoying some property, typically consistency. The algebraic viewpoint compels us to consider that logics are 'stronger' or weaker, depending on the set of their theorems. It forces one to consider completeness with respect to algebraic structures, usually isomorphic to the set of theorems, that is, *gesticulation*. In general, cut-elimination fails for the 'logics' based on pure algebraic considerations.
See: **Broccoli logics, Completeness (external), Cut- elimination, Gesticulation, Paralogics, Substructural logics**.

• Allegory
Take the gastronomic menu: one could build a full metaphorical corpus... for instance explain linear negation by means of change of viewpoint customer *vs.* restaurant. Such iterated metaphors are less and less convincing, but after all the role of a metaphor is to convey an intuition on a specific point, not to replace the theory.
See: **Joke, Metaphor, Gastronomic menu, Prisoners, Sokal**.

• Analysis and synthesis
Analysis is easy; synthesis is difficult. The example of a good analysis is the electrolysis of water, since water can be synthesised from its constituents; but what are we to think of the chemical analysis of a human brain? In other words, one analyses in view of a possible synthesis. For instance, if you want to understand proofs in an interactive way, you may design various games, each of them being a possible analysis of what is a proof... But the synthesis may be impossible or at least lead to artificial reconstructions. In the case of ludics, a first analysis was made by means of the idea of polarity: this yielded a first synthesis (Girard 2000). This first synthesis in turn helped in the production of a more perspicuous analysis, with a more satisfactory synthesis.
See: **Frankenstein, Lorenzen, Ludics**.

• Answer
These are not quite needed, nay wanted. However, a question without any partial answer would not be taken seriously: answers validate questions. By the way, nobody was ever directly interested in Fermat's last theorem, a simple technical question without the slightest application. But the solution induced the creation of various fields such

as algebraic geometry. Hilbert's program, although wrong, was a good question, with 'answers' like sequent calculus, which do not quite contribute to the program (how could they?) but are by-products of the program. A typical good question is full completeness, but again for its side effects – here the invention of ludics. And of course, ludics answers in its way the original question, but only in its way.
See: **Astrology, Explicit, Fermat, Full completeness, Hilbert, Kepler, Laplace, Question**.

• ANTIPHRASIS
This is a form of irony, at work in 'popular democracy'. It is very common in logic, think of 'non-monotonic logic'... Most uses of the word 'semantics' (which is, after all, supposed to explain) are plain examples of antiphrasis.
See: **Control, Herbrand model, Non-monotonic logics, Operational semantics, Oxymoron, Pleonasm, Semantics**.

• ARISTOTLE
It does not appear that Aristotle was conscious of the syntax/semantics distinction. By the way, it does not seem that the distinction can be of any use to syllogistics.
See: **Logic, Scholastics, Syllogism**.

• ARMAGEDDON
The worst sort of formalists work in so-called *artificial intelligence*. These people believe that something is true as long it has not been disproved, which is one of the extreme readings of Popper. Indeed they believe that every truth is eventually bound to be refuted. Their favourite targets are Cantor's diagonal, and, needless to say, the first incompleteness theorem.

These people are like the notorious Capitaine Némo, the bitter hero of Jules Verne's 20000 *lieues sous les mers*, whose target is the extermination of mankind. Our cybermorons more modestly try to destroy mathematics, and they were particularly active in the year 2000. But you can sleep peacefully, since they do not even know basic mathematics.
See: **Artificial Intelligence, Cantor's diagonal, Falsifiable, Fermat, Gödel's incompleteness, Inconsistency proof, Non-monotonic logic, Objects and properties, Pauperism, Unfalsifiable**.

• ARSÈNE LUPIN
Maurice Leblanc makes fun of Conan Doyle in *L'Aiguille Creuse*. His episodic character Beautrelet is the exact opposite of *Herlock Sholmes*.

- *(Filleul :) Il s'agit bien de réfléchir ! Il faut voir d'abord. Il faut étudier les faits, chercher les indices, établir les points de repère. C'est après que, par la réflexion, on coordonne tout cela et qu'on découvre la vérité.*
- *(Beautrelet :) Oui je sais... c'est la méthode usuelle... la bonne sans doute. Moi j'en ai une autre... je réfléchis d'abord, je tâche avant tout de trouver l'idée générale de l'affaire, si je peux m'exprimer ainsi. Puis j'imagine une hypothèse raisonnable, logique, en accord avec cette idée générale. Et c'est après, seulement, que j'examine si les faits veulent bien s'adapter à mon hypothèse.*

See: **Abduction, Sherlock Holmes**.

• Artificial Intelligence

To the best of my knowledge, this is the only scientific area with an intrinsic conflict of interest: unlike medical researchers who are usually in good health, these AI guys badly need the stuff they are after.

See: **Armageddon, Computer science, Cordwainer Smith, Do-it-yourself, Formal, Intelligence, Kepler, Numerology, Paralogics, Question, Sense of rules**.

• Artificiality

'So, you are doing logic... you must be warped...' The popular opinion about logic (here, my hairdresser, March 2000) seems to be shared by a good half of logicians: if something is simple and natural, it is fishy. You need 10-tuples at least for the most basic definition, and never write an equality if you can replace it with an isomorphism; to cut a long story short, something is logical when you do not understand it. This cult of artificiality culminates in bleak competitions of the form: 'My logic is terrible... Sorry, mine is definitely worse!'.

See: **Bergen, Broccoli logics, Coding, Gödel's incompleteness, Lewis Carroll, Naturality, Obfuscation, Perishable, Scott domains, Xenoglossy**.

• Associativity

The focalisation property of Andreoli basically says that we can perform in a single step a cluster of positive operations. For instance, one can write ternary rules for the 'connective' $A \otimes (B \otimes C)$, which happen to be strictly identical to the ternary rules of the connective $(A \otimes B) \otimes C$: in this way one proves associativity of the tensor product. Also, the same focalisation argument would prove the distributivity of $\otimes$ over $\oplus$ *etc*. In general, connectives of the same polarity associate, that is, they enjoy the expected socialisation properties. Nothing of the like happens between connectives of different polarities. By the way, polarity is distinguished in linear logic by different graphical styles (algebraic for positive and logical for negative), and this long before the pregnancy of polarity was disclosed... However, the graphical style was devised as a mnemonic for remarkable isomorphisms (such as distributivity), which precisely occur inside the same polarity.

There is a weaker form of associativity, namely that $+-$ can be replaced (in an irreversible way) by $-+$. Typical examples are

$$(A \otimes (B \parr C)) \multimap ((A \otimes B) \parr C) \tag{193}$$

$$(A \otimes (B \& C)) \multimap ((A \otimes B) \& (A \otimes C)) \tag{194}$$

The negative operation can be delayed. This is why we can, independently of the recent discoveries on prenex forms, expand the scope of universal quantifiers: $((\forall d A_d) \,\Box\, B) \multimap \forall d(A_d \,\Box\, B)$ when $\Box$ is positive. When $\Box$ is negative, this holds too, but this is an equivalence.

Since logic is naturally associative, and connectives are about socialisation of designs, it is impossible to build a reasonable non-associative logic.

See: **Adjunction, Church–Rosser, Closure principle, Commutativity, Distributivity, Focalisation, Non-associative logic, Polarity, Prenex form, Separation, Tensor product**.

• Astrology

A very good question, with astronomy as a by-product. We have to remember that Kepler was an astrologer. After Kepler, precisely, astrology is no longer a good question. The same was true of Hilbert's Program after Gentzen.

See: **Answer, Gentzen, Hilbert, Kepler, Question, Nostradamus, Sokal**.

• Atomic proposition

Logic easily stumbles on the explanation of logical atoms. But what is a logical atom, except a symbol for the unknown formula, or, better, the *unknown* positive behaviour. This means that atoms are bound to be quantified, universally in the usual syntax, and existentially in the usual semantics. Typically, the Fax implements $X \vdash X$ for any 'value' $\mathbf{X}$ of the unknown X.

Atoms may be problematic when used in focusing syntaxes, typically, if X is positive, the rule for $X \otimes X$ should 'go further' and decompose X, which is unknown. The technical answer (see Subsection 10.2.3, p. 379) consists of formally delaying focalisation, up to the identity axiom.

See: **Eta-expansion, Focalisation, Occurrence, Polarity, Stoup, Twins, Variables**.

• Atomic weapon

If ludics were a plain game semantics, then the first player would play the winning design $\mathfrak{One}$, with no possible reply.

See: **Behaviour, Consensus, Dissensus, Game semantics, Ludics, One, Referee, Strategy**.

• $*$-Autonomous category[†]

Barr in his monograph Barr (1979) introduced $*$-*autonomous categories* as a common setting for familiar (mainly topological) duality theories. $*$-autonomous categories turn out to precisely model (the proofs of) multiplicative linear logic, in the same way that cartesian closed categories model intuitionistic $\wedge, \Rightarrow, \top$ logic. Similarly, $*$-autonomous categories with products model **MALL**, the multiplicative-additive fragment, while certain comonadic structures are used for exponentials (Barr 1991). In the Appendix of Barr's original text, Barr's student Chu, following a suggestion of Barr, gave a formal categorical construction, which, starting from a finitely complete symmetric monoidal closed category, constructs a $*$-autonomous 'completion' by formally adjoining duals. Such categories, known as *Chu spaces*, have been shown to include many interesting models of linear logic.

Using delocations, behaviours form a $*$-autonomous category – at least they do if we remove the tensor unit.

See: **Categorical semantics, Category, One**.

• Barbichette

A game that you play with small children. The ideal would be not to laugh, but it is changed into 'not to be the first to laugh'.

Je te tiens
Tu me tiens
Par la barbichette
Le premier qui rira
Aura un' tapette.

[†] By Phil Scott.

https://doi.org/10.1017/S096012950100336X

Winning conditions are about the respect for some (usually inaccessible) ideal. The point is not to respect the ideal, but to put the blame on the other. This is why the *fax*, which basically imitates the moves of `Opponent` is winning: whatever mistake he makes, the other has made it before.
See: **Fax, Interactivity, Obstination, Parsimony, Uniformity, Winning**.

• Behaviour

A behaviour is a sort of abstract formula, independent of any logical system, any syntax, any semantics. The letter $\mathbf{G}$ used for behaviours betrays their game-theoretic origin. However, a behaviour is not a game, since the rule of the game is given by *consensus*, that is, by orthogonality. The 'rule' of $\mathbf{G}$ is given by $\mathbf{G}^{\perp}$, and the 'rule' of $\mathbf{G}^{\perp}$ is given by $\mathbf{G}$: no external Tarskian referee is allowed. A behaviour is a set of designs equal to its biorthogonal, period.

By the way, life can be viewed as a game with no rule. An equilibrium is reached between *Me* and the outer *World*; this is the output of a complex process, deeply rooted in personal history. A shy person has a bigger orthogonal than an aggressive one, but, after all, the fact that we are not currently killing our enemies does not result from an absolute interdiction: we do not dare, or rather we think that this is bad taste, or that there are too many of them… In civil wars, when one side starts changing the rule, the other side soon does likewise: if I start to behave differently, that is, if I change the rule, my rule-as-orthogonal, that is, the outer world starts to behave differently.
See: **Atomic weapon, Bihaviour, Completeness (external and internal), Consensus, Design, Dissensus, Ethics, Formula, Game semantics, Objects and properties, Orthogonality, Referee, Semantics, Soundness, Strategy, Syntax, Truth, Type**.

• Bergen

At the beginning of the last century, in the rainy Norwegian town of Bergen, a horse would whinny in the presence of a man without an umbrella. Similarly, why write an equality when you can write a canonical isomorphism instead?
See: **Artificiality, Category, Isomorphism, Locative product, Obfuscation, Strict**.

• Bias

This originally arose as a way to distinguish between immediate subformulas (modulo focalisation). A *locus* is just a sequence of biases.
See: **Delocation, Directory, Locus, Ramification, Reservoir**.

• Biethics

Biethics is the uniform version of ethics.
See: **Bihaviour, Ethics**.

• Bihaviour

A bihaviour consists of a behaviour together with a partial equivalence $\cong$ on its *partial* designs. The partial equivalence and the orthogonal partial equivalence are related via the Equation (119):

$$\mathfrak{D} \equiv_{\mathbf{G}} \mathfrak{D}' \Leftrightarrow |\mathfrak{D}|_{\mathbf{G}} = |\mathfrak{D}'|_{\mathbf{G}}$$

Any behaviour $\mathbf{G}$ can be seen as a bihaviour, equipped with the PER $\equiv_{\mathbf{G}}$:

$$\mathfrak{D} \equiv_{\mathbf{G}} \mathfrak{E} \Leftrightarrow |\mathfrak{D}|_{\mathbf{G}} = |\mathfrak{E}|_{\mathbf{G}} \tag{195}$$

Bihaviours are essential in full completeness, especially in the Polymorphic Lemma (Theorem 31, p. 388), which establishes the ground case of completeness.

See: **Behaviour, Biethics, Fax, First-order quantifier, Formula, Partial design, PER-model, Pull-back, Uniformity**.

• Black Mass

Jurassic logic keeps celebrating the wedding of Semantics and Syntax through the intercession of the Holy Meta, and, of course, every ritual deserves its own mockery, the Cross upside down, the Gospel in reverse order... In paralogics, there is no syntax, no semantics – or, better, the semantics is called syntax and *vice versa*. These mockeries, religious or logical, betray a paradoxical respect for a tradition that the protagonists never understood.

See: **Abduction, Herbrand model, Jurassic Park, Non-monotonic logics, Numerology, Proof** *vs.* **models, Trinity**.

• Böhm tree

A Böhm tree is the infinite η-expansion of the normal form of a λ-term, see, for example, Barendregt (1984). The symbol Ω is used when the expansion gets stalled (that is, one reaches a non-solvable subterm). Böhm trees are to my knowledge the closest prefiguration of designs, and Böhm's theorem is a prefiguration of the separation theorem. It is fair to say that designs are Böhm trees plus additives and symmetry.

See: **Design, η-expansion, Faith, Separation, Solvable**.

• Boots

$\mathfrak{Boots}$ is a negative design, indeed the only material design in the behaviour $\perp$, which is the neutral element of the four multiplicative disjunctions. There has always been a feeling that 'boots leak', since the proof-net technology has serious problems with this constant, see Girard (1996): the formula $\perp$ must be 'physically' attached to the proof-net, contradicting its alleged neutrality. But this can now be explained: the problem only occurs in situations when $\mathfrak{Boots}$ is tensorised, typically in $\perp \otimes \perp$. But then we are not dealing with the negative behaviour $\perp$ but rather with its shift $\downarrow\perp$, which is no longer a neutral element for the cotensor ⅋, and it is normal to tie this pseudo-neutral... so boots do not quite leak!

See: **Dualiser, Leakage, One, Tensor product, Xenoglossy**.

• Broccoli logics

Not as bad as paralogics, Broccoli logics are deductive. The basic idea is to find a logical operation or principle not yet considered... which is not too difficult: call it *Broccoli*. Then the Tarskian machinery works (here the symbol '✣' stands for the syntactical Broccoli):

A ✣ B is true if A is true **Broccoli** *B is true.*

If you are smart enough to catch this delicate point, *Broccoli* is the meta of '✣'. Broccoli is equipped with principles that have been never yet considered, typically

$$(A \;✣\; B) \Rightarrow (A \;✣\; (B \;✣\; B)) \tag{196}$$

and soundness and completeness are proved with respect to all structures containing a constructor ♡ enjoying

$$(a \heartsuit b) \leqslant (a \heartsuit (b \heartsuit b)) \tag{197}$$

(Hint: to prove completeness, construct the *free Broccolo*.)

By the way, Tarski is not (fully) responsible for this abusive extension of his paradigm, which was originally restricted to classical logic (for which one can say that the paradigm is not wrong, if not very useful), with the idea of one solid reality. The Broccoli logician, does not hesitate to tamper with 'reality', believing that his logic is just classical logic in a different universe, admittedly weird, with the idea that, from the inside of the Broccoli universe, Broccoli logic is just like classical logic. The reality has changed, and us, as part of reality, were unable to notice it! But this is wrong; Broccoli logics rarely enjoy cut-elimination and in spite of their relation to a hypothetical meta-Broccolo, the reflection schema fails... and the reflection schema is the possibility of a formal reasoning on truth within the limits ascribed by Gödel's theorem. In other words, these 'logics' are incompatible with their own 'meta'.

See: **Algebraic logic, Artificiality, Do-it-yourself, Gesticulation, Logical relation, Meta, Non-associative logic, Non-commutative logic, Paralogics, Paraphrases, Phase semantics, Reflection schema, Relevance logics, Semantics, Soundness, Substructural logics, Tarskian semantics**.

• Brouwer

For Brouwer, logical operations such as disjunction and existence should commute with provability. Brouwer's motivation was basically subjectivistic (the 'creative subject' is the only reference). A less controversial justification is the common-sense remark that eventually proofs only interact with proofs: therefore only the (implicit) contents of proofs matters.

Of course this supposes some commutation of the other logical connectives with provability; conjunction commutes with provability already in the classical case... but negation and implication hardly do, and the attempt by paralogics at such a commutation was a total failure, for want of any possible formalisation. However, these paralogicians were illiterate beyond any decent joke: real mathematics is not based on complementation (yielding 'not to prove'), but on orthogonality (yielding 'to prove the orthogonal').

A much subtler form of commutation (Heyting's semantics of proofs) was devised; it involves functions and an important change of viewpoint – the focus is no longer on the formula but on its proof.

See: **Constructivism, Creative subject, Disjunction property, Existence property, Heyting, Intuitionism, Ludics, Lorenzen, Non-monotonic logics, Orthogonality, Saaty volume, Tarskian semantics**.

• Bureaucracy

This is another name for formalism. Many people still believe that writing rules in a pedantic way, transforming them like a machine (but less efficiently) is the essence of logic, at least of proof-theory. It is true that proof-theory can be read (and must be read*able*) in this way, but this is not the essence, just a phenomenon. Obviously, certain people enjoy writing formal proofs, formal codes, Gödel numbers, as if something might come out of it, a sort of aesthetics or cabalistics. An aging formalist can switch to numerology.

See: **Coding, Completeness (external and internal), Formal, Formula, Lorenzen, Numerology, Od-x, Syntax**.

• Cantor's diagonal

Vaguely reminiscent of the Liar's paradox 'I am lying', Cantor's paradox is the bizarre statement that you cannot enumerate real numbers: given a denumerable set of functions f_n, the function $n \rightsquigarrow f_n(n) + 1$ defines a function g distinct from all f_n. The diagonal method has been recycled in Gödel's theorem, or the undecidability of the halting problem. It can also be used positively to prove compactness of certain weak topologies. From time to time it is refuted by a paralogician.

See: **Armageddon, Gödel, Gödel's incompleteness, Incompleteness, Halting problem, Inconsistency proof**.

• Categorical completeness[†]

Several full completeness theorems have been produced, usually for the multiplicative fragment of linear logic; even in that limited case, it is difficult to avoid 'leakage' (for example, the addition of the Mix rule).

Blute and Scott (Blute and Scott 1996; Blute and Scott 1998) used the representation theory of groups and Hopf algebras to give full completeness theorems for multiplicative linear logic (MLL + Mix) and Yetter's cyclic linear logic (Cyll + Mix). They interpret proofs as dinatural transformations between multivariant functors over a category of topological vector spaces (in the case of cyclic linear logic, the dinaturals must be equivariant with respect to the continuous action of a certain noncocommutative Hopf algebra). The main theorems have the form: *the dinatural transformations $Dinat(A, B)$ form a vector space, with basis the cut-free proofs of $A \vdash B$*. Recently, Hamano (Hamano 2000) extended these methods to prove full completeness for MLL without Mix, using a $*$-autonomous category of topological abelian groups together with Pontrjagin duality.

See: **Category, Completeness (external), Full completeness, Leakage, Mix, Perishable**.

• Categorical semantics

Denotational semantics always yields categorical semantics, that is, some concrete category in which logic can be interpreted (formulas as objects, proofs of implication as morphisms). In my experience, the concrete models (Scott domains, coherent spaces, hypercoherences *etc.*) are more interesting than the abstract categorical nonsense interpretation. Typically, interpret intuitionistic logic in arbitrary **CCC** (closed Cartesian category); I am afraid that you get little more than a paraphrase of intuitionistic logic. But now take the concrete **CCC** of coherent spaces (and stable maps): intuitionistic logic is now part of a new logic, linear logic.

See: **$*$-Autonomous category, Category, Coding, Coherent space, Denotational semantics, Form** *vs.* **contents, Hypercoherence, Läuchli semantics, Linear logic, Paraphrases, Reflection schema, Scott domains, Self-interpreter**.

[†] By Phil Scott.

• Category

Category-theory played an important role in the disclosure of the deep structure of logic: for instance, coherent spaces form a categorical model for linear logic, and, by the way, linear logic came from this model, not the other way around. The pregnancy of categories in our area made me style the period 1970–2000 as the *time of categories*, a period that opened with the Curry–Howard isomorphism. Ludics originates in this category-theoretic approach, but eventually took some distance from it.

The limitations of categories, insofar as we can judge them solely from the logical viewpoint, lies in their spiritualism, their extreme spiritualism: *everything is up to isomorphism.* In particular, categories cannot explain locative logical constructions such as intersection types – or, if you prefer, the categorical viewpoint compelled us to consider these artifacts as non-logical. In the same way, category-theory cannot explain the prenex forms of ludics, which are based on *equalities*, and which are definitely impossible to explain by means of isomorphisms.

To sum up, category theory only presents a limited aspect of logic; provided we realise this, it remains a very important tool.

See: **$*$-Autonomous category, Bergen, Categorical semantics, Coherent space, Curry–Howard, Illusions, Intersection type, Isomorphism, Linear logic, Ludics, Prenex forms, Pull-back, Savoir-vivre, Spiritual logic, Spiritualism, Strictness**.

• Chronicle

A chronicle is basically a finite branch in a design, seen as a sequence of actions, so that a design-dessein is the set of its chronicles. One can also see a chronicle as a finite branch in a proof-tree. They are close in spirit to *views*.

See: **Action, Dessein, Dispute, View**.

• Church–Rosser

The Church–Rosser property (Church and Rosser 1936), which was originally a property of pure λ-calculus (Barendregt 1984), states the unicity of normal forms. The property holds for natural deduction, proof-nets, but not for sequent calculus; moreover, the property is problematic in the classical case. Broccoli logicians think that Church–Rosser is just a matter of making normalisation deterministic by artificial destruction of *critical pairs*. But if we only allow, say, leftmost reduction, then $f(a)$ could normalise to b and $g(b)$ to c, whereas $g(f(a))$ could normalise to $c' \neq c$. In other words, Church–Rosser is in the final analysis about associativity.

In ludics, normalisation is Church–Rosser in the narrow sense (a deterministic normalisation), but also in the wider associative sense, expressed by the *closure principle*.

See: **Associativity, Broccoli logics, Classical logic, Closure principle, Critical pair, Cut-net, Natural deduction, Normalisation, Proof-nets, Sequent calculus**.

• Classical logic

Classical logic is the logic of reality. This explains its pregnancy – it is so difficult to depart from realism – and also its limitations. Technically speaking, classical logic does not enjoy Church–Rosser, for deep reasons that we do not quite understand, but here are a few hypotheses:

https://doi.org/10.1017/S096012950100336X

Non-determinism: Classical logic could be naturally non-deterministic, but then only an interpretation like 'quantum mechanics' could match the hypothesis.

Polarisation: Classical logic could be naturally deterministic, provided one restores a 'hidden variable', namely polarity. In fact, the system **LC** of Girard (1991) and its subsystem – the $\lambda\mu$-calculus of Parigot (Parigot 1992) – are deterministic. This determinism is obtained through a rationalisation of the Gödel '$\neg\neg$-interpretation', which becomes associative when polarities – in a sense close to ludics – have been associated to classical formulas.

Forget it: Classical logic could be just a *comment* on some part of a proof with no algorithmic interest. Such a viewpoint could be implemented in ludics by the systematic use of plain unions and intersections, just with a shift when the polarity has to change, and no delocation.

See: **Classical model, Church–Rosser, Consistency, Double negation, Exponentials, Polarity, Procedural logic, Realism, Shift, Weak logics**.

• Classical model

The educated proof of Gödel's completeness theorem consists of attempting to make a cut-free proof of the formula A, see, for example, Girard (1987b), in a systematic way. If this fails, the attempted proof contains an infinite branch that induces a classical model. The branch induces a design, provided the subformulas of A have been reasonably located. However, several designs may correspond to the same classical model: if $B \wedge C$ is false in the model, the design must put the blame on B or C, in an exclusive way.

Classical completeness, although external, has an immense *technical* value, which has no analogue of the same quality for other logical systems.

See: **Classical logic, Completeness, Gesticulation, Gödel, Herbrand model, Kripke model, Phase semantics**.

• Closed-world assumption

This is an illiterate interpretation of logic programming: 'If I cannot prove A, then A is false'. It corresponds to $\Omega = \maltese$ and stumbles on the halting problem, like all similar ideas. This nonsense gave rise to unbelievable paralogical developments, including transfinite 'proofs'... '`PROLOG` is not stalled, just wait a minute!' Apparently some people believe that a mistake – here the overlooking of the halting problem – can be corrected by a transfinite iteration.

The paradigm of negation as failure is more reasonable, although still incorrect.

See: **Halting problem, Logic programming, Negation as failure, Non-monotonic logics, Paralogics, Prisoners, Proof-search**.

• Closure principle

The closure principle basically says that normalisation can be reduced to the closed case, that is, that it is enough to consider closed nets. In fact $[\![\mathfrak{D}, \mathfrak{E}]\!]$ is the unique design such that $[\![[\![\mathfrak{D}, \mathfrak{E}]\!], \mathfrak{R}]\!] = [\![\mathfrak{D}, \mathfrak{E}, \mathfrak{R}]\!]$ for any $\mathfrak{R}$ 'closing' the system. The principle is the combination of associativity and separation. The closure principle is reminiscent of the familiar equation

$$\langle u * (x) \mid y \rangle = \langle x \mid u(y) \rangle$$

which defines the adjoint of an operator. In terms of such equations, ludics is better behaved than operator algebra...

See: **Adjunction, Associativity, Church–Rosser, Composition of strategies, Separation**.

• Coding

Coding is necessary in logic: it is a convenient way to reduce to only one denumerable data type, natural numbers. Moreover, delocation, which is not a convenient or superficial operation, makes use of coding. Coding is used in Gödel's theorem so as to get the incompleteness of arithmetic. However, if we replace arithmetic with a variant containing primitives for syntactical operations, no coding is needed to get the theorem, which still conveys the full methodological meaning of incompleteness.

Coding is often used in logic to hide the actual construction behind the neutrality of numbers. A typical example is the paper of Gentzen (Gentzen 1969d) proving the consistency of the simple theory of types, which amounts to constructing a finite model for types $0, 1, \ldots, n$. Gentzen assigns cabalistic numbers to proofs, but he is just evaluating a classical formula in the model $\{a\}, 2^{\{a\}}, 2^{2^{\{a\}}}, \ldots$ and showing that logical rules preserve truth... Even the great Gentzen could indulge in numerology.

Without minimising the practical interest of codings, one can observe that the practice of artificial codings corresponds to a logical ideology for which the language is a mere bureaucratic device, with no intrinsic properties. The passage from the *time of codings* 1930–1970 to the *time of categories* 1970–2000 is synthesised by the Curry–Howard isomorphism: before one would explain computable functions by means of codings *à la* Kleene (Kleene 1952), later one deals with categorical models, for example, Scott domains. By the way, Scott domains were still using a lot of codings, due to the absence of *minimum* data; their replacement with coherent spaces is the definite rupture with codings.

See: **Artificiality, Bureaucracy, Categorical semantics, Coherent space, Curry–Howard, Notations, Gödel's incompleteness, Incompleteness, Natural deduction, Numerology, Obfuscation, Od-x, Scott domain**.

• Coherent space

Coherent spaces were introduced in Girard (1987a) as a drastic simplification of Scott domains, exploiting the stability condition of Berry (Berry 1978): the overcodings at work in Scott domains were eliminated in favour of a structure that was much smaller in size, and conceptually much simpler. The first consequence was to individuate a decomposition of the intuitionistic arrow as $!A \multimap B$ (which was in fact found using the archaic version of coherent spaces, qualitative domains (Girard 1986)). The second consequence was linear negation, by far the most important discovery of linear logic, and the compulsory door to interactivity.

Coherent spaces are defined up to isomorphism: they actually form one of the best-behaved categories, with all sorts of limits and an involution. Designs can be seen as *cliques* in an absolute coherent space, built from disputes, but they can hardly be reduced to this sole aspect. Nevertheless, coherent spaces remain one of the main intuitions concerning logic, so fall in the respectable category of *treason*. Among beautiful treasons in this line, we should mention the work of Ehrhard on *hypercoherences*, (Ehrhard 1995).

See: **Affine logic, Categorical semantics, Coding, Dispute, Gustave function, Hypercoherence, Linear logic, Linear negation, Parallel or, Pull-back, Scott domain, Stability, Treason**.

• Commutativity

Commutativity is not an essential property, like associativity. Commutative connectives are simpler to handle, non-commutative connectives are in principle more expressive. In ludics, commutativity is *strict*, for example, the tensor product really enjoys $\mathbf{G} \otimes \mathbf{H} = \mathbf{H} \otimes \mathbf{G}$ – in particular, application is independent of the order of arguments: $((\mathfrak{F})\mathfrak{A})\mathfrak{B} = ((\mathfrak{F})\mathfrak{B})\mathfrak{A}$. *Real* application is sensitive to the order of arguments because it combines the commutative application with delocations, for example, $((\mathfrak{F})\varphi(\mathfrak{A}))\psi(\mathfrak{B})$ vs. $((\mathfrak{F})\varphi(\mathfrak{B}))\psi(\mathfrak{A})$.

See: **Adjunction, Associativity, Non-commutative logic, Strictness**.

• Completeness: external version

Completeness originally meant that nothing is missing. The expression has now the slightly different meaning of the appropriateness of a syntax to a semantics: provability misses no valid formula. Technically speaking validity is about the truth of, say, a proposition A in all models; if we quantify over all propositional variables of A (in order to get a closed second-order proposition $B = \forall XA$), the completeness statement is simply that for all *true* closed B, B is provable. These closed formulas are of a restricted form (second-order quantifiers are universal) let us call them $\mathbf{\Pi}^1$. Gödel's incompleteness theorem basically states the failure of his own result for $\mathbf{\Sigma}^1$ formulas... 'Voi che uscite da questa classe ($\mathbf{\Pi}^1$) lasciate ogni speranza.'

Completeness can be stated for proofs: if something in a semantics-of-proofs such as ludics is semantically accepted (for us the something is a winning design in the associated behaviour), then it must arise from a syntactical proof: this is called full completeness. Before pushing the discussion further, let us observe that this formulation admits a forgetful version, namely that truth implies provability: the enhanced form is *a priori* limited to $\mathbf{\Pi}^1$ formulas.

Theoretically speaking, external completeness is hardly more than a good question, since it has no interesting corollary. But the by-products are immense, here the creation of ludics, and the discovery of the *losers*.

Practically speaking, it is of interest to get a finite syntax for ludics. In this respect completeness issues are not that stupid, since a complete syntax is... complete, that is, nothing is missing. But this is a practical justification, not the recognition of the pregnancy of the syntax/semantics schizophrenia.

See: **Algebraic logic, Behaviour, Categorical completeness, Completeness (internal), Consistency, Full completeness, Gödel, Gödel's incompleteness, Harmony, Incompleteness, Linear logic, Loser, Question, Referee, Σ and Π formulas, Schizophrenia, Semantics, Sequent calculus, Soundness, Syntax, Trinity**.

• Completeness: internal version

It is of utmost interest to remark that the class of formulas for which completeness (full or not) works is the same as the class of formulas enjoying the subformula property. But what is the meaning of the subformula property? It asserts that the set of all cut-free proofs (by the way, I forgot, full completeness is restricted to cut-free proofs) of a $\mathbf{\Pi}^1$ formula is independent of the (usually higher order) logic in which it takes place. So to speak, all proofs are already there, nothing is missing... and that is the right intuition. For the first time we see the tail of an internal approach to completeness. The problem is that this internal closure takes place inside syntax, and there is no mathematical entity such as syntax, only bureaucratic systems. Formulating completeness internally (but without syntax, as the essence of cut-elimination) was therefore *the* task.

This has been successful beyond all expectations, for the meaning of logic is much better understood now. In ludics completeness takes the form of the removal of the biorthogonal: if $\mathbf{E}$ is an ethics (we may think of those designs generated by a cut-free sequent calculus), then $\mathbf{E}^{\perp}$ plays the role of (counter-) models, and the set of designs validated by the models is rendered by the biorthogonal $\mathbf{E}^{\perp\perp}$, hence completeness is just $\mathbf{E} = \mathbf{E}^{\perp\perp}$ (up to incarnation).

All spiritual connectives (but second-order existence) have their own form of internal completeness: for instance the disjunction property expresses the internal completeness of $\oplus$. External completeness is a corollary of internal completeness.

See: **Admissible rule, Behaviour, Ethics, Bureaucracy, Completeness (external), Cut-elimination, Disjunction property, Ethics, Incarnation, Paraphrases, Saaty volume, Σ and Π formulas, Spiritualism, Subformula property, Syntax, Takeuti's Conjecture**.

• Composition of strategies

The basic form of composition of strategies is to make two players play against each other. If these players are involved in several games at the same time, this induces a compound strategy, but for the remaining games only. This is well expressed by the cut-rule

$$\frac{\vdash \Gamma, A \quad \ldots \quad \vdash A^{\perp}, \Delta}{\vdash \Gamma, \Delta}$$

Here the two players decided to forget the game $A/A^{\perp}$.

The composition is cruelly missing in the archaic works of the Lorenzen school.

Composition is the central notion and is represented in ludics by normalisation. The closure principle gives a synthetic definition of composition.

See: **Closure principle, Cut-elimination, Cut-net, Cut-rule, Game semantics, Lorenzen, Maul, Normalisation, Sequential algorithm**.

• Computer science

Although man-made, computer science is the physics of logicians. In the beginning:

— Many computer scientists failed to realise the unfeasibility of the halting problem... with, as a result, the building of various paralogics.

— More educated people were still paying too much attention to formal issues, with an excessive, and surrealistic, emphasis on consistency, and the building of too many Broccoli logics.

In general, computers prompted a renewal of positivistic nonsense, artificial intelligence and so on. But how unfair it would be to reduce computer science to these archaic mistakes. It is an immense source of intuitions, such as non-determinism, locations, proof-

search, streams, process algebras... not to mention the mere idea of interactivity. Without computer science, would there still be any room left for logic?
See: **Artifical intelligence, Broccoli logics, Consistency, Explicitation, Halting problem, Interactivity, Locus, Logic, Operational semantics, Paralogics, Process algebra, Proof-search, Stream**.

• CONNECTIVE
This is basically any type of socialisation of behaviours: human beings belong to clubs, political parties, sects, *etc*., logical artifacts use *connectives*, some very useful like implication, some completely artificial like the notorious Sheffer's stroke. It is not our point to judge what is good or bad socialisation: it would be like deciding what is a good or bad mathematical definition. However, the most immediate connectives have been investigated in this monograph.

Usually we have three layers for the same idea of a connective. First, a *strictly* associative version, then a restricted form of the same (that is, a *partial* connective) enjoying completeness, and finally a delocated version of the second form, still enjoying completeness, and yielding a *total* spiritual connective, which is no longer *strictly* associative.
See: **Locative logic, Shift, Spiritual logic, Strictness**.

• CONSENSUS
Ludics corresponds to a notion of game by consensus. This means that the rule of the game is part of the game. When the normalisation diverges, there is a sort of *draw*, hence one is forced to play in a certain way to avoid divergence. If I want `Opponent` to follow the rule, I usually play a *losing* design to force the consensus.
See: **Atomic weapon, Behaviour, Convergence, Dissensus, Divergence, Dog, Game semantics, Loser, Orthogonality, Referee, Strategy, Test, Winning**.

• CONSISTENCY
The essential property of a formalism, according to Hilbert. The completeness theorem shows that any consistent classical theory has a model and consistency is therefore a way to prove the existence of infinite objects without committing one to dubious set-theoretic constructions... At least this was Hilbert's original motivation. Unfortunately, there is almost no example of a proof of existence coming from consistency: in real life consistency is obtained from the model, not the other way around. The commonsense limitation of consistency is that most possible worlds are of no interest, in other words, consistency is presumably not the ultimate internal property.

The limitations of consistency become obvious as soon as we step out of classical logic. Typically intuitionistic logic guarantees that there is an (implicit) contents in a proof, typically a proof of a disjunction $A \vee B$ can be transformed into either a proof of A or a proof of B. But what about the intuitionistic system in which we have added the axiom schema $A \vee \neg A$? This intuitionistic system (which is identical to classical logic) should be declared 'inconsistent' since it asserts nonsense, properties that it cannot justify... but unfortunately no *formal* contradiction can be found. However, we can observe that something can be said: as an intuitionistic system, classical logic has no cut-elimination, that is, there is no way to prove $A \vee \neg A$ in a cut-free system using intuitionistic sequents. This suggests that cut-elimination (which is, by the way, well known to enforce consistency) is much more central than consistency.

Not to be unjust to Hilbert, we should admit that he was seeking a sort of *immanence* out of formal considerations on proofs. And surely consistency is one of the desirable properties of a logical system, but a rather obscure one, like the existence of brakes is one of the desirable properties of a car, but by no means the central one, which is perhaps that the engine works. Consistency eventually looks like the 'poor man's immanence'.

See: **Classical logic, Completeness (external), Computer science, Cut-elimination, Disjunction property, Fundamentalism, Jurassic Park, Obstination, Syntax, Truth**.

• Consistency proof

The second incompleteness theorem forbids the existence of any convincing consistency proof. However, the desire for such results was so great that logicians, especially in Germany, kept on seeking consistency proofs; after all certain sects make money by selling insurances against the explosion of the Earth... All this eventually ended with the construction of larger and larger *Ordinalzahlen*, with very few outputs, but ideological. An exception is to be made with Gentzen who produced actual ideas, like sequent calculus (Gentzen 1969a) and ordinal analysis (Gentzen 1969b).

See: **Foundations, Gentzen, Gödel's incompleteness, Incompleteness, Ordinal analysis, Predicativity, Sequent calculus**.

• Constructions

The Calculus of Constructions of Coquand (Coquand and Huet 1988) is a common extension of system $\mathbb{F}$ and Martin-Löf's system. Its expressive power is exploited in the, very successful, system of proof-assistance **CoQ** developed by Huet and his team.

See: **Formalisable, Martin-Löf system, System $\mathbb{F}$**.

• Constructivism

An ideological combination of (the drawbacks of) Hilbert and Brouwer: you want effective results, but you also want your methods to be pure. But nobody agrees on purity: constructivists split as easily as Trotskyists.

See: **Brouwer, Constructivity, Creative subject, Intuitionistic logic, Saaty volume**.

• Constructivity

This is not the same as constructivism: no sectarianism is involved – 'The colour of the cat does not matter provided she catches mice'. Constructivity deals with the *how*, whereas constructivism is tied to the *why*.

Kreisel was the first to give explicit contents to mathematical proofs – think of his analysis (Kreisel 1958) of a famous theorem of Littlewood: this theorem states that the difference between $\pi(x)$ (the number of prime numbers less than x) and its integral approximation known as $Li(x)$, oscillates, that is, that the sign changes infinitely often. The proof was made in two parts, depending on the truth or falsity of the Riemann hypothesis, which made it non-effective; following the spirit of cut-elimination, Kreisel was, so to speak, able to eliminate cuts between the two parts, and eventually got a bound on the first change of sign of $\pi(x) - Li(x)$.

See: **Constructivism, Cut-elimination, Explicitation, Kreisel, How and why**.

https://doi.org/10.1017/S096012950100336X

• Contraction

The most conspicuous novelty of linear logic (not the deepest one) was the banishing of contraction

$$\frac{\vdash \Gamma, A, A}{\vdash \Gamma, A} \tag{198}$$

An involutive (and constructive) negation can only live in the absence of contraction. Intuitionistic logic banishes contraction to the right under the pretext that only one formula is allowed.

Weakening (the other dubious structural principle) is admitted in ludics, with the proviso that it is considered as losing – dog's play. Contraction is not allowed at all, for geometrical reasons: one could not make sense of normalisation and/or separation would fail.

In linear logic, contraction makes its way through the exponentials. The exponentials look like a formal way to make linear what is quadratic or worse, think of a Fock space. But this is nothing but a way to harbour non-linear features inside a linear framework. To understand the distinction: if $\mathfrak{F}\perp\mathfrak{A}$ for all $\mathfrak{A} \in \mathbf{E}$, then $\mathfrak{F}\perp\mathfrak{A}$ for all $\mathfrak{A} \in \mathbf{E}^{\perp\perp}$, but if $\mathfrak{F}\perp\varphi(\mathfrak{A}) \otimes \psi(\mathfrak{A})$ for all $\mathfrak{A} \in \mathbf{E}$, there is no way to conclude that $\mathfrak{F}\perp\varphi(\mathfrak{A}) \otimes \psi(\mathfrak{A})$ for all $\mathfrak{A} \in \mathbf{E}^{\perp\perp}$. So to speak, the set $\mathfrak{F}^{\perp}$ is a *hyperplane*, whereas $\{\varphi(\mathfrak{A}) \otimes \psi(\mathfrak{A}); \mathfrak{A} \in \mathbf{E}\}$ is a *quadratic variety* that can by no means be described as the intersection of hyperplanes.

See: **Affine logic, Dog, Double negation, Exponentials, Intuitionistic logic, Linear logic, Loser, Mix rule, Relevance logics, Structural rules, Weakening**.

• Control

In logic programming, this is a typical antiphrasis: so-called 'control instructions' are supposed to improve the search algorithm, by taking liberties with respect to logical orthodoxy. The result is a complete loss of control. . .

See: **Antiphrasis, Logic plus control**.

• Convergence

This is the good property of normalisation: (normal) termination. Convergence expresses the consensus between the players, that is, the orthogonality of their respective designs. In infinitary logics, convergence is proved by means of ordinal assignments.

See: **Consensus, Dissensus, Normalisation, Ordinal analysis, Orthogonality**.

• Copycat

Since I am cautious, I play on two boards: White against Karpov; Black against Kasparov (the Chess player). I am negative, that is, the other camp starts. When one of the two Ks has played a move, say `a2-a3`, this is exactly my answer to the other K. If any Chess play had a winner, I would be sure to win on one of my boards.

In ludics the copycat becomes the Fax, to stress the fact that the two boards are not the same. The copycat is the paragon of the good logical joke, which conveys a large part of the intuition of what is an identity axiom. . . one does not need to know Chess to be a successful copycat. What is missing in the metaphor is delocation, but think a second: the natural way to do it is the Web (in simultaneous plays in a single room, you are likely to be caught).

See: **Delocation, Fax, Joke, Metaphor**.

- Cordwainer Smith

What a surprise to discover that several distinguished logicians share my admiration for the science-fiction writer Cordwainer Smith! This guy wrote about intelligence, destiny… in a surprising way: for instance his robots have some animal part, just to make mistakes, which he understands as an essential part of intelligence… With Smith, we are very far from the AI ideology and the 'intelligent = formal' identification.

See: **Artificial Intelligence, Formal, Intelligence, Mistake**.

- Correctness criterion

Assume that we have a proof-structure Θ, that is, a would-be proof of a single formula A, without cut, and that we are given a *switching* $\mathscr{S}$ of Θ, then we can produce a 'proof' of $\vdash A^\perp$. As usual, we start with the conclusion, until we reach axioms. The formulas occurring in the sequents of our paraproof will be the negations of the formulas of Θ. The (non-deterministic) algorithm is as follows (sequents are considered up to order, that is, modulo exchange):

1 If I get a sequent $\vdash \Gamma, B \parr C$, and if $B^\perp, C^\perp$ occur in Θ, then I can apply a $\parr$-rule, with the sequent $\vdash \Gamma, B, C$ as premise.

2 If I get a sequent $\vdash \Gamma, B \otimes C$, and if $B^\perp, C^\perp$ occur in Θ as the premises of a $\parr$-link L, then I can apply a $\otimes$-rule whose premises are

— $\vdash \Gamma, B$ and $\vdash C$ if $\mathscr{S}(L) = l$

— $\vdash B$ and $\vdash \Gamma, C$ if $\mathscr{S}(L) = r$

3 Otherwise $\vdash \Gamma$ is accepted as an axiom (paralogism 'Give up')

Such a 'paraproof' can be represented as a proof-structure, which is exactly as usual, but for the fact that arbitrary axioms (that is, links with no premises with the formulas of Γ as conclusions) are used to represent the axiom $\vdash \Gamma$; except for this detail, this proof-structure is in fact a proof-net, which is uniquely determined by $\mathscr{S}$. Now, observe that

1 Cut-elimination still holds.

2 Paralogisms produce 'enough' paraproofs.

As to (1), I can perform a cut between my proof-net of A and my paraproof-net of $A^\perp$ and perform cut-elimination in this paraproof-net, up to the moment where all $\otimes$ and $\parr$-links have been eliminated. Geometrically, I end up with a connected and acyclic structure, containing only axioms and cuts. This corresponds to the necessity of the criterion. As to (2), if I consider those paraproof-nets coming from switchings, then the sufficiency of the criterion enables me to sequentialise my proof-net. The switchings should be actually seen as a *dense* subset of paraproofs.

The homogeneity between proofs and paraproofs is total, provided we add this 'Give up' to logic: slightly modified it becomes the *Daimon* of ludics.

See: **Church–Rosser, Daimon, η-expansion, Orthogonality, Paralogism, Proof-net, Syllogism, Test**.

- Creative subject

This was an attempt by Brouwer to formalise the activity of the mathematician. For true believers only…

See: **Brouwer, Constructivism, Saaty volume**.

https://doi.org/10.1017/S096012950100336X

• Critical pair

A branching in rewriting, when two rules apply. Lafont found an interesting example, linked to weakening (Girard *et al.* 1990):

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ \vdash \Gamma\end{array}}{\vdash \Gamma, A}\,\textit{Weak.} \qquad \dfrac{\begin{array}{c}\vdots\\ \vdash \Delta\end{array}}{\vdash \Delta, A^{\perp}}\,\textit{Weak.}}{\vdash \Gamma, \Delta}\,\textit{Cut} \tag{199}$$

with two possible rewritings, one coming from the left branch:

$$\dfrac{\begin{array}{c}\vdots\\ \vdash \Gamma\end{array}}{\vdash \Gamma, \Delta}\,\textit{Weak.} \tag{200}$$

and one coming from the right branch:

$$\dfrac{\begin{array}{c}\vdots\\ \vdash \Delta\end{array}}{\vdash \Gamma, \Delta}\,\textit{Weak.} \tag{201}$$

which causes a definite failure of the Church–Rosser property. In a polarised calculus, weakening is restricted to *positive formulas*, and only one among $A, A^{\perp}$ is positive: the critical pair vanishes.

See: **Affine logic, Church–Rosser, Polarity, Weakening**.

• Curry–Howard

The isomorphism between simply typed λ-calculus and natural deduction is a typical product of the late sixties, summarised by Howard's paper Howard (1980), which was widely circulated, but published much later. It must be noticed that the idea of proofs-as-functions was in the air anyway, and that the isomorphism (not a vague mutual encoding with leakage in both directions) is the central transition between the age of codings and the age of categories. Linear logic later proposed proof-as-processes.

Curry–Howard is the first non-realist approach to logic written in decent terms, for realisability and its abuse of coding was pushing too much in the direction of subjectivism, which is not a good alternative.

See: **Category, Coding, λ-calculus, Natural deduction, Heyting's semantics, Leakage, Linear logic, Natural deduction, Realisability, Realism.**

• Cut-elimination

The main achievement of Gentzen, (Gentzen 1969a). Sequent calculus is not made of formulas, but of sequents $\Gamma \vdash \Delta$, where Γ, Δ are finite sequences of formulas and the new symbol '$\vdash$' is not quite implication. The usual *Modus Ponens*, that is, 'From A and $A \Rightarrow B$ deduce B' is replaced with

$$\dfrac{\Gamma \vdash A, \Delta \quad \Gamma', A \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'} \tag{202}$$

the cut-rule, *the rule that you love to hate*. Observe that the calculus of Gentzen distinguishes the rule *Modus Ponens* from the implication $(A \wedge (A \Rightarrow B)) \Rightarrow B$. The cut-elimination theorem, which basically yields an algorithm to eliminate cut is surprising, since it relates two absolutely opposite views of logic. The deductive, or *implicit* approach is about lemmas (which contain the main ideas), which one combines together like modules in programming: a real proof basically uses the cut rule. The *explicit* approach is without cut, and in practice is never used, except by computers, since all ideas (that is, lemmas) have disappeared. But a computer – an idiot with big brains – can use cut-elimination, either to transform an implicit proof into an explicit one, or simply to look for a cut-free proof, since the absence of cut restricts proof-search to subformulas. Cut-elimination is a completeness theorem (the real one indeed!), saying that the explicit approach is as strong as the implicit one. The distinction between explicit and implicit is well explained by this computer analogy: explicit booleans are `Yes` or `No`, but implicit booleans are more interesting, they are of the form `Algorithm + Argument`, that is, of a cut. The implicit boolean is a question, the explicit boolean is the answer[†].

See: **Algebraic logic, Answer, Completeness (internal), Composition of strategies, Consistency, Constructivity, Daimon, Explicitation, Explicit, Implicit, Lemma, Lewis Carroll, Normalisation, Paraphrases, Proof-search, Question, Sequent calculus, Subformula property, Tartuffe, Truth.**

• CUT-NET

Several designs together, whose bases form a connected-acyclic graph. A cut-net is bound to be normalised. A cut is just a coincidence handle/tine between the bases of two designs. Here we stumble on an important point, which has been known since the paper Girard (1989a) on Geometry of Interaction: in the identity axiom $A \vdash A$, the two As are distinct, so that it is not a true identity axiom (or identity fax; by the way, how could we think of a fax as the identity, if the fax of real life were the identity, I would not have one at home!), but in the cut rule

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma', A \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}$$

the two As are actually the same, that is, share the same location.

Commonsense is enough to understand this point: GoI considers the axiom as an extension cord (distant) whereas cut corresponds to plugging: in order to plug you must coincide, a distant plugging would mean a hidden cord, that is, a hidden identity link. The two As of the cut therefore have a common location, and they must be of opposite polarity; only a handle/tine coincidence between the respective bases $\Upsilon \vdash \Lambda, \sigma$ and $\sigma \vdash \Pi$ of two designs matches the idea that the information (positive) contained in $\Upsilon \vdash \Lambda, \sigma$ will be transmitted to $\sigma \vdash \Pi$ through the σ: information exits through tines and enters through handles. During cut-elimination, the original cut endeavours many transformations, in other words, one cut becomes n cuts and two designs become $n+1$ designs, this is why we must describe a general situation with a finite number of cuts, cut-nets.

[†] When I say that we are not interested in answers, just note that the words 'yes' and 'no' have strictly no interest, to be more precise, these are presumably the only booleans with no meaning at all: they are cut-free, that is, pure answers, but answers to nothing.

See: **Church–Rosser, Composition of strategies, Design, Fax, Geometry of interaction, Locus, Pitchfork**.

• Cut-rule

This is the rule that expresses the transitivity of implication in sequent calculus, in fact the only really deductive rule of logic. Cut is the possibility to use and reuse lemmas. Cut-elimination is therefore a sort of miracle. It relates the implicit and the explicit through cut-elimination. The discussion in Girard (1989b) is slightly ancient, but is still valuable.
See: **Composition of strategies, Cut-elimination, Cut-net, Explicit, Implicit, Lemma, Lewis Carroll, Sequent calculus**.

• Daimon

A $\Delta\alpha\acute{\iota}\mu\omega\nu$ has all possible powers, including that of creating an axiom, when needed. In terms of proof-search the daimon can be seen as giving up, and the same in terms of games: in that case you just lose.

In terms of designs, the Daimon is the exact dual of the Skunk. The Daimon normalises against every design, whereas the Skunk normalises against no design, but the Daimon, who is definitely very brave. The Daimon has an optimal socialisation, whereas the Skunk lives alone: the only behaviour containing the Skunk is $\top$, which is made of all negative designs; but these designs are, so to speak, not really here, since they have an empty incarnation.

The daimon is a novel object. However, its creation must be ascribed to the paper Girard (1988), which was the first attempt at a purely internal explanation of logic, formulated in terms of permutations[†].
See: **Correctness criterion, Divergence, Dualiser, Expansive, Faith, Geometry of interaction, Negation as failure, Prisoners, Proof-net, Proof-search, Skunk, Stream, To know not and not to know, Xenoglossy**.

• Delocation

In real life, a way to avoid interference. For instance a traditional bandit will rob everybody but his mother; if you delocate him, you get a more uniform behaviour... In the same way when Superman hits a cable car, one takes him away from the *locus* of the... accident, so as to get... an objective trial.

In mathematics, interference corresponds to capture of bound variables, or the fact that two structures do intersect; delocation renames variables, replaces sum with disjoint union, *etc*. In ludics delocation avoids logical interference, that is, the sharing of any *locus* distinct from the base. Typically **G** will – after delocation – only use even biases, whereas **H** will use odd ones. Delocated behaviours interact spiritually, that is, on the basis of their properties, independently of the locations.

The usual logical operations appear as the combination of a delocation with a strict operation; what is important is that this strict operation is... strictly associative, commutative, *etc*., that is, the locative variant is not a poor relative. Delocation enables one to describe the properties of $\mathbf{G} \,\square\, \mathbf{H}$ from the properties of $\mathbf{G}, \mathbf{H}$: this is the essential source of

[†] And which can be easily rephrased as a full completeness theorem for multiplicative linear logic.

logical completeness. Incompleteness essentially occurs with (second-order) quantification, which is the only operation that is intrinsically locative, for want of enough *loci*…

See: **Bias, Copycat, Distributivity, Fax, First-order quantifier, Geometry of interaction, Hilbert hotel, Identity axiom, Interference, Locative logic, Occurrence, Reservoir, Spiritualism, Spiritual logic, Strictness, Twins, Variables**.

• Denotational semantics

Originated by Scott (Scott 1976) and Ershov in the late sixties (Scott domains), improved by Berry (Berry 1978) in the seventies (stability), leading to coherent spaces (Girard 1987a), hypercoherences (Ehrhard 1995)*etc*., denotational semantics is an important area, and was essential in the discovery of linear logic and ludics. Denotational semantics is a particular case of categorical semantics, with a *concrete* twist.

The opposition between denotational and operational semantics is unfair to denotational semantics.

See: **Categorical semantics, Coherent space, Dualiser, Hypercoherence, Implicit, Operational semantics, Scott domain, Stability**.

• Design

Designs play the role usually devoted to proofs, λ-terms *etc*. in the usual syntax, and functions, cliques *etc*. in denotational semantics. But this is only an analogy.

Designs are not linked to any particular syntax. They are essentially inspired from linear logic, especially the multiplicative/additive fragment, with the idea that this fragment is in some vague sense universal. Other inspirations were taken from pure λ-calculus (especially Böhm trees) and infinitary (non-well founded) logic.

Designs are not syntax at all. First they use no logical symbols, only *locations*; of course, it is possible to encode syntax in locations, but impossible to impose those restrictions typical of syntax: there is no *croupier* that can forbid us, say, to play number 37. Second, they are badly infinite, both in width and depth: the typical example of a design is the *fax* who has full infinite negative branchings and which is badly not well-founded. Last but not least, designs are not even supposed to be recursive.

They are not semantics either, although they issued from the tradition of semantics of proofs, which started with Kolmogorov, and embody later developments such as Scott domains or coherent spaces. Designs can even look like classical models, but their structure is much subtler than any of the usual semantical artifacts.

See: **Behaviour, Böhm tree, Cut-net, Dessein, Dessin, Fax, Locus, Maul, Objects and properties, Phase semantics, Proofs** *vs.* **models, Separation, Slice, Semantics, Strategy, Syntax, Type**.

• Dessein

In French ‘dessein’ means ‘project, plot’, as in the sentence ‘Le savant fou ruminait de noirs desseins.’ The design-as-dessein is the real notion, and is defined as a set of chronicles. Desseins are the streamlike versions of designs. Typically, the pitchforks occurring in a dessin cannot be figured out, since they are ‘in the process of being built’.

See: **Action, Chronicle, Design, Dessin, Non-determinism, Propagation, Separation, Stream, View**.

• Dessin

In French ‘dessin’ means ‘drawing, picture’. Designs-as-dessins are convenient presentations of desseins, in a proof-style reminiscent of infinitary logics: a dessin looks like a proof in sequent calculus, in which sequents have been replaced with pitchforks. But they are not the real object, they are a *reification*, that is, they represent something like the eventual version of a streamlike underlying object. In practice the distinction between desseins and dessins is not that big; everything can be done with dessins, provided we stay streamlike.

See: **Action, Design, Dessein, Pitchfork, Propagation, Reification, Separation, Stream**.

• Dialectica interpretation

Published only in 1958 in the philosophical journal *Dialectica*, but apparently much older, this is Gödel's contribution to an interactive, dialectic, interpretation. What can we say about it without being unfair? It is not Gödel's best work, but it stands miles above the bureaucratic failure of the Lorenzen school.

Roughly speaking, the interpretation looks like $\exists x \forall y A$, where A is quantifier-free. x can be seen as a strategy, winning the game A against any counter-strategy y. The idea is not bad, but the formulation induces a dissymmetry between `Proponent` and `Opponent`, and the interpretation of logical connectives becomes a nightmare (think of the implication) because of this bias. The interpretation works, but it definitely leaks. The attempt by De Paiva (Paiva 1989) at a simplification of *Dialectica* by means of linear logic was interesting, but not sufficient to give a second breath to this original work.

Since it is by Gödel, every textbook on proof-theory has a compulsory chapter on the unfortunate *Dialectica*, but usually written with the left hand.

See: **Leakage, Game, Game semantics, Lorenzen, System $\mathbb{F}$**.

• Dialectics

Illustrated by Hegel, Marx, not to mention comrade Stalin..., this is a word with a history. For this reason, even if ludics has a lot to do with (the early) dialectics, I decided to create a new expression rather than having to assume the (mostly negative) consequences of the reuse of such a notorious word. By the way, this proximity of ludics to dialectics, which it would be unfair to deny, is not the result of, say, an attempt in the style of Engels's *Dialektik der Natur*, but only the result of a long familiarity with logic, and the contemplation of the symmetries of proof-theory.

See: **Ludics**.

• Directory

A directory is a set of ramifications. A negative rule involves a directory, which can therefore be seen as the 'arity' of the rule. The directory $\P\mathbf{G}$ of a behaviour $\mathbf{G}$ indexes its connected components, so as to get the decomposition

$$\mathbf{G} = \bigoplus_{I \in \P\mathbf{G}} \mathbf{G}_I \tag{203}$$

when $\mathbf{G}$ is positive, and

$$\mathbf{G} = \mathop{\&}_{I \in \P\mathbf{G}} \mathbf{G}_I \tag{204}$$

when $\mathbf{G}$ is negative.

The directory can be defined by means of the designs $\mathfrak{Ram}_{(\langle\rangle,I)}$, $\mathfrak{Dir}_{\mathcal{N}}$.

See: **Bias, Disjunction property, Mystery of incarnation, Ramification**.

https://doi.org/10.1017/S096012950100336X

• Disjunction property

This is the commutation of provability with disjunction. It is unproperly stated as 'a proof of $A \vee B$ is either a proof of A or a proof of B': commonsense tells you that if you have proved B you will not state $A \vee B$. The property should rather be stated as 'a proof of $A \vee B$ is *implicitly* either a proof of A or a proof of B', which means that the proof can be seen as program whose execution will eventually yield either a proof of A or a proof of B. In intuitionistic logic the disjunction property is implemented by means of cut-elimination, since a cut-free proof of $A \vee B$ comes actually from a proof of A or a proof of B. The linear additive disjunction, which is the disjunctive part of intuitionistic disjunction, also enjoys the disjunction property.

In ludics, the disjunction property is just completeness of the additive disjunction, which is written $\mathbf{G} \oplus \mathbf{H} = \mathbf{G} \cup \mathbf{H}$.

See: **Brouwer, Completeness (internal), Consistency, Existence property, Explicit, Implicit, Prenex form**.

• Dispute

There is in fact a third layer below behaviours and designs, namely *disputes*, which are so to speak the plays (remember that designs are strategies and behaviours are games). A dispute is a sequence of actions ending with a daimon, and is not quite the same as a chronicle. Designs can be represented as sets of disputes enjoying certain properties, so that the form $\ll \mathfrak{D} \mid \mathfrak{E} \gg$ converges exactly when $\mathfrak{D}, \mathfrak{E}$ intersect. Disputes are an important element of ludics, but I found it so difficult to start with them that I eventually decided to ignore them, and to start with designs.

The theory of disputes, which will make explicit the relation with denotational semantics, for example, designs-as-cliques, is still to be written.

See: **Behaviour, Coherent space, Design, Denotational semantics, Game semantics**.

• Dissensus

This neologism is the negation of consensus. Corresponds to divergence of the normal form $[\![\mathfrak{D}, \mathfrak{E}]\!]$. For instance $[\![\mathfrak{One}, \mathfrak{Dir}_{\wp_*(\mathbb{N})}]\!]$ diverges, and this dissensus is a way to get rid of the atomic weapon.

The possibility of dissensus makes the tests testable in turn, contrary to the naive approach loosely inspired from Popper, for whom the tests are absolute. Remember that Pauperism denies any meaning to a non-falsifiable formula, for instance to Gödel's theorem. In ludics, Gödel's theorem could be refuted, the problem is that the refuting test could in turn be *recused*.

See: **Atomic weapon, Behaviour, Consensus, Game semantics, Obstination, Pauperism, Referee, Unfalsifiable**.

• Distributivity

The distributivity of the multiplicatives over additives is due to the existence of two adjunctions, and the possibility of expressing $\mathbf{G} \parr \mathbf{H}$ either as (1) the set of designs that send $\mathbf{G}^{\perp}$ into $\mathbf{H}$ or (2) the set of designs that send $\mathbf{H}^{\perp}$ into $\mathbf{G}$. If you want distributivity to the right, you use (1), whereas distributivity to the left requires (2).

Distributivity is part of the general 'associativity' properties of connectives of the same polarity. As soon as one does not respect polarities, distributivity fails. Typically, $\otimes$ does not distribute over &: assuming that enough shifts have been added so as to avoid mismatches of polarities, we see that $\mathbf{G} \otimes (\mathbf{H} \,\&\, \mathbf{K})$ can be implemented by one delocated copy of each of $\mathbf{G}, \mathbf{H}, \mathbf{K}$, whereas there is no way to implement $(\mathbf{G} \otimes \mathbf{H}) \,\&\, (\mathbf{G} \otimes \mathbf{K})$ with only one delocation of $\mathbf{G}$. This reason for the non-distribution of the two conjunctions seems to me more convincing than the usual semantic explanations. Observe that the distribution fails because the connective & is spiritual, hence must be delocated,... but the argument no longer applies for quantifiers, and this is why ⊗ distributes over any intersection (even beyond the basic meaning of distributivity, one would not expect $\forall d(\mathbf{G}_d \otimes \mathbf{H}) = (\forall d\ \mathbf{G}_d) \otimes \mathbf{H}$).

See: **Adjunction, Associativity, Delocation, Prenex form, Quantifier**.

• Divergence

Divergence corresponds to the failure of the normalisation of a cut-net. The output of a diverging net is formally written as the Faith, which is the paragon of divergence, just as the Daimon is the paragon of convergence.

See: **Consensus, Daimon, Dissensus, Faith, Normalisation**.

• Dog

The Dog is the guy who is not interested in salads, but thinks you should not eat them. In most games, doggish play is considered somewhat against the spirit of the game: you do not want to win, but to get a draw – see the Chess literature for instance.

The winning conditions of ludics (parsimony, obstination, uniformity) are not general properties of designs, since non-parsimonious designs and non-obstinate designs are very useful. Typically, the completeness of a tensor relies strongly on the existence of non-parsimonious designs, corresponding to weakening, that is, the existence of a reasonable projection. However, these designs break some implicit commitment, for instance, parsimony commits me to focus on each *locus* generated by my opponent, and uniformity is a stronger form of coherence. This is why losing designs are eventually fired when they have done their job, like an ordinary *independent prosecutor*.

See: **Contraction, Loser, Obstination, Paralogism, Parsimony, Uniformity, Weakening, Winning**.

• Do-it-yourself

Amateurs produce a lot of 'logics', which are difficult to distinguish from the various 'Ph. D. logics' produced in the area of so-called 'Artifical Intelligence'. One can distinguish between Broccoli logics, which are deductive, that is, admit a (poor) formal system, and paralogics, which are not even deductive. One of the best examples is the cut-free system in which everything is provable except for the empty sequent, they call it *paraconsistent*... and they are very serious about it.

See: **Artificial intelligence, Broccoli logics, Non-monotonic logics, Paralogics, Tartuffe**.

• Double negation

This was yet another achievement of Gödel: one can faithfully translate classical logic into intuitionistic logic by adding enough double negations. The reason is very simple, namely that the left part of an intuitionistic sequent is very permissive, whereas the right part is 'politically correct', everything being forbidden... On the other hand, the restrictions on the right part (one formula) yield disjunction and existence properties.

https://doi.org/10.1017/S096012950100336X

When a formula cannot socialise enough on the right, a solution is to transfer it to the left (using negation), then perform whatever classical operations one wants (typically, contractions), before transferring it again to the right... But the notorious prefix '$\neg\neg$' betrays its escapade on the other side of the wall.

Across the Curry–Howard isomorphism, the double negation corresponds to transforming a lambda-term into continuation-passing style, see, for example, Hatcliff and Danvy (1994).

See: **Classical logic, Contraction, Intuitionistic logic, Linear negation, Self-interpreter, Sequent calculus**.

• Dualiser

The dualiser is the behaviour $\mathbf{0}$ of base $\vdash$ (the only available choice). So a strange question arises: how is this possible without accepting the principle $\perp = 0$, which is one possible formulation of weakening? If ludics were a categorical semantics, we would be stuck... First observe that the fact that the base is empty is essential: the smallest behaviour $\mathbf{0}_\xi$ of base $\vdash \xi$ cannot be seen as a dualiser (hint: this contradicts parsimony). Moreover, $\perp$ is a negative behaviour and cannot be the dualiser, since $\mathbf{G} \multimap \perp$ only makes sense when $\mathbf{G}$ is positive.

See: **Adjunction, Boots, Behaviour, Denotational semantics, Daimon, Empty sequent, Linear negation, Parsimony, Weakening**.

• Dualism

This is a philosophical approach characterised by a schizophrenia between matter and ideas – think of Descartes. The pregnancy of dualism is responsible for familiar oppositions: matter *vs.* spirit, objects *vs.* properties, contents *vs.* form, and of course semantics *vs.* syntax. These oppositions are useful, which does not mean that they have any conceptual significance.

See: **Form** *vs.* **contents, Laplace, Monism, Objects** *vs.* **properties, Philosophical logic, Realism, Schizophrenia**.

• Dupond et Dupont

The fake twins of Hergé (adepts of La Palice: 'C'est mon opinion et je la partage') are lost with their Jeep in the desert (Hergé 1950, pp. 29–30), up to the moment they find a track to follow; they indeed follow their own path which seems more and more 'crowded' as they make additional loops. A perfect metaphor of Tarskism, Paraphrases and Jurassic foundations.

See: **Jurassic Park, La Palice, Meta, Metaphor, Münchhausen, Paraphrases, Self-interpreter, Tarskian semantics**.

• Empty sequent

Some traditionalists have been shocked to see that ludics is essentially based on the closure principle, that is, normalisation of 'proofs' of the empty sequent. But they forget that all traditional proof-theory is about the study of proofs of the empty sequent, with the idea that there is no such proof. All this work devoted to non-existing objects, what a waste... Ludics had the generous idea of providing proof-theory with a large stock of 'proofs' of the empty sequent, so that our fundamentalists feel no longer frustrated and

contemplate abundant examples of their favourite object.
See: **Dualiser, Fundamentalism, Jurassic Park, Ludics**.

• η-EXPANSION
In terms of categories, η-expansion corresponds to the unicity of the solution of the universal problem defining such and such connective. This can be translated as the decomposition of the general identity axiom $A \vdash A$ of sequent calculus into its atomic instances $X \vdash X$; this process has perfect analogues in natural deduction, λ-calculus, proof-nets... Now the syntactical distinction between atomic and non-atomic axioms cannot be interactively observed: for instance, Böhm's theorem separates solvable terms up to η-expansion. The separation theorem of ludics also corresponds to the choice of η-expansion: in fact we expand 'so much' that the identity axiom never shows up!
See: **Atomic proposition, Böhm tree, Fax, Identity axiom, λ-calculus, Natural deduction, Proof-nets, Separation, Sequent calculus**.

• ETHICS
An ethics corresponds to the idea of those proofs of a given formula that come from the syntax: an ethics is a set of designs $\mathbf{E}$ of a given basis, which generates a behaviour $\mathbf{E}^{\perp\perp}$. An ethics is complete when $\mathbf{E}^{\perp\perp} = \mathbf{E}$ (up to incarnation).
See: **Behaviour, Biethics, Completeness (internal), Gödel's incompleteness, Incarnation, Incompleteness**.

• EXACTNESS
A design is exact when it can be written without waste of contexts, so to speak without weakening. A weaker, but interactive, version of the same is *parsimony*. Full completeness for $\mathbf{MALL}_2$ is established under the assumption of exactness, not of parsimony. The leakage between exact and parsimonious should induce new developments.
See: **Full completeness, Leakage, Parsimony, Weakening, Xenoglossy**.

• EXISTENCE PROPERTY
This is the commutation of provability with existence: 'A proof of $\exists n A[n]$ is *implicitly* a proof of $A[n]$ for a certain integer n'; this is similar to the disjunction property. $\exists n A[n]$ therefore means that we have a program enabling us to extract a witness n such that $A[n]$ from the proof of existence.

Ludics refutes the existence property for general quantifiers: the intuitionistic formula $(\forall d A_d) \Rightarrow B$ can be identified with $\exists d(A_d \Rightarrow B)$, this is an instance of the general prenex form principle of ludics. This does not apply to numerical quantifiers, since $\exists n A$ is in fact short for $\exists d\ (d \in \mathbb{N} \ \wedge \ A)$; the existence property for natural numbers will be valid in ludics.
See: **Brouwer, Disjunction property, Explicit, Implicit, Prenex form, Quantifier**.

• EXPANSIVE
This is an antonym of 'recessive', and corresponds to 'streamlike', $\mathbf{\Pi}^1$, $\mathbf{\Sigma}^0_1$. The typical expansive property is provability, or halting. The Daimon ✠ is the abstract form of expansivity.
See: **Daimon, Falsifiable, Pauperism, Recessive, Σ and Π formulas, Stream**.

https://doi.org/10.1017/S096012950100336X

• Explicit

Something explicit has no value in itself: what do you think of ‘yes’? Of course this word is more interesting when not said by a yes man, but corresponds to a question. The question corresponds to the implicit, the answer to the explicit, and they are linked by explicitation.

See: **Answer, Cut-elimination, Cut-rule, Disjunction property, Existence property, Explicitation, Frege, Implicit, Proofs as programs, Question**.

• Explicit mathematics

This is a perfect oxymoron: reasoning is precisely about the implicit. Mathematics is neither explicit nor formal, but it should be explicitable and formalisable.

See: **Abstraction, Formal, Formalisable, Explicitation, Frege, Implicit, Oxymoron**.

• Explicitation

This is the art of extracting the implicit contents. It corresponds to normalisation and cut-elimination in logic; in computer science it corresponds to the execution of a program.

See: **Computer science, Constructivity, Cut-elimination, Explicit, Explicit mathematics, Implicit, Normalisation, Proofs as programs**.

• Exponentials

These connectives have been introduced to keep a reasonable amount of classicism within linear logic: linear logic is issued from the denotational equation

$$A \Rightarrow B \;= {!A} \multimap B \tag{205}$$

$!A$ is traditionally interpreted as ‘*A ad libitum*’, that is, as the absence of any resource bound (see the gastronomic menu). They are therefore closer to the traditional view of logic, which deals with eternal truths, and they can be described as the non-linear part of linear logic.

It took me a very long time (at least two years, 1997–99) to understand the actual decomposition of exponentials, the main difficulty being to adapt the seemingly universal polarity principle to that case:

— $!\mathbf{G}$ is defined for negative $\mathbf{G}$ and is positive.

— $!\mathbf{G}$ can be decomposed as $\downarrow\sharp\mathbf{G}$, where $\downarrow$ is the usual shift (change of polarity) and $\sharp\mathbf{G}$ is a negative behaviour, the actual exponential.

$?\mathbf{G}$ is dually written as $\uparrow\flat\mathbf{G}$, where $\flat$ is a connective mapping positive behaviours to positive behaviours.

See: **Classical logic, Contraction, Gastronomic menu, Linear logic, Polarity, Xenoglossy**.

• Faith

The only definitely partial design, the exact opposite of the daimon, and the paragon of recessivity, that is, of divergence. The relation

$$\mathfrak{Fid} \preceq \mathfrak{D} \preceq \mathfrak{Dai} \tag{206}$$

expresses the fact that all real dessins lie between Faith and Daimon. The daimon provides us with an immediate negative answer: the daimon is a total easy-going design. Faith would also provide us with the same negative answer (that is, that the design does not start with a proper rule), if we were patient, but we have the faith... Jesus said, just before taking off ‘I am coming back’ – and some people are still waiting.

When a process is stalled on the computer, either we wait (Faith) or we use `C-c` (Daimon). In both cases we do not get what we are after, but we have some compensations, the hope that it will eventually come, or the possibility to try something else. The same dilemma $\Omega/\maltese$ occurs in most activities, for example, when waiting for a bus in Roma. Paralogics are usually based on something like the identification between Ω and $\maltese$, and you can figure out the disaster arising.

See: **Böhm tree, Daimon, Divergence, Geometry of interaction, Halting problem, Intuitionistic loic, Locus, Non-monotonic logics, Partial design, Prisoners, Reification, Recessive, Solvable, Stream, To know not and not to know**.

• Falsifiable

The neo-positivist philosopher Popper introduced this terminology to characterise what a real scientific statement should be: a property, that you can verify up to an arbitrary degree of precision – ‘for any precision N the property holds up to size N’ – and such a statement is true as long as we have no counterexample, that is, no falsification. These formulas are familiar to the logician, and called $\mathbf{\Pi}^0_1$ ($\mathbf{\Sigma}^1$ in ludics); the emphasis on such properties is indeed Hilbert’s *credo*, which is expressed in Hilbert (1926).

I prefer to use the word ‘recessive’ for this type of property, since the more you check, the less you get. Observe that provability, like all streamlike activities, is not recessive, the more you check, the more you get. The mismatch between expansive and recessive, expressed by the incompleteness theorem or the undecidability of the halting problem shows the limitation of Pauperism – the nickname I give to this abusive extension to mathematics of Popper’s paradigm.

The emphasis on falsifiability can be understood from the naive ‘physician ideology’ for which science looks like a cook book: a law is something that has not been concretely disproved yet.

See: **Armageddon, Expansive, Fermat, Hilbert, Inconsistency proof, Halting problem, Gödel’s incompleteness, Incompleteness, Interactivity, Medicine, Monsantism, Objects and properties, Pauperism, Recessive, $\mathbf{\Sigma}$ and $\mathbf{\Pi}$ formulas, Stream, Unfalsifiable**.

• Fax

The most general form of a fax is a design of base $\xi \vdash \xi'$ implementing a delocation between ξ and ξ'. The fax corresponds to the identity axiom of logic, and in terms of strategies, it works like the copycat strategy, which consists of systematically recopying each ‘move’ of the other.

The base is negative, that is, `Opponent` starts. The first negative rule has a premise of index I for any ramification I, corresponding to any possible move of the `Opponent`. Above I, `Proponent` answers with a similar move (the same I, but on ξ'), and we are left with a similar problem, but with the $\xi' * i \vdash \xi * i$ instead. The same structure is repeated *ad nauseam*, in particular, the fax is very badly not well-founded. The fax is the paragon of the *barbichette*: he systematically recopies, hence he cannot make the mistake first!

See: **Atomic proposition, Barbichette, Copycat, Cut-net, Delocation, η-expansion, Identity axiom, Occurrence, Twins**.

• Fermat

Fermat's last theorem is the paragon of the falsifiable (or recessive) property: something holds for all natural numbers *n*. In spite of the proof of Wiles, it might still be possible to falsify this result, namely by finding *ad hoc* natural numbers such that $a^n + b^n = c^n$; in this unlikely situation, we would conclude that Fermat's last theorem is wrong, and also that mathematics is inconsistent. The situation is quite different with Gödel's theorem.

See: **Answer, Falsifiable, Inconsistency proof, Question, Recessive, Unfalsifiable**.

• First-order quantifier

A universal first-order quantifier is usually understood as a $\&$, possibly uniform. Ludics interprets first-order quantification by a plain intersection, so that first-order quantification gets the qualities (prenex forms) and the drawbacks (incompleteness, for example, prenex forms) of the second-order case. I definitely prefer these simple and incomplete first-order quantifiers to the traditional ones. But the real question is not preference, it is use, and there are clearly uses for a complete first-order existential.

Observe that the constructive tradition (for example, denotational semantics) has been extremely unimaginative on this issue: as far as I know, people have only been able to interpret $\forall$ as a sort of infinitary conjunction, $\&_{d \in \mathbb{D}}$, which is incomplete too, since it inherits all the drawbacks of infinitary logics, without any interesting property such as prenex forms. The solution presumably lies in a use of uniformity: replace $\&$ by a 'symmetric' variant in the spirit of what we did in Chapter 8 for the *symmetric sum*.

By the way, observe that we cannot interpret second-order quantification as a $\&$, for we cannot delocate: there are many more behaviours than *loci* !

See: **Bihaviour, Delocation, Harmony, Infinitary logics, Prenex form, Quantifier, Uniformity**.

• Fixed point

The streamlike, expansive nature of objects, makes it possible to get various fixed-point theorems, for example, in λ-calculus. But this badly fails for properties.

See: **λ-calculus, Objects and properties, Recursive type**.

• Focalisation

This property, due to Andreoli, (Andreoli and Pareschi 1991), is dual to invertibility. It says that, provided one does it at the right time, one can consider a cluster of positive connectives as a synthetic connective, and perform the relevant rules simultaneously. Connectives of the same polarity therefore associate, which, depending on the situation, may mean commutativity, associativity, neutrality, distributivity... It turns out that the graphical style $\parr, \bot, \&, \top$ versus $\otimes, 1, \oplus, 0$ clearly distinguishes the negatives from the positive... but this graphism was found long before the discovery of Andreoli. The idea behind the graphism was to memorise the remarkable isomorphisms (for example, the distribution $\otimes / \oplus$), but focalisation shows that these isomorphisms are actually due to the identity of polarities: this explaining that. The dual properties of invertibility/focalisation express the *associativity* of logic.

See: **Associativity, Focus, Invertibility, Locus, Polarity, Sequentiality, Stoup, Synthetic connective, Time in logic**.

- Focus

The focus is the formula on which a focalisation is performed. In ludics, it is the locus of an action.

See: **Action, Focalisation, Locus**.

- Forgetful interpretation

The forgetful interpretation of system $\mathbb{F}$ consists of removing all type decorations in terms, so as to get a pure λ-term. The correct interpretation is that this underlying λ-term is the real object, whereas the typings are comments, to be ignored at runtime. We do the same in ludics, but more systematically, by forgetting everything but the locations.

See: **λ-calculus, Locus, PER-model, System $\mathbb{F}$**.

- Formal

Hilbert treated mathematics as a formal activity, which is a nonsense, if we take it literally... But what should we think of those who take thought as a formal activity?

See: **Abduction, Artificial intelligence, Bureaucracy, Cordwainer Smith, Explicit mathematics, Formalisable, Laplace, Sense of rules**.

- Formalisable

Mathematics is formalisable. Theorem provers do not, and cannot, work, but proof-checkers are very efficient. For instance, my first theorem, normalisation for system $\mathbb{F}$ (Girard 1971), has been formally checked by a computer (Altenkirch 1993): the work was begun by Berardi and finished by Altenkirch.

See: **Abduction, Constructions, Explicit mathematics, Formal, Higher-order logics, System $\mathbb{F}$**.

- Form *vs.* contents

This is a schizophrenia typical of dualism, and of no conceptual value. Most of the story of modern proof-theory (before ludics) was precisely to extract the actual content from the form.

See: **Categorical semantics, Dualism, Ludics, Schizophrenia**.

- Formula

Formulas are one of the main syntactical artifacts. In ludics, formulas become behaviours, or bihaviours. The notion of behaviour being syntax-free, it is linked to no particular bureaucratic system.

See: **Behaviour, Bihaviour, Bureaucracy**.

- Foundations

There is a desire for foundational studies. However, Gödel's theorem forbids any reductionist foundation, such as fundamentalism. What remains is the possibility of disclosing deeper structures, and I believe that ludics is part of this process.

See: **Consistency proof, Fundamentalism, How and why, Jurassic Park, Ludics**.

- Frankenstein

Pr. Dr. Frankenstein created a monster out of a faulty analysis. In the same way, the hasty analysis of Lorenzen *et al.* (Lorenzen 1960; Lorenz 1968; Felscher 1985) could only lead to a monstrous synthesis.

See: **Analysis and synthesis, Lorenzen**.

https://doi.org/10.1017/S096012950100336X

• Frege

Isn't the distinction between *Sinn* and *Bedeutung*, sense and denotation, just the same as the implicit/explicit distinction?

See: **Explicit, Explicitation, Implicit**.

• Full completeness

The terminology is due to Abramsky, although the idea was first formulated in my paper Girard (1991). Abramsky tried also to convey the fullness of the functor (Abramsky and Jagadeesan 1994) that interprets proofs: 'With full completeness, one has the tightest possible connection between syntax and semantics. We are not aware of any previous published results of this type: however, the idea is related to representation theorems in category theory (Freyd and Scedrov 1990), to full abstraction theorems in programming language semantics (Milner 1975; Plotkin 1977), to studies of parametric polymorphism (Bainbridge *et al.* 1990; Hyland *et al.* 1989), and to the completeness conjecture in Girard (1991).'

The challenge was to prove full completeness for linear logic. This has proved to be a very good question, for which I first gave an (unsatisfactory) answer in Girard (1999a) (unsatisfactory because of the pregnancy of the *referee*, disguised as a *Par-monoid*, for the fluent reader). In Chapter 11 we established the result in the absence of exponentials, under a slightly stronger hypothesis, exactness.

See: **Answer, Categorical completeness, Completeness (external), Exactness, Geometry of interaction, Läuchli semantics, Leakage, Mix, Question, Referee, Xenoglossy**.

• Fundamentalism

Fundamentalism is the current reductionist approach to *foundations*: in spite of Gödel's theorem, it should be possible to found mathematics on something simpler. As a personal recollection (1972), when I said to Schütte that, according to Kreisel, a consistency proof of set-theory would have no value at all, he replied 'Maybe, but I would feel better if I saw one!'.

See: **Constructivism, Empty sequent, Foundations, Jurassic Park, Predicativity**.

• Game

Gentzen's first consistency proof (1936) (Gentzen 1969c) is presumably more interesting than the second one. In this proof he developed (this is explained in a confusing way) an interactive interpretation of arithmetic, which must be considered as the first interactive explanation of logic. Gödel's Dialectica interpretation (Gödel 1958) is also a game semantics. However the idea must be ascribed to Lorenzen, even if the work of his school is technically of little interest, which is not the case of the works of Gentzen and Gödel.

In between the Lorenzen School and recent work, the curious work of Blass (Blass 1972), see also Blass (1992), which was inspired from set-theory, stands alone. This is a genuine work, which anticipates (fifteen years earlier) linear logic to some extent.

See: **Atomic weapon, Behaviour, Composition of strategies, Consensus, Dialectica interpretation, Dispute, Dissensus, Gentzen, Game semantics, Leakage, Lorenzen, Strategy**.

https://doi.org/10.1017/S096012950100336X

• Game semantics[†]

Contrary to the usual denotational semantics, this approach makes time explicit. Basically, two agents (traditionally called `Proponent` and `Opponent`) construct an alternating sequence of tokens, subject to certain rules depending on the type of the game. This sequence can be understood as the trace of the interaction between a program (`Proponent`) and its environment (`Opponent`) (Abramsky and Jagadeesan 1994; Danos *et al.* 1996). Three traditions have merged, giving rise to a very flexible tool for constructing the semantics of the execution of programming languages:

— The *proof theoretic* tradition, initiated by Gentzen and followed by the Lorenzen school, and more recently by Blass (Blass 1972; Blass 1992), and Abramsky, Malacaria and Jagadeesan (Abramsky *et al.* 2000) interprets a proof of A as a (winning) strategy allowing a *defender* of A to answer any attempt to contradict A.

— The *recursivity* tradition, initiated by Kleene and Gandy, and recently by the $\mathtt{H}^2\mathtt{O}$-games of (Hyland and Ong 2000) and (Nickau 1994), originated in a description of higher order recursive functionals as strategies in adequate games.

— Finally, the *denotational* tradition goes back to the work of Berry and Curien (Berry and Curien 1982) on sequential algorithms.

A comprehensive survey can be found in (Abramsky and McCusker 1999).
See: **Game, Linear negation, Lorenzen, Loser, Sequential algorithm, Shift.**

• Gastronomic menu

This is a good metaphor due to Lafont. The following menu is proposed

```
Price = FF 320
Starter: Huîtres ou Melon (en fonction des arrivages)
Main: Hamburger
Vegetables: Frites à volonté
Last: Fromage ou Dessert
```

Which is rendered by $\mathtt{Price} \Rightarrow (\mathtt{Starter} \otimes \mathtt{Main} \otimes \mathtt{Vegetables} \otimes \mathtt{Last})$. The linear implication '$\multimap$' conveys the idea of consuming a resource (here money) to get several items together (the 'and' is rendered by $\otimes$). The two *or* are rendered by $\oplus$ and &, depending on who controls the choice (in the starter it is the outer world, in the last it is me), and the frites being unlimited get a 'bang'. We eventually arrive at

$$\mathtt{320} \Rightarrow ((\mathtt{Huîtres} \oplus \mathtt{Melon}) \otimes \mathtt{Hamburger} \otimes !\mathtt{Frites} \otimes (\mathtt{Fromage} \,\&\, \mathtt{Dessert})) \qquad (207)$$

One can replace `Price, Huîtres,Frites,...` with real objects, typically abstract machines, in which problems of resources, external or internal non-determinism make sense. This was first done with Petri Nets by Asperti (Asperti 1987), and then extended to various machines (Lincoln *et al.* 1990; Kanovitch 1991). The complexity characterisations of various fragments of linear logic convey the precise meaning of what Lafont's menu suggests: linear logic expresses much more than the usual logic.
See: **Allegory, Exponentials, Metaphor, Phase semantics, Resource.**

[†] By Laurent Regnier.

https://doi.org/10.1017/S096012950100336X

• Gentzen

Gentzen is responsible for three essential ideas, all of them introduced in the hope of completing Hilbert's consistency program, namely sequent calculus (Gentzen 1969a), game semantics (Gentzen 1969c) and ordinal analysis (Gentzen 1969b).

See: **Astrology, Consistency proof, Game, Game semantics, Hilbert, Infinitary logics, Lewis Carroll, Ordinal analysis, Sequent calculus**.

• Geometry of interaction

Geometry of interaction (GoI) (Girard 1989a) interprets proofs by means of operators on the Hilbert space $\mathscr{H}$, always the same. The tensor product consists in forming, from two operators Φ, Ψ on space $\mathscr{H}$ a new operator 'sum' that only depends of the isomorphism class of Φ, Ψ: the natural one is given by the matrix

$$\Theta = \begin{bmatrix} \Phi & 0 \\ 0 & \Psi \end{bmatrix} \tag{208}$$

But Θ operates on the wrong space, namely $\mathscr{H} \oplus \mathscr{H}$, and one makes use of an isomorphism $x \oplus y \mapsto p(x) + q(y)$ of $\mathscr{H} \oplus \mathscr{H}$ in $\mathscr{H}$ to get an operator of $\mathscr{H}$, namely $p\Phi p^* + q\Psi q^*$. Here p, q enjoy the equalities $p^*p = q^*q = 1, p^*q = q^*p = 0$, so that $p\Phi p^*$ and $q\Psi q^*$ are nothing but 'disjoint' isomorphic copies of Φ and Ψ.

In ludics, the delocations φ, ψ introduced in Section 5.2.2 are the exact analogous of the p, q of GoI: if $\mathbf{e}_n$ is a base of $\mathscr{H}$, we can define p, q by $p(\mathbf{e}_n) = \mathbf{e}_{3n}$, $q(\mathbf{e}_n) = \mathbf{e}_{3n+1}$.

Geometry of interaction was conceived as an interactive approach to logic – not quite a game semantics: this aspect was made explicit in Abramsky *et al.* (2000). To quote Abramsky (private communication) on this connection: 'The work of myself and Jagadeesan started with the attempt to fashion a proper categorical model from game semantics ideas, taking Blass's work (Blass 1992), which did not succeed in doing this, as a starting point. In the course of proving full completeness in the paper (Abramsky and Jagadeesan 1994), I realised that proof-like strategies were history-free, and history-freeness became a key condition in the model. Then, we realized that a correspondence could be made between semantics of proofs as history-free strategies, and Geometry of Interaction – which I think was a genuine insight at the time. For example, composition of strategies is defined in a very different looking way from the Execution formula, and one has to show they actually correspond.'

The problem with GoI is that it leaks, in the sense that it cannot tell the difference between $\mathfrak{Fid}$ and $\mathfrak{Dai}$, both of them being rendered by the operator 0. Of course GoI must eventually be revisited.

See: **Cut-net, Daimon, Delocation, Faith, Full completeness, Game semantics, Hilbert hotel, Leakage, Reservoir, Resource, Spiritualism, Treason, Xenoglossy**.

• Gesticulation

It is not true that proof-search does not work, with the package `Broccoli` you can prove completeness for your favourite logic.

```
%\semantics[option]{argument}. Yields a semantics (in bold
% face) for argument. Options [c] for a completeness
% theorem, [s] for a syntax (written in italics) for argument.
```

See: **Algebraic logic, Broccoli logics, Classical model, Higher-order logics, Kripke models, Non-associative logic, Semantics, Soundness, Syntax**.

• GÖDEL

Gödel appears here for his completeness theorem, which was proved before the syntax/semantics distinction was introduced, the incompleteness theorem, the undefinability of truth (formulated by Tarski), and the Dialectica interpretation.

See: **Completeness (external), Classical model, Dialectica interpretation, Gödel's incompleteness, Incompleteness, Truth**.

• GÖDEL'S INCOMPLETENESS

There are two incompleteness theorems:

— The second incompleteness theorem states the impossibility of fixing your spectacles while wearing them. This is common sense – by the way what would you think of somebody whose credit would rely on his own declarations? There is a popular literature on the topic, trying to point out that after all Gödel was warped and that the theorem is hardly more than a puzzle.

— The first incompleteness theorem, which is the refutation of Hilbert's program. The proof originates in Cantor's diagonal, and induced Turing's undefinabilty of the Halting problem.

It must be remarked that the usual interpretation of the first incompleteness theorem, the mismatch between provability and truth, although technically correct, is perhaps misleading: one gets the impression of a totality (the true statements) that one cannot reach (by formal methods)... But is there, conceptually speaking, such a totality? I don't know, and I would guess that this totality is nothing more than a (convenient) convention. By the way, the first incompleteness theorem can be restated as the existence of a closed formula that cannot be proved nor disproved, which means that provability is incomplete in itself.

See: **Armageddon, Artificiality, Cantor's diagonal, Completeness (internal), Expansive, Fermat, Gödel, Halting problem, Incompleteness, Münchhausen, Paralogics, Recessive, Unbounded operator**.

• GUSTAVE FUNCTION

This fantastic counterexample is due to Berry (Berry 1978) and is a major contribution to the theory of sequentiality. The Gustave function takes three Boolean arguments and returns a completely irrelevant output. The equations are the following

$$\begin{array}{rcl} G(\mathtt{tt},\mathtt{ff},z) & = & a \\ G(x,\mathtt{tt},\mathtt{ff}) & = & b \\ G(\mathtt{ff},y,\mathtt{tt}) & = & c \\ G(\mathtt{tt},\mathtt{tt},\mathtt{tt}) & = & d \\ G(\mathtt{ff},\mathtt{ff},\mathtt{ff}) & = & e \end{array} \tag{209}$$

(the last two equations have been added to Gustave's definition to make the function total).

The algorithm thus defined is not *sequential*, that is, when we compute G, we have no first question to ask about the input (for example, '*Give me the first argument*': when

$y = \mathtt{yy}, z = \mathtt{ff}$, the first argument is irrelevant). Of course, sequentiality is restored if one replaces the second equation by

$$\begin{array}{rcl} G(\mathtt{tt}, \mathtt{tt}, \mathtt{ff}) & = & b \\ G(\mathtt{ff}, \mathtt{tt}, \mathtt{ff}) & = & b \end{array} \tag{210}$$

In fact, Berry studied another counterexample, the *parallel or* of Plotkin. This counterexample is refuted by coherent spaces[†]. The Gustave function is refuted by hypercoherences. In terms of logic, Gustave can be rephrased as the would-be proof of $\vdash A \oplus (B \,\&\, C), A' \oplus (B' \,\&\, C'), A'' \oplus (B'' \,\&\, C'')$ made from 5 cases, $\vdash B, C', A''$, $\vdash A, B', C''$, $\vdash C, A', B''$, $\vdash B, B', B''$, $\vdash C, C', C''$ (B, C, A correspond to 'true', 'false' and 'don't care', respectively). Ludics took the simplest solution, namely, by saying that a positive sequent (pitchfork) $\vdash A \oplus (B \,\&\, C), A' \oplus (B' \,\&\, C'), A'' \oplus (B'' \,\&\, C'')$ comes from a positive rule, which excludes Gustave.

See: **Coherent space, Hypercoherence, Parallel or, Polarity, Scott domain, Sequentiality, Stability**.

• Halting problem

When a program gets stalled, can you decide whether or not to use `C-c`? The answer, due to Turing, is definitely *no* and is known as the undecidability of the halting problem; it is a minor variation of Gödel's theorem, or of Cantor's diagonal argument.

See: **Cantor's diagonal, Closed-world assumption, Computer science, Daimon, Faith, Falsifiable, Gödel's incompleteness, Incompleteness, Negation as failure, Non-monotonic logics, Paralogics, Prisoners, Unbounded operator**.

• Harmony

Albert Camus[‡] said in 1957: 'Entre la justice et ma mère, je choisirai toujours ma mère.' We can imagine that the tension between spiritual, (that is traditional) logic and locative logic will increase: between the respect of principles (completeness *etc.*) and harmony (prenex forms *etc.*), harmony is preferable. Completeness is a valuable principle, a way of organising the logical space, but is incompleteness that bad? Arithmetic is incomplete, and perhaps slightly more interesting than the complete predicate calculus.

See: **Admissible rule, Completeness (external), First-order quantifier, One, Prenex form, Savoir-vivre**.

• Hauptsatz

This literally means the 'main result': this is the expression used by Gentzen in his paper Gentzen (1934) for the cut-elimination theorem.

See: **Cut-elimination, Sequent calculus**.

• Herbrand model

This is a typical antiphrasis: Herbrand constructed a quantifier-free *theory*, which cannot be treated as a model, unless one insists on mocking classical logic. For instance, instead of provability, some speak of truth in the 'least Herbrand model'... As soon as they depart from the basic predicate case, these 'Herbrand models' become a true nonsense.

See: **Antiphrasis, Black Mass, Classical model, Non-monotonic logics, Proofs vs. models**.

[†] Originally by Berry's notion of *stability*.

[‡] This controversial statement was linked to the Algerian War: justice is spiritual, whereas the mother (Algeria) is locative.

• Heyting's semantics

Also due to Kolmogorov, the semantics of proofs interprets proofs in a functional way, 'A proof of $A \Rightarrow B$ is a function mapping proofs of A to proofs of B'. This prefiguration of Curry–Howard, which is perfectly correct by the way, had a difficult life – in particular because of sectarian polemics.

See: **Brouwer, Curry–Howard, Logical relation, Realisability, Saaty volume**.

• Higher order logics

These are generalisations of logic above the second order. There is a strong difference between first and second-order, but third order is very much like second order; it is also of little use, except for the writing of easy papers; one can use the Emacs command `M-x higherorder-my-file` to that effect. This is an example of really formal mathematics, doable by a computer.

See: **Formalisable, Gesticulation**.

• Hilbert

Hilbert launched his famous program on consistency proofs in the twenties; see, for instance, Hilbert (1926). The program was refuted by Gödel's incompleteness theorem in 1931, which did not prevent people from continuing the program. What is remarkable is not this illustration that beliefs are stronger than truth, but that essential ideas, in particular sequent calculus, came out of the program. The program is therefore the paragon of the good question: a definite negative answer, and outstanding side effects. Of course these side effects were found long long ago, in the thirties, so maybe people should work on something else...

See: **Answer, Astrology, Gentzen, Gödel's incompleteness, Incompleteness, Jurassic Park, Laplace, Proof-theory, Question, Sequent calculus**.

• Hilbert hotel

The Hilbert hotel provides an example of a typical delocation: assume that $\mathfrak{D}$ has booked rooms $1, 12, 13$, and $\mathfrak{E}$ has booked rooms $7, 13, 21$, then Hilbert uses delocations $n \rightsquigarrow 3n$ and $n \rightsquigarrow 3n + 1$ to accommodate everybody, $\mathfrak{D}$ in rooms $3, 36, 39$ and $\mathfrak{E}$ in rooms $22, 40, 64$[†]. Although rooms are perhaps not strictly isomorphic, the clients will not notice the difference.

See: **Delocation, Geometry of interaction, Spiritualism, Tensor product**.

• How and why

The *why* is the question everybody would like to answer, a noble question. So far logic has concerned itself with the why, typically in the quest for 'foundations'. The modesty of the output only matched the pretension of the question. I do think that logic should rather look at the *how*, that is, the immanent structures at work. Only when a critical amount of material has been gathered can we start to, partly, cope with the *why*.

See: **Constructivism, Constructivity, Foundations**.

[†] The rooms $3n + 2$ are not used, so there is a fresh Hilbert hotel, in fact the hotel keeper does not quite know how many people will eventually show up in groups $\mathfrak{D}, \mathfrak{E}$.

• Hypercoherence[†]

An analysis of sequentiality, as it is described in the sequential algorithm model of Berry and Curien (Berry and Curien 1982), first led Bucciarelli and Ehrhard to the notion of *strong stability*. A strongly stable function is required to preserve more meets than simply those of all bounded sets (Bucciarelli and Ehrhard 1993). Then Ehrhard introduced hypercoherences in Ehrhard (1995), a model of linear logic where morphisms are strongly stable. Hypercoherences are hypergraphs, whereas coherent spaces are graphs. He also showed that the hypercoherent hierarchy of simple types is the 'extensional collapse' of the sequential algorithms hierarchy in Ehrhard (1999), and van Oosten and Longley obtained similar characterisations in realisability settings (Van Oosten 1997; Longley 1998).

See: **Categorical semantics, Coherent space, Denotational semantics, Gustave function, Linear logic, Sequential algorithm, Sequentiality, Stability**.

• Identity axiom

This should be written $A \vdash A'$, where A, A' are isomorphic, with disjoint bases ξ, ξ'. The principle is implemented by the fax of base $\xi \vdash \xi'$ corresponding to the isomorphism. The principle should be called 'isomorphism, delocation' rather than 'identity'.

See: **Delocation, η-expansion, Fax, Occurrence, Twins**.

• Illusions

Ludics definitely establishes the falsity of certain spiritual principles at work in mathematics, especially in category theory. This should not be taken as an attack against category-theory, not to mention the fact that I perhaps do not know my free-and-bound-variables. Category-theory played an immense role in the disclosure of linear logic, ludics, in the correct understanding of intuitionistic logic… We now realise that it is not that ultimate tool, so to speak, we lose our illusions. But nothing is more helpful than an illusion.

See: **Category, Mistake, Spiritualism**.

• Implicit

Realism is unable to understand the *implicit*. The natural tendency is therefore to reify the implicit in favour of all possible developments. Typically, a finite dynamics will be exchanged with static invariants like in denotational semantics… Mathematically speaking, the method is beyond criticism, but there is something unsatisfactory about it, especially when we push *implicit* towards *potential*.

See: **Answer, Cut-elimination, Cut-rule, Denotational semantics, Existence property, Explicit, Explicitation, Explicit mathematics, Frege, Potential, Proofs as programs, Question, Realism, Reification**.

• Incarnation

In a behaviour **G**, the inclusion between designs generates an equivalence relation, and designs of **G** should be considered up to this equivalence. But, fortunately, each class has a distinguished element: the incarnation $|\mathfrak{D}|$ is the part of a $\mathfrak{D}$ that 'matters' with respect to behaviour **G**, and this part has the good taste to be a design of **G**. The typical example is that of a behaviour 'With' $\mathbf{G} \,\&\, \mathbf{H}$ that is the intersection of its two supertypes $\mathbf{G}, \mathbf{H}$. If

[†] By Thomas Ehrhard.

https://doi.org/10.1017/S096012950100336X

$\mathfrak{D} \in \mathbf{G} \mathbin{\&} \mathbf{H}$, then $|\mathfrak{D}|_{\mathbf{G}\&\mathbf{H}} = |\mathfrak{D}|_{\mathbf{G}} \cup |\mathfrak{D}|_{\mathbf{H}}$, the union being disjoint. This is known as the *mystery of incarnation*

$$|\mathbf{G} \mathbin{\&} \mathbf{H}| = |\mathbf{G}| \times |\mathbf{H}| \qquad (211)$$

This is an equality, not an isomorphism, provided one uses the locative product.

See: **Bergen, Completeness (internal), Delocation, Ethics, Game semantics, Locative product, Money, Mystery of incarnation, Reification, Skunk, Strategy, Subtyping, Winning**.

• Incompleteness

Something is missing. For instance Peano's arithmetic does not prove all arithmetical truths. The question is whether incompleteness refers to an ideal totality (which may be out of reach), that is, a (or several) potential completions. In ludics incompleteness is the pregnancy of the biorthogonal.

Gödel's incompleteness is of an enumerative nature (it is a variation on Cantor's non-denumerability of reals), and applies in the presence of existential second-order quantifiers. Ludics explains this phenomenon by the impossibility of giving a delocated definition of second-order quantification – for reasons of cardinality, Herr Cantor. But more primal forms of incompleteness, of non-enumerative nature appear, typically prenex forms. All these examples correspond to unions $\mathbf{G} \cup \mathbf{H} \neq (\mathbf{G} \cup \mathbf{H})^{\perp\perp}$.

See: **Cantor's diagonal, Coding, Completeness (external), Ethics, Falsifiable, Gödel, Gödel's incompleteness, Non-monotonic logics, Prenex form, Prisoners, Truth**.

• Inconsistency proof

If we take the standard pauper vision of logic, namely that something is true as long it has not been disproved, then our knowledge should decrease. Eventually, what remains? Perhaps nothing... This is why there are so many attempts at proving inconsistency: for instance in March 2000 I received two new refutations of Cantor's diagonal argument.

See: **Armageddon, Cantor's diagonal, Fermat, Pauperism, Unfalsifiable**.

• Infinitary logics

Schütte reformulated Gentzen's second consistency proof (Gentzen 1969b) as a full cut-elimination result in a system of infinitary logic. Although not directly effective, this sort of logic has a good structure, and its proofs are designs in our sense, in other words, infinitary logic is part of ludics.

ω-logic, mainly used by Schütte and his school (Schütte 1960a), corresponds to the specialisation of one quantifier to natural numbers, yielding an infinite rule, with cut-elimination. The ordinal number ϵ_0 is the natural bound occurring in the simplest non-trivial cut-elimination. Later on I introduced $\mathbf{\Pi}^1_2$-logic (Girard 1984), which is more infinite than ω-logic in terms of logical complexity, but more finite (for example, involves fewer codings) on other grounds.

The major limitation of infinite logics is their difficult relationship with finite systems. Typically, actual infinite proofs are recursive, and one must at some moment encode by means of *ad hoc* recursive indices..., which is definitely ugly. Infinite logics have large wings but small feet: 'Ses ailes de géant l'empêchent de marcher'.

Ludics naturally accepts any form of infinitary proof; well-foundedness conditions for ω-logic or $\mathbf{\Pi}^1_2$-logic should be naturally expressed by putting the 'forbidden' infinite

https://doi.org/10.1017/S096012950100336X

branches under the form of appropriate anti-designs. But I have not tried very hard in that direction.

See: **First-order quantifier, Gentzen, Jurassic Park, Ordinal analysis**.

• Intelligence

The problem of machine intelligence, what a question... If we go back to An-fang, where all things start, it might be of interest to note that intelligence barely comes without some form of excess – up to craziness. An intelligent machine would, for instance, be full of wrath, prejudices... unless you are looking for a yes-machine, which is not the point, yes-men would have refused the stone axe. The problem is the power that you are likely to bestow on these would-be intelligent things.

See: **Artificial Intelligence, Cordwainer Smith, Mistake**.

• Intensional

The expression 'intensional' (together with its accomplice 'extensional') is one of those expressions – like 'meta', 'predicative' – characteristic of Jurassic logic[†]. One can get a rough idea of the quality of a paper by the frequency of such words. But what are the possible meanings of 'intensional'?

Spiritism: The most vulgar meaning, which is almost magical, is of some inaccessible soul behind material things: the pedestal table is the extension and the talking spirit is the intension... This is positivistic irrationalism at its apex, for which Occam's razor does wonders.

Forgetting: This is the idea that a crude definition can be refined. For instance, in domain theory the expression 'extensional order' suggests that the stable ordering is intensional, whatever this means... In topology, this would amount to styling pointwise convergence *extensional* and uniform convergence *intensional*. By the way, ludics defined the 'extensional' order $\preceq$ in terms of orthogonality of designs, but the stable order (inclusion) can also be defined by orthogonality:

$$\mathfrak{D} \subset \mathfrak{D}' \Leftrightarrow \forall\, \mathfrak{E} \subset \mathfrak{E}' (\ll \mathfrak{D} \mid \mathfrak{E} \gg = \ll \mathfrak{D} \mid \mathfrak{E}' \gg \,\cap\, \ll \mathfrak{D}' \mid \mathfrak{E} \gg) \tag{212}$$

Hence the distinction is just a matter of knowing what we are talking about.

Introspection: However, one can imagine properties that are purely internal, *introspective* so to speak. This is the case of the *winning* properties of designs, which do not refer to the result of an interaction, but to the interaction itself.

See: **Jurassic Park, Introspective, Logical relation, Meta, Occam's razor, Predicativity, Spiritism, Winning**.

• Interactivity

In computer science, we have the man–machine interaction, which is a great idea. Ludics makes interaction symmetrical, without deciding whether or not one of the partners is smarter than the other.

See: **Barbichette, Computer science, Falsifiable, Lorenzen, Test**.

[†] As in the title: 'Intensional higher-order meta-systems I: predicative part'.

• Interference

Interference is a typical locative phenomenon, due to the fact that two behaviours share certain *loci*. Spiritual logic avoids interference by means of systematic delocations. The good point of delocation is that bad jokes like **true** $\odot$ **true** $=$ **false**, see Section 9.2.2, **false** $\odot$ **false** $=$ **true**, see Section 9.2.3, are impossible, so we can get external completeness. But interference has positive aspects too, typically the new prenex forms come from the sharing of *loci*. To sum up, the challenge is not to banish interference, but to control it[†].

See: **Delocation, Spiritualism, Variables**.

• Intersection type[‡]

Intersection types were introduced about twenty years ago (Coppo *et al.* 1981), to increase the typability power of Curry's type discipline. The intersection type discipline allows us to describe and capture various properties of λ-terms, and it has also a very distinctive semantical flavour. In fact, intersection type assignment systems can be viewed as *finitary logical* definitions of the interpretation of λ-terms in a particular class of models of λ-calculus, the *filter λ-models*. Namely, a typing judgement can be interpreted as saying that a finite element of a model (described by a type) belongs to the interpretation of a given term (Honsell and Ronchi 1992). The importance of intersection types depends on the fact that the most interesting models of λ-calculus, those based on continuous algebraic lattices and coherent spaces, can be described as filter λ-models.

See: **λ-calculus, Subtyping, Torino School**.

• Introspective

I propose to replace the suspect expression 'intensional' by 'introspective', with an opposition to 'extrospective': typically everything defined in terms of some sort of orthogonality, logical relations... is extrospective, but winning conditions are introspective.

See: **Intensional, Logical relation, Maul, Occam's razor, Reducibility, Separation, Winning**.

• Intuitionistic logic

This was the first real 'alternative logic', and was due to Heyting. Technically speaking, the intuitionistic sequent calculus accepts only one formula to the right of sequents, which makes contraction impossible, so that – modulo cut-elimination – one gets the disjunction and existence properties. The only limitation of intuitionistic logic is the absence of a real negation. For the geometric meaning of negation is the exchange left/right, but how can you swap between zones with different maintenances? This is precisely the point of the $\neg\neg$-interpretation.

Linear logic reintroduces the symmetry by systematically forbidding weakening and contraction, not out of some hypocritical 'at most one formula there', but as a general principle. As a result, negation becomes involutive, whereas weakening and contraction become attributes of special connectives – exponentials.

See: **Brouwer, Constructivism, Contraction, Double negation, Exponentials, Heyting, Linear logic**.

[†] According to Curien, the work of Reynolds, and after him O'Hearn and Tennent on the control of interference in the context of shared (imperative) variables, see, for example, (Reynolds 1978; O'Hearn 1999) could be fruitfully revisited using ludics.

[‡] By Simona Ronchi.

https://doi.org/10.1017/S096012950100336X

• Invertibility

Some connectives, typically $\Rightarrow, \wedge, \forall$ in intuitionistic logic, and $⅋, \&, \forall$ in linear logic are invertible, that is, there is only one rule that produces this formula, and the rule can always be applied. This remark, coming from the proof-search community, is very deep: in the presence of a cluster of negative formulas, we can iterate the inversion, so as to get several rules done in a single step, a *synthetic* connective. It turns out that half of the connectives are negative, and the other half, the positive ones, can be handled by means of a dual property, focalisation.

See: **Focalisation, Polarity, Proof-search, Synthetic connective, Time in logic**.

• Isomorphism

In category theory, one defines the notion of a canonical isomorphism, which is quite an achievement. In ludics, we prefer plain equalities, as a matter of taste... But this is not a matter of taste: the prenex forms cannot be explained by isomorphisms, you badly need equalities!

See: **Bergen, Category, Prenex form**.

• Joke

Scientific standards are terrible – one should never joke[†]. To be taken seriously, put people to sleep! In a paper, there should be no funny drawings, it would be a waste of paper. However, full pages of repetitive definitions, of Prussian formalism, are not considered a waste.

A good joke makes you understand a complex methodological point. For instance, when I say that a consistency proof is like an insurance against the explosion of Earth, you get it directly. However, some jokes are dishonest, since they do not respect the very spirit of what they are alluding to, typically the allegory of the prisoners.

See: **Allegory, Copycat, Metaphor, Numerology, Prisoners, Square wheels**.

• Jurassic Park

The dinosaurs are still alive.

See: **Black Mass, Constructivism, Dupond et Dupont, Empty sequent, Foundations, Fundamentalism, Hilbert, Infinitary logics, Intensional, Laplace, Objects and properties, Predicativity, Trinity**.

• Kepler

The computer program BACON of Nobel Price winner H. Simon was given the distances of a planet from the sun together with their period of revolution and it independently rediscovered Kepler's third law, illustrating how far the positivism at work in 'AI' can go. But Kepler's achievement was not to determine a (straightforward) relation between two rows of numbers: it was to figure out *which* numbers should be related, and Kepler's real achievement was actually to find the right question. Incidentally, Kepler stated a fourth law relating planets with perfect polyhedra, and one wonders why this fourth law has not been rediscovered by computer yet, independently of course... The same method works regularly for another astrologer, Nostradamus.

See: **Abduction, Answer, Artificial Intelligence, Astrology, Nostradamus, Question, Sokal**.

[†] Except in the compulsory *opening joke*.

https://doi.org/10.1017/S096012950100336X

• Kreisel

One of the greatest logicians of last century, Kreisel was particularly active in the sixties. With respect to proof-theory, he tried to discard all ideologies: *dixit* van Heijenoort 'Il fait précipiter les grandes formations brumeuses.' Kreisel was a strong opponent of 'so-called consistency proofs'. His *reflection schema*, which internalises the syntax/semantics relation, is typical of his style: the Tarskian nonsense, once formalised, becomes a non-trivial tool. But, in reality, it is cut-elimination that makes things work. With respect to intuitionism, he was less successful, because he tried to formalise too much, so as to sometimes completely miss the point, like in the notorious Saaty volume affair.

See: **Constructivity, Fundamentalism, Lorenzen, Occam's razor, Ordinal analysis, Predicativity, Reflection schema, Saaty volume, Tarskian semantics, Tradition, Weak logics**.

• Kripke model

This is a sort of model for intuitionistic logic based on 'parallel universes'. However, when you change your carriage, your principles change as well: you do not fix a tyre like a horseshoe as you do not feed a horse with gasoline. Intuitionistic logic is not about provability, but about proofs. The same applies to the phase semantics of linear logic; but it is much easier to gesticulate with Kripke models than with phase spaces.

See: **Broccoli logics, Classical model, Gesticulation, Phase semantics**.

• λ-calculus

λ-calculus is a very smart and robust system: only the absence of additives and linear negation prevent us being completely happy with λ-calculus.

See: **Böhm tree, Church–Rosser, Curry–Howard, η-expansion, Fixed point, Forgetful interpretation, Intersection types, Ludics**.

• Lambek calculus

Lambek calculus was the best prefiguration of linear logic (Lambek 1958), although it is restricted to the multiplicative fragment, and written in intuitionistic style.

See: **Linear logic, Non-commutative logic**.

• La Palice

La Palice was famous for the sentence 'Un quart d'heure avant sa mort il était encore en vie': the stupefying remark that, when you are not dead, you are alive, and this stated in 1525, four centuries before Tarski's definition '$A \wedge B$ is true if A is true and B is true'! This is why the Lapalissian notion of truth, known as 'vérité de la Palice' is so famous in France, more famous than the Tarskian one. Typically, if somebody says 'I prefer to be rich and beautiful than poor and ugly', we do not call it a 'vérité à la Tarski' but a 'lapalissade'. To be fair to Tarski, he invented the *meta*, which allows one to distinguish between $\wedge$ and 'and'. To be as strong as the Tarskian truism, La Palice should have written something like 'Un quart d'heure avant sa mort *il remarqua qu'* il était encore en vie'.

See: **Dupond et Dupont, Meta, Pleonasm, Semantics, Tarskian semantics, Truism**.

https://doi.org/10.1017/S096012950100336X

• Laplace[†]

Laplace was a major mathematician, known for his seminal work in Infinitesimal Analysis, Astronomy, Probability Theory. Laplace proposed a paradigm for the mathematical analysis of Physics, the so called 'Laplacian determinism'. From this perspective, systems of (differential) equations could *completely* describe the physical world. More precisely, if one wanted to know the state of the physical world at a future moment, with a given approximation, it is sufficient to know the current state of affairs up to an approximation of a *comparable order of magnitude*. By formally computing a solution of the intended equations, or by suitable approximations by Fourier series (as will be said later), one could deduce (or predict or decide) the future states, up to the expected level of approximation.

Poincaré, as a consequence of his famous theorem on the three bodies problem, proved that *minor variations* of the initial conditions could give *enormous changes* in the final result or, even, that the solutions could depend discontinuously on the initial conditions. Then, predictability, as 'completeness with respect to the world' of suitable sets of differential equations, failed.

About one century later, Hilbert resumed Laplace's program in a different context. He first set the basis for the rigorous notion of a 'formal system', as well as for the distinction between 'theory' and 'metatheory'. He later conjectured that the key system for Number Theory, Peano's Arithmetic (in which he had interpreted Geometry, 1899), was complete with respect to the intended structure of numbers (or that any proposition about the 'world of numbers' could be decided by formal or 'potentially mechanisable' tools).

A few soon reacted to Hilbert's program, such as the 'lone wolf' among Hilbert's students, Hermann Weyl, who (hesitantly) conjectured in Weyl (1918) (notice the date!), the incompleteness of formal arithmetic (at the end of §3). He also firmly stressed in several places that the idea of mechanisation of Mathematics trivialises it and misses the reference to meaning and structures. Besides Weyl (and Poincaré and a few others), Wittgenstein was another thinker who criticised Hilbert's program. For him 'Hilbert's metamathematics will turn out to be a disguised Mathematics' (Waismann 1979), since '[A metamathematical proof] should be based on entirely different principles with respect to those of the proof of a proposition . . . in no essential way there may exist a metamathematics', and . . . 'I may play Chess according to certain rules. But I may also invent a game where I play with the rules themselves. The pieces of the game are then the rules of chess and the rules of the game are, say, the rules of logic. In this case, I have *yet another game*, not *a metagame*', see Wittgenstein (1968, p. 319).

As for formal Arithmetic – the key theory for finitistic foundationalism – these remarks may now be understood in the light of Gödel's Representation Lemma (Gödel 1931): by this very technical result, one may encode the metatheory of arithmetic into arithmetic itself, and thus the 'rules of the metagame' are viewed just as . . . rules of the 'arithmetical game'. Moreover, many proofs that entail the consistency of Arithmetic (for example, the normalisation of system $\mathbb{F}$ and Takeuti's conjecture (Girard 1971)) need a blend

[†] By Giuseppe Longo.

https://doi.org/10.1017/S096012950100336X

of metalanguage and language; there are even some purely combinatorial statements (for example, Friedman's Finite Form of Kruskal's theorem) that provably require the same entangled use of metatheory, theory and semantics, through the 'impredicative' notions involved (Harrington *et al.* 1985). These examples give an indirect confirmation of Wittgenstein's philosophical insights (and Weyl's, as to incompleteness).

Both the Laplace and Hilbert programs – which are strictly parallel and contributed to positivist philosophies in physics and in mathematics – opened the way to very relevant mathematical work: when precise and robust, even wrong programs may lead to extraordinary developments (XIX$^{\text{th}}$ century Analysis, partly motivated by the Laplacian 'calculus of (gravitational) pertubations' or the rigourous notions of mechanisable computation of the '30's and their fall-out: actual computers, as purely 'theoretical' symbol pushers). However, the corresponding incompleteness theorems, Poincaré's and Gödel's or more recent 'concrete' ones such as the two mentioned above, should finally take us away from the underlying philosophies, and also to go further with mathematics. Poincaré's result, for example, is the origin of beautiful and new mathematical theories (the geometry of dynamical systems), where qualitative predictions replace quantitative ones, and the 'mathematical understanding' does not need to coincide with completeness or predictability by formal tools. In mathematical logic we are not yet at a similar revolution, but the basis is being set for breaking the metaphysics of the relevant, but artificial, organisation of the discourse proposed by Hilbert, the theory/metatheory frame. Similarly, we have to overcome the belief that language 'predicates' about the world: the language and structures (of mathematics and physics) are in permanent *resonance*. They construct themselves while singling out concepts and objects, in a permanent tension that requires a parallel analysis of the foundation of these disciplines.

One futher step is now being taken. Twentieth century physics departed from the Newtonian 'causal lawfulness' of nature (and the mysterious instantaneous actions at a distance, such as gravitation) and stressed the geometric structuring of the world: the latter gives the geodesics and provides a unification even with the most recent advances in microphysics. In a sense, it is the structure of space and the location only, that matter. See Longo (2001) for further remarks.

See: **Answer, Dualism, Formal, Hilbert, Jurassic Park, Meta, Predicativity, Realism, System $\mathbb{F}$, Takeuti's conjecture**.

• Laüchli semantics[†]

Läuchli's completeness theorems (Läuchli 1970) for intuitionistic propositional and predicate logic were influential precursors of many modern developments: logical relations, categorical proof theory, full completeness theorems... His semantics interprets provability in the category of hereditary permutations (that is, sets-with-permutations, with equivariant maps) by the existence of an invariant element in the interpretation of each formula.

Technically, it says the following for propositional calculus: 'A $\{\top, \wedge, \Rightarrow, \vee\}$-formula σ

[†] By Phil Scott.

of intuitionistic propositional calculus is provable if and only if for every interpretation of its atoms, its meaning $[[\sigma]]$ contains an invariant element.'

While ultimately this semantics is a semantics of provability, this was the first attempt to characterise a 'space of proofs' abstractly.

See: **Categorical models, Full completeness, Logical relation, Uniformity**.

• Leakage

Leakage is the central problem of logic: the logical rules have been known for centuries, but how do we justify them? In fact it is easy to interpret them in various structures, and since logic is natural, the interpretation works. But each structure usually interprets some extra principle, which is never the same. Ludics is the first non-leaking explanation of logic.

Leakage is very useful, for it indicates where to search, and what to modify; this is why the attitude consisting of replacing the natural definitions by something *ad hoc*, for example, parsimony with exactness to get a full completeness result, is deontologically wrong: when there is a mismatch, one should enlarge the gap, so as to understand what is wrong.

See: **Boots, Curry–Howard, Exactness, Full completeness, Geometry of interaction, Loser, Ludics, Mix, Perishable, Parsimony, Prenex form, Process algebras, Realisability, Xenoglossy**.

• Lemma

A lemma is much more important than a theorem, since it is more likely to be reused. This is why some of the most important theorems of ludics have been called lemmas. Mathematical creation is basically about finding plausible lemmas, then establishing them, possibly with the help of other lemmas. The opposition between Lupin and Holmes is very instructive: they both recognise that reasoning goes backwards, but Lupin insists on intuition, whereas Holmes considers this to be a mechanical activity. Technically, one is with cuts (the lemmas) the other is cut-free.

See: **Abduction, Arsène Lupin, Cut-elimination, Cut-rule, Logic programming, Sense of rules, Sherlock Holmes**.

• Lewis Carroll

The Reverend Dodgson is remembered for *Alice in Wonderland*, not to mention the photos of young Alice Liddell. As a logician he was one of the originators of this idea of logic-as-a-puzzle: 'You are a logician, you must be warped' and is one of the precursors of that monument of vulgarity, Gödel-Escher-Bach.

However, it is not impossible to find some ideas in Lewis Carroll, for instance: Achilles and Tortoise argue about logic. Achilles wants to infer B from A and $A \Rightarrow B$; Tortoise accepts $(A \wedge (A \Rightarrow B) \Rightarrow B)$ but refuses the *Modus Ponens*, that is, refuses B; Achilles tries again with $(A \wedge (A \Rightarrow B)) \Rightarrow B$ and $A \wedge (A \Rightarrow B)$; Tortoise accepts $(A \wedge (A \Rightarrow B) \wedge (A \wedge (A \Rightarrow B)) \Rightarrow B) \Rightarrow B$, but refuses the *Modus Ponens*, *etc*. Eventually, this is a good joke, which may help in understanding cut-free provability.

See: **Artificiality, Cut-elimination, Cut-rule, Gentzen, Jokes**.

https://doi.org/10.1017/S096012950100336X

• Linear logic

Linear logic, (Girard 1987a; Girard 1995b) appeared as a by-product of coherent semantics. The novelty was the emphasis on structural rules, thus individuating linear negation. Linear logic is spiritual, like classical and intuitionistic logics.

See: **Affine logic, Categorical semantics, Category, Coherent space, Completeness (external), Contraction, Curry–Howard, Exponentials, Hypercoherence, Intuitionistic logic, Lambek calculus, Linear negation, Lorenzen, Petri nets, Proof-net, Stability, Structural rules, Substructural logics, Syllogism, Weak logics, Xenoglossy**.

• Linear negation

Stability in coherent semantics says that a question on $f(a)$ can be replaced with a question on a. In other words, every function has an adjoint, a sort of *feedback*. This is the basic meaning of negation, the exchange of questions and answers, input and output. Before the invention of linear negation, an interactive interpretation of logic was beyond reach, since the partners were declared unequal; linear negation is just the involution that exchanges the players.

See: **Coherent space, Double negation, Dualiser, Game semantics, Linear logic, Lorenzen, Material implication, Orthogonality, Pull-back, Stability, Test**.

• Locative logic

If the usual logic is *spiritual*, the new logic should be *temporal*, to stick to religious terminology. However, the word has already been taken for a bleak activity, so the new logic will be locative.

The main question is what to do with the new operations, the new principles. Definitely not turn them into some syntactical chewing gum, at least not systematically. Since internal completeness works for certain locative operations (intersection types, tensor of independent behaviours), it is of interest to try to formalise this part; but the wildest things, the incomplete new connectives, incomplete but *strictly* associative *etc.* should find their use. Perhaps it is possible to study them and to apply them directly, after all other parts of mathematics use incomplete objects and do not seem to suffer too much from the want of syntax. Perhaps we may eventually get the logical counterpart of process algebras.

See: **Abstraction, Connective, Delocation, Locus, Intersection type, Process algebras, Resource, Spiritual logic, Spiritualism, Temporal logics**.

• Locative product

Let us start with the sum $X + Y = X \times \{0\} \cup Y \times \{1\}$. Its main virtue is to socialise X, Y up to isomorphism; it induces remarkable properties summarised by the cardinal equality

$$\sharp(X + Y) = \sharp(X) + \sharp(Y) \tag{213}$$

However, this beautiful spiritual notion is neither associative, nor commutative and without a neutral. The non-trivial achievement of category-theory was to understand the meaning of '. . . -ity up to isomorphism'.

If we define the 'delocations' $\Phi(X) = X \times \{0\}$, $\Psi(X) = X \times \{1\}$, we see that $X + Y = \varphi(X) \cup \psi(Y)$, that is, that the sum can be reduced, modulo delocation to the union… which is truly (that is, strictly) commutative, associative, *etc*., but socialises in a hazardous way: typically, the cardinal equality becomes an inequality

$$\sharp(X \cup Y) \leqslant \sharp(X) + \sharp(Y) \tag{214}$$

The sum is spiritual, the union is locative; one must admit that, if the sum is more useful, the union is more essential, primal.

The same happens with the notion of Cartesian product of sets; the familiar notion, which satisfies the cardinal equality

$$\sharp(X \times Y) = \sharp(X).\sharp(Y) \tag{215}$$

is associative, commutative, *etc*., but only up to (canonical) isomorphism. However, consider the 'locative product' $X \mathbin{\cup\!\!\!\!\!\times} Y = \{x \cup y ; x \in X, \quad y \in Y\}$: this is quite associative, commutative, with $\{\emptyset\}$ as neutral; and, of course, its socialisation power is expressed by the inequality

$$\sharp(X \mathbin{\cup\!\!\!\!\!\times} Y) \leqslant \sharp(X).\sharp(Y) \tag{216}$$

The usual product (or rather an isomorphic variant of the usual product) can be defined from the locative product as $\Phi(X) \mathbin{\cup\!\!\!\!\!\times} \Psi(Y)$.

In ludics one discovers that connectives have more primal locative versions, see the four locative tensors. These connectives are not that bad since they enjoy many properties, typically associativity; the connectives we are accustomed to are just delocations $\varphi(A) \,\square\, \psi(B)$. The delocated versions are more likely to be complete, *etc*., but their properties are only up to isomorphism.

See: **Bergen, Incarnation, Locative logic, Mystery of incarnation, Tensor product**.

• Locus

Literally, locus means 'place, location'. The word refers to the spatial location of a formula. But what could be the address of a formula? The question makes sense as a *relative* one: assume that we are only interested in A, then we can surely locate A where we want, typically at the root $\langle\rangle$ of a tree. Now a cut-free proof of A will make use of subformulas of A and these subformulas can be located inside the subformula tree of A; in this way any formula involved in a proof of A receives a precise location. However, observe that several occurrences of the same subformula receive distinct locations.

We do not necessarily follow syntax in a strict way: we replace connectives with 'synthetic' ones, that is, we apply focalisation, so that we are not seeking immediate subformulas, but rather immediate subformulas of the opposite parity. In the case of simple binary connectives (typically in the multiplicative/additive fragment of linear logic), it is simple to determine the subformulas and to locate them; this becomes more delicate in the case of exponentials (because of the contraction rule) and quantifiers – the exact locations are not that obvious to find.

The most important point is that the negation $A^{\perp}$ of A will share the same address and sub-addresses.

See: **Computer science, Cut-net, Design, Exponentials, Focalisation, Forgetful interpretation, Locative logic, Occurrence, Synthetic connective**.

https://doi.org/10.1017/S096012950100336X

• Logic

Not logic*s*, as if there could be as many logics as pages in a Handbook... By the way logic came as the study of $\lambda\acute{o}\gamma o\varsigma$, the discourse or verb, and is by nature purely internal, something like syntax explained by syntax, as in the mutual transformations of the syllogistic forms *Disamis*, *Celarent*.

See: **Aristotle, Syllogism**.

• Logic plus control

This is an outcome of the drama of logic programming: logic was just seen as a way to pose the problem by an *ad hoc* formalisation, completely external to execution, which was a matter of engineering skill. Control was made necessary because of the inefficiency of a systematic proof-search, and worked *against* logic. In fact logic programming makes sense only if the external (declarative) logic is the same as the internal (procedural) logic. The oxymoron 'logic plus control' expresses the abnegation of such a natural identification.

See: **Control, Logic programming, Oxymoron, Procedural logic**.

• Logic programming

Logic programming uses the paradigm of proof-search, and has been implemented in languages like `PROLOG`. The language was very popular – especially because of the 5th generation program[†] – The main idea of logic programming was to specify logically a question and then to solve it by proof-search. This was a smart idea, but oversold. This style of programming is very efficient at very basic and repetitive tasks – think of the maintenance of a database. But if you logically specify a sorting question, the proof-search mechanism will hardly be as efficient as `quicksort`, not to mention `mergesort`. The reason is that these algorithms rely on smart lemmas, that is, the cut-rule, whereas proof-search is without lemmas, without imagination. Instead of trying to restrict logic programming to its field of excellence, a vehicle that can fly, swim, run *etc.* was created, and did not really work... not to mention the fact that 'control' instructions were added, so as to tamper with the strict obedience to logic, destroying the original *motto* 'pose the problem, `PROLOG` will do the rest'.

Current logic programming is more modest and more efficient. For instance, Andreoli introduced logic programming in linear logic (Andreoli and Pareschi 1991), and in order to improve efficiency, discovered focalisation. See also Miller (1996) and Cervesato and Pfenning (1996).

See: **Closed world assumption, Lemma, Logic plus control, Negation as failure, Proof-search, Subformula property**.

• Logical relation

This is a style of definition that follows the logical connectives, typically 'f of type $A \Rightarrow B$ is XXX iff for all a of type A that is XXX, $f(a)$ is XXX'. The idea is to show that everybody is XXX, by induction on the construction of the objects. The method was successfully used by Tait in Tait (1967) to prove normalisation of Gödel's system $\mathbb{T}$,

[†] The mistake of Japanese industry was to promote a genuine idea, which was presumably immature. However, they did not repeat this mistake: fuzzy 'logic' is not immature – keep a stone warm, it will never produce a chicken.

and later extended by myself to cope with second-order, so as to prove normalisation for system 𝔽 or equivalently Takeuti's conjecture. The additional ingredient is that of a *candidate of XXX-ity* (Girard 1971).

The definition of behaviours in ludics follows the logical relation style, that is, the set of designs in the behaviour interpreting a formula is given in that way, but no other definition is given in that style, as this would conflict with implicit requirements, such as subtyping. By the way, the logical relation style eventually reduces to a definition by orthogonality.

Logical relations should be rejected – except for the definition of connectives. But, in their day, they had immense value: a logical relation paraphrases the logical formula from which it comes. Change a minor point of the logical rules in the *Broccoli* style, and logical relations give up: the evidence, besides their apparent triviality, that they know what is and is not logic. Of course they do not give you the source code, but at least you know that not everything is like everything.

In terms of properties, they are the paragon of extrospection.

See: **Broccoli logics, Heyting's semantics, Intensional, Introspective, Laüchli semantics, Orthogonality, Paraphrases, Reducibility, Takeuti's conjecture, Test, Winning**.

• Lorenzen

Lorenzen was presumably the first to think of a dialectic logic, which is an interactive interpretation of intuitionistic logic. This was a strange idea in the fifties: intuitionism was not that popular in Germany... and we can imagine that the idea of a dialectic interpretation of intuitionism must have displeased the aging Brouwer. So there is a certain originality, up to marginality, in Lorenzen. However, it is difficult to be fair to an enterprise that produced some of the masterpieces of bureaucracy, as typified by Lorenzen (1960), Lorenz (1968) and Felscher (1985). In fact the time was not ripe, and three basic events had not yet occurred, namely:

— The conciliation of constructivity with proof-theory, in which Kreisel played a prominent role.

— The Curry–Howard isomorphism, proofs-as-functions before proofs-as-strategies.

— Linear logic, that is, symmetry in constructivity, which is strictly necessary for a game-theoretic approach.

It is fair to quote Lorenzen, because, as Borges said, every revolution invents its own tradition, and Lorenzen is somewhere on the track. But the half-baked achievements of this School can be assumed to have done more harm than good to the idea of interactivity in logic.

See: **Brouwer, Bureaucracy, Composition of strategies, Dialectica interpretation, Frankenstein, Game semantics, Interactivity, Linear logic, Linear negation**.

• Loser

The main discovery of ludics is that the logical space is never empty, contrary to what happens in the usual leaking interpretations. The typical loser is a design in charge of 'the rule of the game'.

See: **Contraction, Completeness (external), Consensus, Dog, Game semantics, Leakage, Ludics, Paralogism, Winning**.

https://doi.org/10.1017/S096012950100336X

• Ludics

Ludics arose as the study of the interaction between syntax and syntax, typically in cut-elimination. It was necessary to replace syntax with something more geometrical, and this is why ludics lies between syntax and semantics, as a 'semantics of syntax-as-syntax', a monist explanation of logic. The thesis of ludics, which was already present in the programmatic paper Girard (1989b), is that logic reflects the hidden geometrical properties of something.

See: **Analysis and synthesis, Atomic weapon, Category, Dialectics, Empty sequent, Form** *vs.* **contents, Foundations, λ-calculus, Intersection type, Leakage, Loser, Monism, Naturality, Negation as failure, Process algebras, Semantics, Sequent calculus, Syntax, Torino School, Xenoglossy**.

• Martin-Löf system

This was one of the great creations of the seventies (Martin-Löf 1984). Martin-Löf developed his *type theory* out of a very original philosophical analysis, which was completely alien to the Tarskian truisms. The technical originality of the system lies in its basic primitive $\pi \in A$, something like 'π is a proof of A'. The most interesting constructions by far are the connectives $\prod x \in A\ B[x]$ and $\sum x \in A\ B[x]$, the *dependent product* and *dependent sum*.

See: **Constructions, System $\mathbb{F}$**.

• Material implication

If B holds, $A \Rightarrow B$ holds for material reasons, that is, without causality. The attempt at building alternative logics on the mere refusal of this principle, also known as weakening, was a failure, for contraction has to be removed first.

See: **Affine logic, Contraction, Linear Logic, Parsimony, Relevance logics, Weakening**.

• Maul

This is a typical introspective notion: the maul is the process of normalisation. Technically speaking, the maul is obtained by identifying, in a balanced slice, actions κ with the corresponding anti-actions $\widetilde{\kappa}$.

See: **Action, Composition of strategies, Design, Introspective, Normalisation, Slice, Time in logic**.

• Medicine

The typical technique in medicine is to work only with positive information 'As far as we know, one cannot get AIDS by blood transfusion'. Medical truth is therefore what has not been refuted so far. Physicians have, therefore, attempted to impose their views to real science: the pregnancy of formal protocols, the mere idea of falsifiability, *etc.*

See: **Falsifiable, Monsantism, Pauperism, Recessive, Science**.

• Meta

This is an expression used to hide the absence of any mathematical idea; the frequency of this word can serve as a first indication as to the quality of a work (for instance in the expression 'meta logical framework')[†].

The original Greek *μέτά* denoted 'beside, after', and not at all 'before, original'. Typical uses are to be found in 'metamorphosis, metaphor, metastasis', Aristotle's *Metaphysics*

[†] This was not always the case, for instance the old book of Kleene (Kleene 1952) is perfectly respectable.

https://doi.org/10.1017/S096012950100336X

being the book *after* the physics. The expression has invaded all human activities: grammar should now be called meta-language; I recently heard about a meta-movie; tomorrow the constitution will be called the meta-law... We can only hope that the promoters of the meta will be paid back in meta-money!

'Meta' is problematic since it is too ambiguous: typically, take the Tarskian definition of truth – it is currently assumed that *and* exists before $\wedge$. But a more perspicuous analysis, in the ludic style, would say that truth is nothing but a convenient way to reflect properties of formalism: typically the reflection schema only works when the syntax enjoys cut-elimination. It would therefore be more prudent not to try to make a hierarchy between $\wedge$ and *and*, that is, to stick to the 'beside' meaning ('paraphrase') of 'meta'.

Personally, I never use this expression in front of children.

See: **Broccoli logics, Dupond et Dupont, Jurassic Park, Intensional, La Palice, Laplace, Paraphrases, Reflection schema, Self-interpreter, Tarskian semantics, Trinity**.

• Metaphor

Certain basic features of linear logic were very well explained in terms of metaphors such as the gastronomic menu of Lafont. However, one should avoid *allegory*, which consists of replacing the object with its metaphor.

See: **Allegory, Dupond et Dupont, Gastronomic menu, Joke, Münchhausen, Numerology, Obfuscation, Prisoners, Sokal, Square wheels**.

• Mistake

Ludics is based on voluntary mistakes, that is, paralogisms. Paralogicians make mistakes too, but they are involuntary... Just like the failure of the $n+1^{\text{th}}$ offensive of General Joffre in 1915 was involuntary.

As pointed out by Cordwainer Smith, mistakes are essential to intelligence.

See: **Cordwainer Smith, Illusion, Intelligence, Joke, Paralogics, Paralogism**.

• Mix rule

Alternative structural rule

$$\frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta} \tag{217}$$

The rule should not be taken as a particular case of weakening, since the two premises are supposed to contribute to the conclusion. Mix appeared as a by-product of coherent spaces: the tensor product of coherent spaces is not smart enough to refuse this rule. But in ludics, it does not work, even as a paralogism: there is no natural way to mingle designs, and sometimes, in the presence of empty ramifications, no way at all. This rule should be considered as pure leakage, which is not the case of weakening or contraction.

See: **Categorical completeness, Coherent space, Contraction, Full completeness, Leakage, Perishable, Weakening**.

• Money

This is a typical form of abstraction; for instance you are not supposed to use money to light a fire, like ordinary paper. The spiritual (if one can use such a word) content of money is the figure displayed, for example, £20, and money usually interacts on the mere basis of its nominal value. However, there is a locative parameter, the series number,

which is used for security and which therefore belongs to the incarnation. Since two distinct notes of £20 have distinct numbers, that is, distinct locations, the tensor product is a well-defined operation. But, think of gangsters, half a banknote of £20 plus half a banknote of £20 hardly makes £20 unless the series numbers match. A banknote is therefore the *symmetric* tensor product of its halves.
See: **Abstraction, Incarnation, Locative logic, Tensor product**.

• Monism
Monism is the concept in which only one type of object is at work, in sharp contrast to dualism, which is based on heterogeneity, think of matter *vs.* spirit, or syntax *vs.* semantics.
See: **Correctness criterion, Dualism, Form** *vs.* **contents, Ludics, Proof-nets**.

• Monsantism
This applies to those multinationals who, since their products have not yet been proved to be dangerous, consider that they are harmless, and have sent their warships everywhere, in a sort of new Opium War: 'Buy our hormones or[†] die'. It is interesting to note the link with pauperism and more generally non-monotonic logics: something that has not been disproved is correct, and should even be treated 'deductively'. Here the deductive treatment is to force everybody to buy their dubious products.
See: **Faith, Medicine, Non-monotonic logics, Pauperism, Recessive**.

• Münchhausen
Hans Magnus Enzenberger used the well-known story of the Baron of Münchhausen taking himself out of water by ... pulling his own hair, as a metaphor for Gödel's incompleteness. Saying that one cannot fix one's spectacles while wearing them is less poetical, but closer to the real point. Maybe the image can be reused to speak of paraphrases, which is an ambiguous activity.
See: **Dupond et Dupont, Gödel's incompleteness, Metaphor, Paraphrases**.

• Mystery of incarnation
Imagine that Petrucchio has an appointment with Catarina, whom he has not seen for a long time. Since he has no idea as to her present behaviour, he brings with him a bunch of flowers... and a whip. Depending on her behaviour:

A nice girl: Gives the flowers, and forgets the whip.

A bitch: Flowers are good for nothing, uses the whip.

Depends on the weather: Keeps both items.

The three cases correspond to three behaviours $\mathbf{N}, \mathbf{B}, \mathbf{N} \,\&\, \mathbf{B}$, in the first case the incarnation of your gift is $\mathfrak{F}$, in the second case it is $\mathfrak{W}$, in the third it is $\mathfrak{F} \cup \mathfrak{W}$. Hence your reified (material) design of the third case is the Cartesian product of the first two cases.
See: **Incarnation, Locative product**.

[†] The 'or' is not exclusive.

• Natural deduction

The main achievement of Prawitz was the investigation of natural deduction in the mid-sixties: this was the first manifestation of the internal power of language, the first rupture with the time of codings.

See: **Church–Rosser, Curry–Howard, η-expansion, Prawitz, Syntax**.

• Naturality

There are two meanings for 'it is logic': either 'it is warped', or 'it is natural'. Logic should be the most natural thing in the world, and if it is not, this is due to cheap explanations, which are not always honest: for instance, the presentations of Gödel's theorem in the logic-as-a-puzzle style insinuate that, after all, this theorem is a very artificial thing. . .

See: **Artificiality, Ludics, Lewis Carroll, Xenoglossy**.

• Negation as failure

The principle is more reasonable than the notorious closed-world assumption: it says 'if proof-search for A fails, A is false', which can be axiomatised, for failure is something that you can observe, a sort of Daimon. The implementation was monstrous: take a conjunction $A \wedge B$; the algorithm will first try A, and if it fails on A, return a failure, but if we had started with B, perhaps the evaluation loops, and no answer is returned. Is it a reason to introduce non-commutativity here and the endless complications that follow. . . surely not! In fact this nonsense comes from the fact that negation as failure refers to a single evaluation protocol, whereas it should refer to all possible protocols, good or bad, as in ludics. It is surely possible to see the duality of ludics as the civilised version of negation as failure.

See: **Closed-world assumption, Daimon, Halting problem, Logic programming, Ludics, Obfuscation, Proof-search, Square wheels**.

• Non-associative logic

. . . For want of anything worse.

See: **Adjunction, Associativity, Broccoli logics, Gesticulation**.

• Non-commutative logic

I found the first version of non-commutative linear logic in 1987 (expounded by Yetter in Yetter (1990)); the only limitation was that this *cyclic linear logic* excluded the commutative case, which is by far the most important. Long after, Ruet found a way to reconcile commutative and non-commutative connectives, and Abrusci characterised the associated proof-nets (Abrusci and Ruet 2000). By the way, this is simply called *non-commutative logic*, since only a very bad Broccoli logic could be non-commutative without being linear. This logic is spiritual, and its interpretation in terms of ludics is an open question, which may involve additional structure in designs, typically ordered ramifications.

Ludics has a non-commutative tensor, $\olessthan$, which is locative. Four natural tensors arise, namely $\mathbf{G} \odot \mathbf{H}, \mathbf{G} \olessthan \mathbf{H}, \mathbf{H} \olessthan \mathbf{G}, \mathbf{G} \obar \mathbf{H}$, and they collapse into the commutative $\mathbf{G} \otimes \mathbf{H}$ under a spiritual hypothesis: being alien. But these tensors are associative *etc.*, and they are complete under a weaker hypothesis: being foreign. The interest of our non-commutative tensors (and the extra commutative tensor $\obar$) is that they do not require any additional structure, like ordered ramifications. The problem is that they are not spiritual so that one does not see how to connect with the work of Ruet.

See: **Broccoli logics, Lambek calculus, Structural rules, Tensor product**.

• Non-determinism

The fact that logic should eventually be non-deterministic is plain commonsense to me. But it is not because you would like to get closer to, say, quantum mechanics that it works. Ludics, as so far developed, is deterministic, since I was unable to find enough non-determinism in the usual logic, by which I mean something that you can study, not a teratological compilation. However, things are likely to change very quickly, roughly speaking, by dropping coherence in the definition of designs-desseins. Recent semantic developments by Bucciarelli and Ehrhard (Bucciarelli and Ehrhard 2000) go in the same direction.

See: **Dessein, Xenoglossy**.

• Non-monotonic logics

These arise from the idea of making negation commute with provability. Unfortunately, one should have taken negation in its most abstract sense, namely 'duality'. If not, one is bound to confuse 'to know not' with 'not to know' and to run across the main negative results of the thirties. The allegory of the prisoners is a would-be justification for this atrocity.

See: **Antiphrasis, Armageddon, Black mass, Brouwer, Closed-world assumption, Do-it-yourself, Faith, Gödel incompleteness, Herbrand model, Incompleteness, Monsantism, Paralogics, Prisoners, Proofs** *vs.* **models, Science, To know not and not to know**.

• Normalisation

This is usually a variant of cut-elimination. The expression is often used to stress some positive points, such as the unicity of the normal form. For instance, the usual sequent calculus enjoys cut-elimination, but the algorithm is non-deterministic and yields several outputs, whereas normalisation in natural deduction enjoys unicity of the output, the *normal form*, see Zucker (1974) for a discussion. Ludics uses normalisation together with sequent calculus: this is because focalisation individuates an intrinsic timing of logic, with none of those unpleasant commutations that are at work in sequent calculus (and also in the *commutative conversions* for $\exists, \vee$ in natural deduction).

See: **Church–Rosser, Composition of strategies, Convergence, Cut-elimination, Divergence, Maul, Pitchfork, Proof-search**.

• Nostradamus

Abduction has made a lot of progress in recent years; for instance in 1937, a Mr. Ruir was able to predict the Spanish Civil War by an abductive reading of Nostradamus. Now, with computer-aided abduction, it has been possible, in 1998, to read back the death of Princess Diana in the verses of Nostradamus... unfortunately, after the accident took place.

See: **Abduction, Astrology, Kepler, Sokal**.

• Notations

There is something interesting in the choice of notation in a new area. Typically, when I introduced linear logic, several alternative notations were proposed; it is to be remarked

https://doi.org/10.1017/S096012950100336X

that the notation $+$ (or $\oplus$) occurred several times as a substitute for $⅋$. My point against this was that you cannot make an addition distribute over something, for example, the additive conjunction & (which for some respectable reasons one could write $\times$), think of $A + (B \times C) = (A + B) \times (A + C)$... I was answered that symbols are symbols and that one can use them freely: one can call the table 'horse' and the horse 'table'! The refusal to give any special status to essential mathematical features like distributivity betrays a formalistic philosophy for which everything is arbitrary, and nothing is more important than anything else.
See: **Artificiality, Coding, Numerology**.

• Numerology
The tendency towards numerology is implicit in formalism, typically with the abuse of coding. However, people know what they are doing and what they want to hide (usually something awfully *ad hoc*). In the various paralogics coming from so-called AI – say the notorious abduction – there only remains a formal incantation, in which formulas are treated as if they had some esoteric, gnostic meaning. If many papers in logic look like a religious service, here we get dangerously close to the Black Mass.

If you look at the literature, you will find serious people that do not hesitate to publish pages of senseless code – the telephone directory, or worse – as if the ultimate meaning could be there, in symbols, or maybe as if there were no meaning at all.
See: **Abduction, Artificial intelligence, Black Mass, Bureaucracy, Coding, Joke, Metaphor, Notations, Obfuscation, Od -x, Semantics, Spiritism**.

• Obfuscation
Formalism is supposed to clarify things, but it is often used to obscure them. How many papers have been published because the referee quailed before unreadable formulas? Beyond a certain degree of encryption, everything looks the same. This explains also why some do not hesitate to attack colleagues on bracketing, bound variables...
See: **Bergen, Coding, Joke, Metaphor, Negation as failure, Numerology, Od-x, Savoir-vivre, Square wheels**.

• Objects and properties
Objects and their properties are here designs and behaviours. Should they be treated equally?

The dominant positivistic view of logic has had a tendency to treat objects and properties in the same way. If it is the case that a computer process (here: the object) gives its values in a streamlike way, why not extend the streamlike standards to properties? In the same way, the discovery of fixed points in recursion theory, lambda-calculus, induced a notion of fixed point for properties (so-called 'recursive types'), which is highly problematic, to be polite: we have known since Russell's paradox that such a fixed point need not exist. People naturally split into two kinds:

— Those for whom properties are adverse to objects, anti-streamlike and recessive. Concretely, properties are of the form 'consistency', that is, $\mathbf{\Pi}^0_1$ ($\mathbf{\Sigma}^1$ in our classification). This is what I call pauperism: the Hilbert tradition, which is still 'alive' in Jurassic proof-theory.

— Those for whom properties are friendly to objects, streamlike and expansive. Properties are of the form 'inconsistency', that is, $\mathbf{\Sigma}^0_1$ ($\mathbf{\Pi}^1$ in our classification). Paralogicians are usually on this side, something being true for want of a refutation.

But it is definitely impossible to restrict properties to $\mathbf{\Sigma}^0_1$ or to $\mathbf{\Pi}^0_1$. Objects and properties definitely do not possess the same nature.

Of course this could bring us back to dualism, properties as ideas, objects as matter... In ludics, objects are designs, properties are behaviours, and superficially a certain dualism is restored. I do believe that this dualism is due to the limitations of the mathematical language, which must break the unity of things to analyse them. It is easy to understand that, given the right definition of design, one can reconstruct behaviours, but this also works the other way around: a century ago, one basically knew the definition of a formula, and a long and complex process made us understand that these formulas are actually inhabited, and who the inhabitants are. This shows that the idea generates the object, and to some extent the existence of both ideas and objects. But if behaviours generate designs, I hardly see how to explain them as sets of designs such that... Objects and properties, ideas and matter, they run together, and we should not try to separate them.

See: **Armageddon, Behaviour, Design, Dualism, Expansive, Fixed point, Jurassic Park, Paralogics, Pauperism, Recessive, Recursive type, Stream**.

• Obstination

This is the most conspicuous winning condition, since it entails consistency. It says that you are not responsible for the finiteness of the dispute, that is, you are not first to give up. When the two players are obstinate, nobody wins (dissensus): in a behaviour there is an implicit rule of the game because the players play reasonably fairly, that is, they are not systematically obstinate – they can give up.

See: **Barbichette, Dissensus, Dog, Loser, Winning**.

• Occam's razor

One of the favourite expressions of Kreisel, alluding to the elimination of useless hypotheses, seen as a useless beard. An abuse of Occam's razor would be systematic extrospection, that is, to judge things by their use, not by their immanent structure, think of Sherlock Holmes. Of course, the usual logical expressions like 'intensional, meta, ... ' need a serious shaving, which is what we have done to some extent in this monograph.

See: **Kreisel, Intensional, Introspective, Meta, Paraphrases, Sherlock Holmes**.

• Occurrence

Traditionally, this term is used to distinguish two different uses of the same formula, typically in $A \vdash A$, one speaks of two occurrences of the same formula A, as if we were speaking of twin brothers as two occurrences of the same person. The viewpoint of ludics is clear: there is nothing like an occurrence, since behaviours (which correspond to logical formulas) have locations: different occurrences cannot be the same, but they can be isomorphic. The correct written form should therefore be $A' \vdash A''$, where A', A'' are two isomorphic copies of A.

See: **Atomic proposition, Identity axiom, Locus, Twins**.

• Od -x

```
0000000 5365 6520 666f 7220 696e 7374 616e 6365
0000020 2043 6861 6974 696e 2c20 472e 2c20 416c
0000040 676f 7269 7468 6d69 6320 496e 666f 726d
0000060 6174 696f 6e0a 5468 656f 7279 2c20 4361
0000100 6d62 7269 6467 6520 556e 6976 6572 7369
0000120 7479 2050 7265 7373 2c20 3139 3838 2c20
0000140 7070 2e20 3436 2d35 302c 2038 372d 3930
0000160 2e0a 0a00
0000163
```

See: **Bureaucracy, Coding, Numerology, Obfuscation**.

• One

This is the neutral element of the tensor product. The most important point is that 𝔒𝔫𝔢 cannot be freely delocated, hence there is a conflict between harmony and principles. Choosing principles, would amount to giving up the neutrality of 𝔒𝔫𝔢, hence we have preferred to exclude this constant from completeness rather than having instead a mock ‘One’.

See: **Atomic weapon, *-Autonomous category, Boots, Game semantics, Harmony, Tensor product**.

• Operational semantics

This is the sort of expression coming from computer science, and reflecting the want of dynamicity of the usual semantics: people like to oppose denotational and operational semantics. However, the comparison is unfair, for whatever the limitations of denotational semantics, it is globally an exciting area with reasonably high standards. Nothing of the like can be said of operational semantics, which, independently of the real work done by people, does not go beyond the level of mere paraphrases.

Long ago (1987), I wanted to produce an ‘operational semantics’ for linear logic. But Longo convinced me that, with the mathematical structures I had in mind, I would be better off creating a new expression… This was the origin of *Geometry of Interaction*, which is of course an operational semantics, but done within mathematics.

See: **Antiphrasis, Computer science, Denotational semantics, Geometry of interaction, Paraphrases**.

• Ordinal analysis

Originally introduced by Gentzen in his second consistency proof of arithmetic (Gentzen 1969b) (1938), a transfinite induction up to the ordinal $\epsilon_0 = \omega^{\omega^{\omega^{\cdots}}}$ proves the consistency of arithmetic, by means of a restricted form of cut-elimination. As remarked by Kreisel, André Weil’s joke ‘Gentzen, c’est le type qui a démontré la cohérence de l’arithmétique – *i.e.*, l’induction jusqu’à ω – par une induction jusqu’à ϵ_0’ is unfair: in fact Gentzen limits his transfinite induction to very elementary properties (quantifier free), and the gap between arithmetic and the principles used to prove its consistency is not that big – tiny in fact. Nevertheless, Gödel’s incompleteness applies and Gentzen’s consistency proof has no value as such for this reason. Ordinal analysis was continued by the followers of Gentzen, Schütte, *etc*. This was ordinal *Panzerdivisionen* with respect to *Panzerdivisionen* of theories: neither of the two sides was very exciting... A clarification of the *nature* of their relationship would have been more convincing.

See: **Consistency proof, Convergence, Gentzen, Gödel's incompleteness**.

• Orthogonality

Orthogonality is loosely inspired from geometry. It has taken several forms:

Phase semantics: The basic idea was orthogonality of contexts, $\Gamma \perp \Delta$ when $\vdash \Gamma, \Delta$ is provable.

Proof-nets: These involve the orthogonality of permutations (Girard 1988) or partitions in the style of Danos & Regnier.

Geometry of interaction: $u \perp v$ iff uv is nilpotent.

Ludics: $\mathfrak{D} \perp \mathfrak{E} \Leftrightarrow \ll \mathfrak{D} \mid \mathfrak{E} \gg \; = \mathfrak{Dai}$

All these definitions try to capture linear negation, seen as duality. The logical relation style can eventually be reduced to orthogonality, for example, replace '$\mathfrak{F}$ maps $\mathbf{G}$ into $\mathbf{H}$' with $\mathfrak{F} \perp \mathbf{G} \otimes \mathbf{H}^{\perp}$.

See: **Behaviour, Brouwer, Consensus, Convergence, Correctness criterion, Linear Negation, Logical relation, Phase semantics, Proof-net, Test, Type**.

• Oxymoron

This is a figure of speech based on opposition: there is an internal contradiction in 'burning snow', not to mention 'Church of Scientology', or 'military justice'. It is also a very popular figure in logic: for instance, 'philosophical logic' or 'fuzzy logic', in which *fuzzy* (which should be applied to the methods, the results obtained...) is opposed to *logic*, in a sort of *profiterole*. Oxymoron hunting is an interesting activity – recently I stumbled on 'analytical philosophy'.

See: **Antiphrasis, Explicit mathematics, Logic plus control, Philosophical logic, Pleonasm, Recursive type, Semantics, Spiritualism**.

• Parallel or

This was an essential contribution by Plotkin (Plotkin 1977) to the theory of sequentiality:

$$\begin{array}{rcl} P(\mathtt{tt}, y) & = & \mathtt{tt} \\ P(x, \mathtt{tt}) & = & \mathtt{tt} \\ P(\mathtt{ff}, \mathtt{ff}) & = & \mathtt{ff} \end{array} \tag{218}$$

There is no obvious sequential way to execute the algorithm. However, it can be interpreted in Scott domains, with various consequences. Typically, there is no minimum datum responsible for

$$P(\mathtt{tt}, \mathtt{tt}) \quad = \quad \mathtt{tt} \tag{219}$$

There are in fact two minimal choices $(\mathtt{tt}, \emptyset)$ and $(\emptyset, \mathtt{tt})$, and the maintenance of Scott domains forces one to encode the Equation (219) by means of these two choices and also the non-minimal choice $(\mathtt{tt}, \mathtt{tt})$, not to mention the fact that one must explain the relation between the various representations... Berry introduced stability to eliminate this counter-example, though stability is not enough to cope with the Gustave function.

See: **Coding, Coherent space, Gustave function, Pull-back, Scott domain, Sequentiality, Stability**.

• Paralogics

These 'logics' with not even a deductive system were proposed around 1980 by *amateur* logicians. The common idea was to add a principle of the form 'If A is not provable, then conclude $\neg A$', in other words, to make negation commute with provability, an idea that is vaguely reminiscent of the disjunction and existence properties. Adding this principle to current formal systems makes them complete (one of $A, \neg A$ becomes provable), and Gödel's theorem destroys any hope of obtaining a decent formalism for them.

The idea comes from a more general naive computer science attempt: make all programs terminate by means of a 'loop detector'. But not all loops can be detected (this is the undecidability of the halting problem), so programs cannot be 'completed' into terminating ones. By the way, the idea of completing programs or theories is as crazy as the idea of 'completing' an unbounded operator on the Hilbert space.

It is interesting to note that this destruction of the deductive paradigm was supposed to improve classical deduction, just as the similar destruction of the computing paradigm was supposed to speed up computation. . . , but after all communism was supposed to improve bourgeois (classical) democracy.

To sum up, provability does not commute with negation: 'not knowing that' and 'knowing that not' are essentially distinct.

See: **Abduction, Algebraic logic, Broccoli logics, Do-it-yourself, Gödel's incompleteness, Halting problem, Non-monotonic logics, Objects and properties, Paralogism, Perishable, Proofs** *vs.* **models, To know not and not to know, Unbounded operator**.

• Paralogism

This was originally a mistake of logic, and therefore frequent in paralogics. In ludics, there are paralogisms too (typically the daimon, weakening *etc.*), but they are not of the same nature: after all our mistakes are done *on purpose*. They are so to speak 'good mistakes', which are essential to the completion of the logical space, and their discovery was a long and painful process.

See: **Artifical Intelligence, Closed-world assumption, Computer science, Correctness criterion, Dog, Loser, Mistake, Monism, Proof-net, Paralogics**.

• Paraphrases

The fact that 'meta' is almost a pornographic expression does not mean that the expression conveys nothing. After analysing, or shaving if you prefer, I retain only one meaning, that of a paraphrase. The question is now to judge the value of paraphrase:

Very bad: A paraphrase explains nothing, it is a pure pleonasm.

Very good: It might be useful, think of abstract machines influenced by abstract categories, of certain uses of operational semantics. Self-interpreters and the reflection schema are the best positive illustrations of paraphrases, and perhaps the notorious Baron of Münchhausen yields a reasonable metaphor for this activity.

If we agree that a paraphrase explains nothing, we are reduced to the second aspect, that is, to determine whether or not a notion can interact successfully with its paraphrase, without the hierarchisation implicitly at work in the expression 'meta'. Then one discovers that not everything admits a useful paraphrase, typically it will fail for a random Broccoli

https://doi.org/10.1017/S096012950100336X

logic. The usual tool responsible for internalisations, reflections, self-interpretations, *etc.* is cut-elimination, which is another name for internal completeness.

See: **Broccoli logics, Categorical semantics, Completeness (internal), Cut-elimination, Dupond et Dupont, Logical relation, Meta, Münchhausen, Occam's razor, Operational semantics, Pleonasm, Reflection schema, Self-interpreter, Tarskian semantics**.

• Parsimony

This is one of the winning conditions, which states that, up to the fact that my opponent perhaps prevented me from behaving, I have consumed all the loci. In other words, everything created is of some use. This introspective notion reacts against material implication, that is, weakening.

See: **Affine logic, Barbichette, Dog, Dualiser, Exactness, Leakage, Loser, Material implication, Weakening, Winning, Xenoglossy**.

• Partial design

This notion is relative to a given behaviour: $\mathfrak{D}$ is partial in $\mathbf{G}$ when it is included in a (total) design of $\mathbf{G}$. Partiality is the ingredient of bihaviours. Observe that Ω is the only absolutely partial design.

See: **Biethics, Bihaviour, Faith**.

• Pauperism

This is the correct expression for the philosophy of Popper – or at least its popular version: something is true as long it has not been refuted. In other words, a building is safe as long it hasn't collapsed (commonsense would rather tell us that a safe building is a building constructed according to good principles, whatever that means). There is a confusion here between the fact that what we can observe – say on a computer – is streamlike, and the fact that abstract entities should be streamlike (or anti-streamlike). The problem with all those positivists is that they consider theoretical thinking to be the secretion of a sick gland (the brain), the use of which should be minimised by all means.

See: **Abstraction, Armageddon, Dissensus, Expansive, Falsifiable, Inconsistency proof, Medicine, Monsantism, Objects and properties, Recessive, Stream**.

• Perishable

Due to leakage, most semantics are perishable: a good semantics is replaced with a better one, in order to fix some problem, for example, the mix rule. Paralogics are perishable too, one just replaces something bad with something worse.

See: **Artificiality, Categorical completeness, Leakage, Mix rule, Paralogics, Xenoglossy**.

• PER-model

This was originally an unpublished idea of Kreisel around 1958 (**HRO**: hereditarily recursive operations, **HEO**: hereditarily effective (or extensional) operations) and was extended to second order by Troelstra (Troelstra 1973) ($\mathbf{HRO}_2$) and myself (Girard 1972) ($\mathbf{HEO}_2$), that is, to system $\mathbb{F}$. Later on $\mathbf{HEO}_2$ was renamed the 'PER-model'. A type is seen as a set of pure lambda-terms, together with a partial equivalence relation, and terms are interpreted forgetfully. The structure of PER is still present in ludics, think of bihaviours.

See: **Bihaviour, Forgetful interpretation, Quantifier, Realisability, System $\mathbb{F}$**.

• Petri net

In spite of the early recognition by Asperti (Asperti 1987) of the relevance of linear logic to Petri nets, little came of it.

See: **Gastronomic menu, Linear logic**.

• Phase semantics

This is the 'Tarskian' semantics for linear logic, but it is not a Broccoli semantics. A phase space is nothing but a commutative monoid together with a distinguished subset (representing $\perp$). The basic notion is that of orthogonality, $x \perp y \Leftrightarrow xy \in \perp$, and the task of ludics was to replace this orthogonality between elements of an abstract (spiritual) monoid defined by means of an abstract set $\perp$, with an orthogonality between concrete objects, designs, with a 'physical' sense.

In so-called 'substructural logics', nothing like phase semantics survives: people are working with *quantales* whose properties are changed according to the humour of the day, one day Broccoli, the next day Spinach... As shown in Girard (1999b), it is almost impossible to tamper with phase semantics: we cannot change the properties of logic by changing the properties of the monoid, typically, if we drop commutativity, we get... a non-commutative logic, sure, which is also non-associative. Phase semantics is therefore relatively respectable: like coherent semantics, it is part of the picture, a reasonable treason. To see that there is something in it, just look at the work of Lafont (Lafont 1996).

See: **Broccoli logics, Classical model, Design, Kripke model, Gastronomic menu, Orthogonality, Resource, Substructural logics, Treason**.

• Philosophical logic

This has the same problem as with proof-theorists, but the internal clock shows '1600'.

See: **Dualism, Oxymoron, Proof-theory, Relevance logics**.

• Pitchfork

This is what remains of a sequent when the formulas have been forgotten and only their locations remain. A pitchfork is like an electronic interface with plugs (the *loci*) but without the specifications (the formulas). When we plug a handle with a tine, something happens, good or bad, which is expressed by normalisation.

See: **Cut-net, Dessin, Normalisation, Polarity, Reification**.

• Pleonasm

An essential figure of speech, illustrated by La Palice, and more recently by Tarski: it consists of saying the same thing twice, to look more profound. Observe that the antiphrasis 'popular democracy' is made more cruel by the use of pleonasm.

See: **Antiphrasis, La Palice, Oxymoron, Paraphrases, Tarskian semantics, Truism**.

• Polarity

The positive/negative distinction (or synchronous/asynchronous, to stick to Andreoli's terminology) is general. It seems to break on propositional atoms, but this is only an illusion, these atoms refer to an unknown formula, quantified universally or existentially, and such formulas receive a polarity as well, positive for X, negative for $X^{\perp}$, that is, we decide that we are speaking of the unknown formula *of a given polarity* (+ by convention, since negation is available for $-$). Exponentials were a problem for polarisation, and it took me a couple of years to figure out the real solution: $!A$ is in fact $\downarrow\sharp A$, where $\sharp A$ takes the negative A into something, still negative, which is not quite a formula: like the formal atom NH^4, $\sharp A$ only exists in combinations.

Polarity is the main key to ludics – after linear negation, to be fair. First the notion of immediate subformula is changed to 'the closest subformula of opposite polarity'. In this way the implicit associativity of logic becomes an explicit feature of designs – the associativity theorem. Second, remember that sequent calculus is a machinery devised to prove formulas, not sequents. Then the use of focalisation enables us to restrict ourselves to sequents with at most one negative formula: this is true for the conclusion, a sequent with a negative formula comes, through iterated inversions, from sequents that are completely positive, and a completely positive sequent comes, through focalisation, from sequents with exactly one negative formula.

A sequent $\vdash P^{\perp}, \Gamma$, with a single negative formula $P^{\perp}$ can be rewritten as $P \vdash \Gamma$, which is the origin of pitchforks.

Polarity should be related to sequentiality, which deals with the determinism, not of the result of a computation, but of the computation itself. A sequential algorithm is an algorithm with an implicit timing, and eventually, sequentiality is not about determinism, but about time itself: sequentiality is perhaps nothing more than polarisation.

See: **Action, Associativity, Atomic proposition, Classical logic, Critical pair, Exponentials, Focalisation, Invertibility, Gustave function, Pitchfork, Sequantial algorithm, Sequentiality, Shift, Time in logic, Weakening, Xenoglossy**.

• Potential

This is an old logical item (potential *vs.* actual), which gave nothing, due to the want of imagination of the people in charge. Although ludics is written in plain set-theory, it should be apparent that it is not committed to a particular view of infinity – see, for instance, the streamlike features of designs. One of the present limitations of ludics is that it is, however, still possible to handle potentiality by means of the set of its... potential actualisations. I am sorry, if potential means potential, this set does not make sense, but how can I say this, my God?

See: **Implicit, Objects and properties, Stream**.

• Prawitz

I believe Prawitz was the first to state the symmetries of logic precisely: introduction rules of natural deduction 'match' the corresponding eliminations. The symmetry works well for negative connectives $\Rightarrow, \wedge, \forall$, but is more problematic for the positive connectives $\vee, \exists$. Prawitz had deep insights, in particular he thought that natural deduction was more primitive than sequent calculus, which must be manipulated with endless commutation rules.

See: **Natural deduction**.

• Predicativity

This was originally a mistake by Poincaré, giving his own (half-baked) answer to paradoxes 'An object should not be defined in terms of a set containing it'. Predicativity is a pure 'ism', which yielded no output at all in almost a century, think of so-called *predicative* *analysis*, which has been unable to tell the difference between what is predicative and what is not. Like consistency proofs, predicativity sells insurances against the Apocalypse, but at a cheaper price: the predicativist checks the conformity of the system with his principles like others check the conformity of food with the Book. Predicativism is like a Kosher vegetarian restaurant with a unique dish, namely tomatoes without the juice... moreover there is a shortage of tomatoes.

By the way, Kreisel long ago remarked that 'the smallest natural number such that ...' is defined impredicatively, which might be the deepest statement, not about predicativity, but about natural numbers...

See: **Consistency proof, Intensional, Jurassic Park, Kreisel, Laplace, Process algebras, Proof-theory, Tradition**.

• Prenex form

The usual polarities for quantifiers are $\forall$ is negative and $\exists$ positive, and this induces a certain number of commutations of quantifiers with other connectives. To summarise, $\forall$ commutes with negative connectives, that is, we can freely extend or restrict the scope of the quantifier. Traditionally (that is, when positive connectives are involved), we can only enlarge the scope of universal quantification, for example, replace $(\forall d A_d) \otimes B$ with $\forall d(A_d \otimes B)$.

But the *quantificateur nouveau* emerges from ludics, that is, the positive version of $\forall$ and the negative version of $\exists$; these two guys are needed if we want a clean approach to commutation $\forall$/positive and $\exists$/negative... And the essential discovery is that the commutation works without restriction, typically

***Prenex forms do exist in ludics*[†].**

Since this may not be easy to understand, let us take a plain realisability interpretation of second-order logic: $e \circledR \forall X\ (A[X] \vee B[X])$ iff for all $\mathbf{X}$ either $e = 1 * f$ and $f \circledR A[\mathbf{X}]$ or $e = 2 * f$ and $f \circledR B[\mathbf{X}]$... hence $f \circledR \forall X\ A[X]$ or $f \circledR \forall X\ B[X]$... This shows that part of the prenex forms were already there, but if anybody noticed them, they must have ascribed it to one more leakage of the realisability interpretation.

This does not contradict the disjunction property, nor the existence property (as long as we have a concrete quantification on some data type, which is not quite a quantifier). This also shows that our constructive predicate calculi are badly, indeed very badly, incomplete. Moreover, equalities like Equation (101)

$$\forall d(Gd \oplus Hd) = (\forall d Gd) \oplus (\forall d Hd)$$

contradict classical logic. Of course, I could have hidden these quantifiers in the same way that the Victorians used to put pants on donkeys... One must decide between tradition (the usual logical rules) and harmony.

See: **Admissible rule, Associativity, Category, Disjunction property, Distributivity, Existence property, First-order quantifier, Harmony, Implicit, Incompleteness, Interference, Isomorphisms, Leakage, Quantifier, Realisability, Spiritualism**.

[†] In the absence of exponentials.

- Prisoners

Two prisoners with a painted dot on the forehead – black or white – must guess their own colour; they can see each other, but of course they have no mirror. Moreover, they have been told that at least one of them has been painted white. The first guy says 'I don't know', hence the second answers in turn 'white' and is released.

This metaphor has been used to advocate various paralogics. But since it has been impossible to produce any decent logical system, the original metaphor became an allegory: for instance it looks more spectacular with 25 people, the 'Corsican cuckolds' who eventually kill their wives.

Instead of iterating this vulgar joke, one should rather understand what is wrong in the basic case: imagine that the first to speak is a moron who saw a black dot on the other's forehead, but was unable to conclude... Everything relies on the pattern of an unbounded deductive power: if I do not know, I *cannot* know. In other words, one assumes nothing less than the commutation of provability with negation (equivalently the identification between Faith and Daimon), in contradiction to the undecidability of the halting problem and the incompleteness theorem, not to mention the commonsense remark that the absence of a red light is not the same as the presence of a green light.

See: **Allegory, Artificial Intelligence, Closed-world assumption, Daimon, Faith, Joke, Halting problem, Incompleteness, Metaphor, Non-monotonic logics, To know not and not to know**.

- Procedural logic

Classical logic is about reality. But intuitionistic or linear logic are not about an external reality, they are about themselves, about their own rules. This corresponds to the matching between the rules of logic and the logic of rules.

See: **Classical logic, Logic plus control**.

- Process algebras

Milner's theory of concurrency has evolved in the last ten years so as to include the idea of mobility (the π-calculus and its variants). These calculi always leak somewhere – otherwise why so many variants? But this is not a reason to overlook the input of these ideas, compared to, say, ratiocinations about predicativity. In particular, the relationship between ludics and process algebras is potentially of utmost interest; of course, one should try to interpret π-calculi within ludics and not the other way around – for instance how could one define an associative tensor product in the absence of the basic adjunctions? Some obviously locative features of the π-calculi should benefit from a ludic interpretation; on the other hand the non-determinism of these calculi may provide some feedback to ludics.

See: **Computer science, Leakage, Locative logic, Ludics, Non-determinism, Predicativity**.

- Proofs-as-programs

The idea is simple: use proofs (and cut-elimination) to write programs. This is a very good idea, but it gets stalled in practice, for mathematics insists on the *why* whereas computer science is polarised by the *how*. Typically, one can prove that France is connected by showing that every town can be linked to Paris... but the program implicit in this proof (and implemented by French Railways) yields the notorious Web centred on the capital.

Anyway, the idea is interesting, provided one tries to prove less brutal statements that 'can be linked'.

See: **Explicitation, Explicit, Implicit**.

• PROOF-NET

These are surely the most original artifact of the paper Girard (1987a). Proof-nets are graphs that present a non-sequential proof-system. Since they have no explicit timing (unlike designs, they have several sequentialisations), the question of the mere existence of a sequentialisation becomes essential. The answer is known as the correctness criterion for proof-nets of Girard (1987a). This criterion was later simplified by Danos and Regnier (Danos and Regnier 1989). More recently, Guerrini (Guerrini 1999) proved that the criterion can be checked in linear time. There is also an interesting homological interpretation of the criterion by Métayer (Métayer 1994). The importance of proof-nets lies in the early recognition (Girard 1988) that the switchings at work in proof-nets are homogeneous with proofs, sort of paraproofs, and this was the origin of our monist program.

See: **Correctness criterion, Design, Linear logic, Monism, Orthogonality, Paralogism**.

• PROOF-SEARCH

There are two religions as to proofs. The proof-theorist views them as given entities, which are bound to be transformed via cut-elimination. On the other hand the adept of proof-search, who is usually not that educated, but whose viewpoint is sometimes much more creative (not to say more), views the proof as a process: starting with the conclusion, one produces a last rule, then above a selected premise of the last rule, yet another rule *etc*. For the proof-searcher, the proof is never (or exceptionally) completed. The two viewpoints are not irreconcilable, for cut-elimination basically proceeds from the conclusion: only a truncated end-piece of the proof locally matters. In other words, proofs are *streams* and the first difference between proof-search and proof-normalisation is that proof-normalisation considers that the proof is complete from the beginning.

The similarity between proof-search and proof-normalisation has been blurred by heaps of illiterate 'improvements' of proof-search, for example, the notorious *closed-world assumption*. People went on considering proof-search with respect to systems satisfying everything but cut-elimination: in proof-normalisation, something is given implicitly (typically under the form $F(A)$, which involves a cut between F and A), and we normalise it, whereas in proof-search, only the explicit, cut-free part of the system is used, which is why people tried to tamper with cut-elimination. But this is nonsense, since cut-elimination relates the output of different computations: imagine that we program a function F by means of proof-search, and that A evaluates as 0, and $F(0)$ as 7; only cut-elimination can ensure that $F(A)$ evaluates as 7 and not 38.

The proof-search/proof-normalisation distinction[†] is obsolete. With adequate hypotheses, the two activities coincide, at least formally. We have already noticed that the streamlike style makes the two notions of proof identical. The identification becomes especially clear in ludics: a proof-search for A is the same as the normalisation of a cut

[†] At least this is the case theoretically, but the practices are so different!

between A and $A^{\perp}$, the auxiliary proof of $A \vdash$ corresponding to the control on proof-search. Let us just give an example. At some moment you try to prove a formula $A \mathbin{\&} B$, then you ask your opponent 'which side', and if the opponent answers 'B', you proceed with B. But you can imagine that the opponent is unwinding a proof of $A^{\perp} \oplus B^{\perp}$, and that the premise $B^{\perp}$ has been chosen; if you were normalising $A \mathbin{\&} B$ against $A^{\perp} \oplus B^{\perp}$, your cut would be replaced with a cut between B and $B^{\perp}$... By the way, observe that the daimon ✠ has a natural procedural interpretation: if I (proof-searcher) use the daimon, this just means that I give up; if my opponent uses the daimon, this means that he was satisfied with the portion of proof shown to him. Of course this view of proof-search is not the building of a complete proof, but of some parts of it: in another round, the opponent may choose A. The separation theorem says that a real proof is determined by all these partial proofs.
See: **Abduction, Closed-world assumption, Computer science, Cut-elimination, Daimon, Invertibility, Logic programming, Negation as failure, Normalisation, Propagation, Sense of rules, Separation, Stream, Subformula property**.

• Proof-theory
The proof theorist is the guy that proposes Hilbert's Program as the challenge for the new century, a victim of the millennium bug so to speak.
See: **Hilbert, Jurassic Park, Philosophical logic, Predicativity**.

• Proofs *vs.* models
Surely the ultimate achievement of ludics is the concept of design, which unifies the ideas of a proof and a model. $\mathfrak{D} \perp \mathfrak{E}$ means that $\mathfrak{D}$ is a sort of a model (or a counter-model) for $\mathfrak{E}$ and *vice-versa*. Usually one models a theory and not a proof, and this is why naive attempts at the same lead to various tortures of the idea of a classical theory.
See: **Black Mass, Design, Herbrand model, Orthogonality, Non-monotonic logics**.

• Propagation
Let us try to give the procedural interpretation of the ambiguity in context splitting: returning to the example in Subsection 2.3.2, it is fair to say that the first rule did not decide anything as to the dispatching of σ, τ between 3 and 7. After this first (that is, this ultimate) rule, σ, τ look like satellites of the twin stars $\xi * 3$ and $\xi * 7$, without any possible way of telling the difference. Only after a focusing has been made on σ 'above' $\xi * 3$ can we tell that, on the whole, σ belonged to $\xi * 3$, but nothing of the like happens for τ. Proof-searchers would say that the splitting of the context is dynamical; a correct statement, but less exciting than the idea of logical particles that have not yet decided about their ultimate allocation. The condition of propagation says that σ cannot belong to both of $\xi * 3$ and $\xi * 7$; one can imagine a negative message arriving at $\xi * 7$, and saying 'sorry, σ has been consumed somewhere, forget it and its subloci as well'.
See: **Design, Dessein, Dessin, Proof-search**.

• Pull-back
The notion of pull-back, which is the simplest form of an inverse limit, is one of the major inputs of category-theory. It states the existence of a minimum (not minimal, minimum!) witness to various problems. The major limitation of Scott domains was the absence of

a pull-back condition, a limitation that was later fixed by stability, which led to coherent spaces.

See: **Bihaviour, Category, Coherent space, Parallel or, Savoir-vivre, Scott domain, Stability**.

• Quantifier

Any intersection $\bigcap_i \mathbf{G}_i$ of behaviours must be considered as a universal quantifier, and dually any 'union' $\biguplus_i \mathbf{G}_i$ as an existential quantifier. Observe that the union does not witness the copy (better: cannot), so quantification is rather like a comment `%\exists i`. Existential quantifiers need not satisfy the existence property, which is only useful for numerical quantifiers, which are not quite quantifiers.

The logical tradition, which cannot accommodate locative features, made a systematic confusion between universal quantification and infinitary conjunction – with the exception of PER-models, where second-order quantification is a plain intersection. This confusion is legitimate, or at least plausible, in the first-order case, since one quantifies over a denumerable domain, so that we can delocate the 'conjuncts'. In the second-order case, this becomes a nonsense, since there are more behaviours on which we quantify than available loci. As to first-order quantification, in addition to the plain locative treatment there is the possibility of interpreting quantification as $\&_{\mathbb{D}}$, indexed by the domain $\mathbb{D}$ together with a (final) universal quantification over possible $\mathbb{D}$.

See: **Distributivity, Existence property, First-order quantifier, PER-model, Prenex form**.

• Question

This is what we are really after in science: we are not seeking answers, but only the path leading to answers. A good question is one that we can solve, a very good one receives no definite answers, but leaves a methodological track. The AI morons think that answers are more important than questions.

See: **Answer, Artificial intelligence, Astrology, Completeness (external), Cut-elimination, Explicit, Explicitation, Fermat, Full completeness, Hilbert, Implicit, Kepler**.

• Ramification

In a standard logical rule several immediate subformulas interact. The set of their (relative) locations, that is, biases, is a ramification. A ramification occurs either as the indexing set of a positive rule, or as the index of one of the premises of a negative rule.

See: **Action, Bias, Directory**.

• Realisability

The value of realisability is to present, independently of sectarian polemics, an approximation to Heyting's semantics of proofs. Realisability supposes a space with pairing, a naive notion of function, *etc*. Partial recursive indices, pure λ-calculus and... designs satisfy these requirements. In fact, if realisability did not go that far, this must be ascribed to the limitations of λ-calculus (the want of duality).

One assumes that $a \circledR A$ makes sense for atomic A, then

Implication: $(f \circledR A \Rightarrow B) \Leftrightarrow (\forall a(a \circledR A \Rightarrow f(a) \circledR B)$.
Conjunction: $(c \circledR A \wedge B) \Leftrightarrow (\pi^1 c \circledR A \wedge \pi^2 c \circledR B)$
Disjunction: $(c \circledR A \vee B) \Leftrightarrow \exists d((c = 1 * d \wedge d \circledR A) \vee (c = 2 * d \wedge d \circledR B)))$
Quantification: $(c \circledR \forall X)A \Leftrightarrow (\forall C\ c \circledR A[C/X])$

https://doi.org/10.1017/S096012950100336X

The realisation of second-order quantification is in the style of Troelstra, (Troelstra 1973), that is, by intersection. Observe that such a definition validates $(\forall X A[X] \vee B[X]) \Rightarrow ((\forall X A[X]) \vee (\forall X B[X]))$.

See: **Classical logic, Curry–Howard, Leakage, Logical relation, Orthogonality, PER-model, Prenex form, Saaty volume, Semantics of proofs, Test**.

• Realism

There is no doubt that there is a reality, whatever that means. But realism is more than the recognition of reality, it is a simple-minded explanation of the world, seen as made of solid bricks. Realists believe in Laplacian determinism and the absoluteness of time, and deny quantum mechanics: a realist cannot imagine what Audiberti styled as *la secrète noirceur du lait*. In logic, realists think that syntax refers to some preexisting semantics.

Indeed, there is only one thing that definitely cannot be real: reality itself.

See: **Dualism, Implicit, Jurassic Park, Laplace, Reification, Science, Syntax, Truth**.

• Recessive

The more you know, the less you get – typically you need two blue genes to have blue eyes. The positivists, from Hilbert to Popper, the Big Brothers of Monsantism, the various paralogicians, they all agree on that. That is a possibility, but there is a conflict with deducibility, that is, portability, a statement that has not yet been refuted might be of interest to you, but to make a law of it seems delicate, nay criminal. For instance, I remember a very dangerous woman who did not know how to drive; she was beloved by her insurance company – believe it or not, she did not have the slightest crash in years... Surely the company was happy, but could we give her behaviour as an example? Surely not: she stayed alive only because the drivers she met did not behave in the same way.

See: **Consistency, Expansive, Faith, Falsifiable, Fermat, Medicine, Monsantism, Objects and properties, Pauperism, Reification, Σ and Π formulas, Stream**.

• Recursive type

This is a typical oxymoron, for a type is a property, whereas only objects may have fixed points in full generality.

See: **Falsifiable, Fixed point, Pauperism, Objects and properties, Oxymoron, Stream**.

• Reducibility

This was originally a method introduced by Tait in Tait (1967). In the ground case, reducibility asserts normalisation; it is extended to the general case by a logical relation 'f of type $A \Rightarrow B$ is reducible iff for all a of type A that is reducible, $f(a)$ is reducible'. The second-order notion 'candidats de réductibilité' was my first work (Girard 1971). There is a vague smell of this in the treatment of second-order quantification in ludics. Note that reducibility is basically extrospective.

See: **Fixed point, Introspective, Logical relation, Objects and properties**.

• Referee

The usual 'game semantics' interprets formula A by means of a game **G** between the players `Proponent`, who tries to prove A, and `Opponent`, trying to refute A, but there is a hidden third partner, namely the referee in charge of the rule of **G**. But if A really refers to its negation and *vice-versa*, one can hardly see why there should be a third partner.

With an adequate bribing of the referee, you can basically validate whatever you like, using jokes of the form '`Proponent` proposes a rule; `Opponent` says `yes` or `no`'. Behaviours leave no room for the referee.

See: **Atomic weapon, Behaviour, Consensus, Dissensus, Full completeness, Type**.

• Reflection schema

What is syntax, what is semantics? At least the answer is not clear. The idea in Kreisel and Levy (1968) was to internalise, so that the truth of A becomes A and its syntactical properties become arithmetical properties of its Gödel number $\ulcorner A \urcorner$. In particular, the formal implication $\mathrm{REF}_A := \mathrm{Thm}(\ulcorner A \urcorner) \Rightarrow A$ is established in Peano's arithmetic – here Thm refers to provability in a finitely axiomatised subsystem of Peano's arithmetic. The idea is to internalise the brilliant Tarskian evidence that the axioms are true and that the rules of inference preserve truth. But a theorem of Tarski (indeed the first incompleteness theorem of Gödel) makes this impossible: there is definitely no truth predicate. But, using the fact that the proof uses a finite number of axioms and the subformula property, it is possible to actually define a *bounded* truth predicate that can cope with the situation. The parametric version $\mathrm{REF}_{A[x]} := \mathrm{Thm}(\ulcorner A[\overline{x}] \urcorner) \Rightarrow A[x]$ is a particularly useful internalisation lemma, and the reflection schema proves in turn that Peano's arithmetic cannot be finitely axiomatised.

The schema is a clue as to the real meaning of truth, a notion perhaps without sense, but that we can internalise. The reflection schema fails for Broccoli logics: due to the failure of cut-elimination, one can no longer bound the size of formulas in the proof of A. Contrary to a current prejudice, one cannot tamper with the 'meta-universe'.

See: **Broccoli logics, Categorical semantics, Kreisel, Meta, Paraphrase, Occam's razor, Saaty volume, Self-interpreter**.

• Reification

The typical reification is the eventual output of an infinite process. Reification is useful as long as it is compatible with the streamlike viewpoint; beyond that point it only contributes to this ideology of another century, realism. Typical reifications in this monograph are:

Pitchforks: One hardly knows the actual base of a pitchfork, since the context is split dynamically.

Faith: This is the eventual output of a diverging computation.

Incarnation: What is actually used in behaviour **G** depends on the will of your partner.

To be part of an incarnation of a design is streamlike (expansive) ($\boldsymbol{\Sigma}^0_1$), but Faith (that is, divergence) is recessive (anti-streamlike). Eventually you will get the full incarnation of a design, but you will never ($\boldsymbol{\Pi}^0_1$) be able to be sure of the divergence of normalisation. Depending on the way we use them, pitchforks can be streamlike or anti-streamlike.

See: **Dessin, Faith, Implicit, Incarnation, Pitchfork, Realism, Recessive, Σ and Π formulas, Stream**.

• Relevance logics

The idea of rejecting weakening comes from an old criticism concerning material implication and was popular among philosophers; this was implemented, again by philosophers, and led to various relevance logics.

See: **Affine logic, Broccoli logics, Contraction, Material implication, Philosophical logic, Substructural logics, Weakening**.

• Reservoir

This is a set of biases, which is usually infinite. Spiritual operations are handled by means of disjoint reservoirs, for example, even biases/odd biases.

See: **Bias, Delocation, Geometry of interaction, Spiritual logic**.

• Resource

This is the traditional interpretation of linear logic: it is not enough to say *yes* or *no*, you must say *how much*. See Girard (1989b) or Girard (1995b) for basic examples.

The denial of weakening and contraction is basically that two uses are not one use, and that no use is not use. The tendency of ludics (and of geometry of interaction) is to interpret the denial of weakening and contraction by locative constraints.

See: **Gastronomic menu, Geometry of interaction, Locative logic, Phase semantics**.

• Saaty volume

Heyting's semantics of proofs interprets a proof of an implication $A \Rightarrow B$ as a function f from proofs of A to proofs of B. But how do we know that such a would-be function actually does the job? Kreisel (Kreisel 1965) proposed the addition of a second datum, namely 'a proof that the function does the job in a fixed *formal* system'. The same sort of twist is used in the reflection schema, but here Kreisel missed the point. No output at all, only endless quarrels[†], as to the orthodoxy of this idea with respect to Brouwer, the creative subject... As we see in ludics with the definition of $\parr$, Heyting's definition works without any additional control. Whether it yields completeness or not has nothing to do with codings and other acts of will, it is deeply rooted in the structure of biorthogonality, that is, the existence of a complete ethics.

See: **Brouwer, Completeness (internal), Constructivism, Creative subject, Heyting's semantics, Realisability, Reflection schema**.

• Savoir-vivre

Personally, I am not very excited by the theorems of category-theory; when I try to apply one of them in a concrete case, it is simpler to make a direct proof. But is that the point, in other words, are we seeking complicated theorems or are we seeking harmony? In my opinion, category-theory is the best school of socialisation, the *savoir-vivre* of the concepts – think, for instance, of pull-backs: there are those who have heard about pull-backs and can use them, and the others... The limitations of categories are the same as those of savoir-vivre, that is, real good manners are without ostentation.

See: **Category, Harmony, Obfuscation, Pull-back**.

• Schizophrenia

This is the only expression that can be applied to the exclusive habit of logicians who present their systems in two steps: first the syntax, then the semantics, or *vice-versa*. This brings to mind those maniacs who buy books in pairs as if they were socks... This is very convenient, since junk syntax can be explained by garbage semantics, and conversely garbage semantics finds its interest because of the existence of junk syntax.

[†] It is impossible to convey the atmosphere of dogmatism, mutual excommunications... in the name of a rather half-baked idea.

https://doi.org/10.1017/S096012950100336X

It seems possible – at least ludics is pushing very hard in that direction – to speak of logic as a single activity. Other branches of mathematics define their object in a straightforward way, which does not make them any less successful than logic.
See: **Completeness (external), Dualism, Form** *vs.* **contents, Semantics, Soundness, Syntax, Tarskian semantics, Trinity**.

• Scholastics
Medieval scholastics used to interpret syllogisms by other syllogisms, so that *Disamis* could interpret *Celarent* and *vice versa*: this explanation has been thought of as ridiculous mainly because of the sclerosis of the philosophical tradition, at least in logic. But this is definitely more demanding than the Tarskian tradition, which interprets *Barbara* 'Every A is B, every B is C, hence every A is C' by the transitivity of inclusion: $A \subset B, B \subset C \Rightarrow A \subset C$. This is a nonsense, since *Barbara* is presumably more basic than transitivity of inclusion, and Tarskian semantics eventually appears as it is: the real scholastics – in the acceptance of an empty academic activity.
See: **Aristotle, Syllogism, Tarskian semantics, Trinity**.

• Science
Science is an activity that does not deal with reality, in contrast to techniques such as medicine. It is of course better when science is vaguely related to some external phenomenon, but not that much is needed. Wrong sciences, phlogistics, non-monotonic logics, are not wrong because they do not apply: simply because they are ridiculous from an internal viewpoint.
See: **Medicine, Non-monotonic logics, Realism**.

• Scott domain
Scott domains (Scott 1976) – and the contemporary f-spaces of Ershov – were the first step (around 1969) in the direction of an autonomous explanation of logic. Scott constructed a category of topological spaces in which the canonical maps (especially the ones concerned with the function space) were continuous. What is there to say about this? The topological content is bleak, since, in order to cope with the uniform/pointwise dilemma, the spaces are only $\mathscr{T}_0$, and a function in two arguments is continuous when separately continuous. But the real drawback was the overlooking of pull-backs... think of the 'parallel or' of Plotkin.
See: **Artificiality, Categorical semantics, Coding, Coherent space, Denotational semantics, Gustave function, Parallel or, Pull-back, Separation, Stability, Xenoglossy**.

• Self-interpreter[†]
An interpreter is a program (written in a programming language $\mathscr{I}$) for executing other programs (written in a programming language $\mathscr{E}$). It can thus be used to define the programming language $\mathscr{E}$, for example, as an executable specification. $\mathscr{I}$ is then the defining language while $\mathscr{E}$ is the defined language.

In a self-interpreter, defining language and defined language are the same, and therefore, a self-interpreter can execute (a copy of) itself. For example, the programming language

[†] Or meta-circular interpreter, by Olivier Danvy.

https://doi.org/10.1017/S096012950100336X

LISP was first specified with a self-interpreter – in fact this did much harm to the reputation of LISP, because this interpreter elicited a debate very similar to the one about Tarski's definition of truth in logic (Stoy 1977, pp. 181–182). And indeed, as Reynolds has pointed out (Reynolds 1998), in a self-interpreter in direct style where literals are defined as literals, functions as functions, applications as applications, *etc*., the evaluation order of the defining language determines the evaluation order of the defined language. Reynolds, however, also pointed out that a self-interpreter in continuation-passing style (where the defining language is thus a sublanguage of the defined language) makes the evaluation order of the defined language independent of the evaluation order of the defining language, as formalised in Plotkin's independence theorem (Plotkin 1975).

In practice, self-interpreters are used (1) as expressivity tests for the defined language; (2) as means of language extension, by making the defined language a superset of the defining language; and (3) for tracing and debugging purposes: the interpreter is instrumented to maintain extra information about the program it executes.

A similar situation occurs in the area of compiler construction. A compiler is a program (written in a programming language $\mathscr{I}$) translating programs (written in a programming language $\mathscr{S}$) into other programs (written in a programming language $\mathscr{T}$). If $\mathscr{I}$ is a sub-language of $\mathscr{S}$, the corresponding compiler can translate (a copy of) itself.

Partial evaluation (Consel and Danvy 1993) provides yet another example. A partial evaluator is a program (written in a programming language $\mathscr{I}$) specialising programs (written in a programming language $\mathscr{S}$) with respect to part of their input and producing specialised programs (written in a programming language $\mathscr{T}$). Usually, $\mathscr{S}$ and $\mathscr{T}$ are the same language. For example, Kleene's S^m_n-function (Kleene 1952) is a primitive (that is, non-optimising) partial evaluator. If $\mathscr{I}$ is a sub-language of $\mathscr{S}$, the corresponding partial evaluator can specialise (a copy of) itself with respect to a program, yielding a specialiser dedicated to this program.

Partial evaluation provides a nice connection between interpreters and compilers if one considers that a program computes a function from its input to its output: specialising an interpreter with respect to a program has the effect of translating this program from the defined language to the defining language. Therefore a compiled program is a specialised version of an interpreter and a compiler is a partial evaluator that has been specialised with respect to an interpreter. For example, the translation associated with a self-interpreter in direct style is the identity translation[†]. For another example, the translation associated with a self-interpreter written in continuation-passing style (CPS) is a CPS transformation.

Overall, a computer system is constructed inductively as a (finite) tower of interpreters, from the micro-code all the way up to the graphical user interface. Compilers and partial evaluators were invented to collapse interpretive levels because too many levels make a computer system impracticably slow. The concept of meta levels therefore is *forced* on computer scientists: I cannot make my program work, but maybe the bug is in the compiler? Or is it in the compiler that compiled the compiler? Maybe the misbehaviour is due to a system upgrade? Do we need to reboot? and so on. Most of the time, this

[†] As such, a self-interpreter does not define very much indeed, but, as Jones points out, it is useful as an optimality test for a partial evaluator.

https://doi.org/10.1017/S096012950100336X

kind of conceptual regression is daunting even though it is rooted in the history of the system at hand, and thus necessarily finite.

See: **Double negation, Dupond et Dupont, Intensional (introspection), Meta, Paraphrases, Reflection schema, Tarski**.

• Semantics

From the Greek σῆμα, semantics interprets *signs*. Necromancy, numerology... are therefore part of semantics. The best known form of semantics is due to La Palice and Tarski. Very often semantics takes the form of *gesticulation*, that is, giving sense just for the sake of giving sense, for example, in Broccoli logics. But the very sense of semantics is to be found in *treason*, that is, in devious interpretations, such as interpreting *Talibans* as *students*.

I conceived ludics as the semantics of syntax-as-syntax, but soon realised that the word, like 'dialectics', conveys so many implicit meanings adverse to its explicit reading that I decided to shun this expression. In order to style ludics as semantics, I would need half a dozen adjectives like natural, geometrical, monist... and anyway the word would always suggest the existence of some lurking syntax.

See: **Antiphrasis, Behaviour, Broccoli logics, Design, Gesticulation, La Palice, Ludics, Numerology, Schizophrenia, Syntax, Tarskian semantics, Treason, Trinity, Xenoglossy**.

• Sense of rules

Proof-search, proof-normalisation and the actual process of thinking are all directed from conclusion to premise. For instance, I want to prove B, and I figure out a plausible lemma A that entails B, and then I try to prove A, *etc.* This just means that implication works in the direction opposite to what the arrow suggests. But when the rule is eventually written, it is 'from A and $A \Rightarrow B$ deduce B' and not this unbelievable abductologist nonsense: 'from B and $A \Rightarrow B$ deduce A'. These people are good pupils of Conan Doyle, since they believe that the creative process is formal, and they write implication in the wrong direction.

See: **Abduction, Formal, Lemma, Proof-search, Stream**.

• Separation

This is a topological problem: can we separate points? If two points belong to exactly the same topological artifacts, they can be identified without remorse. When this is not the case, the preorder

$$x \preceq y \Leftrightarrow \overline{y} \subset \overline{x} \tag{220}$$

is actually an order, that is, is antisymmetric. This is the weakest possible form of separation: in which case, the topology is styled $\mathscr{T}_0$ 'There is a neighbourhood of x not containing y or a neighbourhood of y not containing x'.

A stronger form of separation is to require every point to be closed, or, equivalently, to require the existence of both a neighbourhood of x not containing y and a neighbourhood of y not containing x. This is the same as requiring the order $\preceq$ to be the equality: such topologies are called $\mathscr{T}_1$.

Finally, the strongest (and current) form of separation is called $\mathscr{T}_2$ (or *Hausdorff*): the two neighbourhoods do not intersect. The topology on Scott domains and the topology on designs are $\mathscr{T}_o$. The duality between *dessins* induces a topology that is not $\mathscr{T}_o$, and the quotient is precisely the *desseins*.

See: **Associativity, Böhm tree, Closure principle, Design, Dessein, Dessin, η-expansion, Introspective, Proof-search, Scott domain**.

• Sequent calculus

This was the major invention of Gentzen (1934) (Gentzen 1969a). The main result is the cut-elimination theorem, sometimes called *Hauptsatz*. Ludics is first of all a reflection on cut-elimination, seen as the real (that is, internal) form of completeness.

See: **Church–Rosser, Completeness (internal), Consistency proof, Cut-elimination, Cut-net, Double negation, η-expansion, Hauptsatz, Hilbert, Ludics, Normalisation, Proof-net, Stoup, Subformula property, Tartuffe**.

• Sequential algorithm[†]

In Berry and Curien (1982) Berry and Curien used the concrete data structures introduced previously by Kahn and Plotkin for modelling sequentiality at all simple types. The basic bricks are *cells* that can be filled by *values*. As shown by Lamarche and Curien (Lamarche 1992; Curien 1994), cells correspond to opponent moves and values to player moves in games. Programs (proofs) are interpreted by 'sequential algorithms' – or strategies. A higher-order cell embodies a question of the form 'what does the program do with input x?' and a higher-order value is either some output value or a request of the form 'this further portion of the input has to be explored'. By design, sequential algorithms are thus streamlike. They were turned by Berry and Curien into a programming language called CDS, in which one can program 'tasters' that distinguish two programs computing the same function, but differently. Sequential algorithms are the first example of an explicitly interactive computational model.

A closely related route to higher-order sequentiality had been opened independently by Kleene (Kleene 1978), in an attempt at curing syntactic flaws in his earlier work on higher-order computability. Kleene had a nice vocabulary, speaking not of moves, cells or values, but of envelopes being handed to oracles. Before his death, Gandy was working along these lines with his student Pani, trying to capture exactly the definable elements of the model. Some of their ideas have been parallelled in $\mathtt{H}^2\mathtt{O}$-games.

See: **Composition of strategies, Game semantics, Hypercoherence, Sequentiality, Stream, View**.

• Sequentiality[‡]

In the denotational semantics of programming languages, sequentiality has been introduced independently by Milner (Milner 1977) and Vuillemin (Vuillemin 1974). A function is sequential, intuitively, when its computational process can be 'linearly scheduled' in time and the Gustave function is the typical stable but non-sequential function. Milner and Vuillemin found a nice characterisation of this idea for 'type 1' functions, that is, functions of type $\mathbb{N} \times \cdots \times \mathbb{N} \rightarrow \mathbb{N}$, which, *a posteriori*, can be understood in terms of focalisation. Sequentiality can be extended to the whole hierarchy of simple types using sequential algorithms (or more generally, strategies in games) or strongly stable functions.

[†] By Thomas Ehrhard.

[‡] By Thomas Ehrhard.

See: **Focalisation, Gustave function, Hypercoherence, Parallel or, Polarity, Sequential algorithm, Synthetic connective, Time in logic**.

• Sherlock Holmes

The guy was able from his positive science of ashes to determine that the murderer was 46, that he had the smallpox and was a retired colonel back from India. The same guy boasted of not knowing the peculiarities of the solar system – an information of no use to him:

(Watson): My surprise reached a climax, however, when I found incidentally that he was ignorant of the Copernican Theory and of the composition of the Solar System [...]

(Holmes): 'You say that we go round the sun. If we went round the moon it would not make a pennyworth of difference to me or to my work.'

Positivists sit between a formal 'bordereau' and a pedestal table... Sir Arthur Conan Doyle ended in spiritism, which is to religion what positivism is to science. One would like to understand this link between the pettiest view of science and the most stupid form of idealism... presumably just a surcompensation.

See: **Abduction, Arsène Lupin, Lemma, Medicine, Occam's razor, Sense of rules, Spiritism**.

• Shift

In terms of games, this connective, which was first introduced in Girard (2000), consists of adding an initial dummy move so as to change polarity. This is, for instance, the point in Anderssen's opening at Chess a2-a3 (he did not want to play White against the notorious Morphy), but, of course, after this first move, you do not get a swapped copy of Chess[†]. The same is true in ludics, the possibility of a daimon replacing the initial dummy move, makes $\updownarrow\updownarrow \mathbf{G}$ non-isomorphic to $\mathbf{G}$. The shift is connected with logical time, that is, change of polarity. Traditionally, this operation has not been represented, and this is why the usual semantics is unable to cope with small objects like the additive neutrals, which would collapse all types in which they occur – if the shift were not there to fill the space.

See: **Connective, Game semantics, Polarity, Time in logic**.

• Σ and Π formulas

In arithmetic, (classical) formulas are classified as $\mathbf{\Sigma}^0_n$, $\mathbf{\Pi}^0_n$, according to the number of alternating numerical quantifiers in their prenex forms, and the nature, existential or universal, of the first one. $\mathbf{\Sigma}^0_1$ formulas '$\exists n A[n]$' are complete (that is, provable exactly when true, in any 'reasonable system'), whereas their negations, the $\mathbf{\Pi}^0_1$ formulas '$\forall n A[n]$', are subject to incompleteness.

In pure logic, numerical quantification is rendered through the second-order definition of natural numbers, due to Dedekind

$$n \in \mathbb{N} \Leftrightarrow \forall X(0 \in X \wedge \forall z(z \in X \Rightarrow z+1 \in X) \Rightarrow n \in X) \tag{221}$$

so that if we translate $\exists n$ as $\exists x(x \in N \wedge \ldots)$ and $\forall n$ as $\forall x(x \in N \Rightarrow \ldots)$, we discover that $\mathbf{\Sigma}^0_1$ formulas are what we call $\mathbf{\Pi}^1$ and, dually, that $\mathbf{\Pi}^0_1$ formulas are $\mathbf{\Sigma}^1$. This classification, $(\mathbf{\Sigma}^1/\mathbf{\Pi}^1)$, which does not emphasise natural numbers, corresponds to the alternation

[†] For instance, the (symmetric of the) Lopez opening is no longer available.

of second-order quantifiers, so that $\mathbf{\Pi}^1$ means 'second-order quantifiers are universal', whereas $\mathbf{\Sigma}^1$ means 'second-order quantifiers are existential'.
See: **Completeness (internal, external), Expansive, Falsifiable, Recessive, Reification**.

• Skunk
The skunk is a negative design orthogonal to nobody, except the daimon. The principal behaviour of the Skunk is the set $\top$ of all negative designs of a given base, hence the Skunk does not look that asocial, everybody lives with him – but possibly not in harmony... In fact the Skunk is the only *material* inhabitant of his lair $\top$: inside $\top$ a design loses all possibilities of recognising the outer world – as it were, the proximity of the Skunk makes one lose any sense of smell. There are positive Skunks as well, which are almost as asocial as their negative prototype.
See: **Daimon, Incarnation**.

• Slice
A slice is a design in which negative rules are at most unary. Slices basically correspond to the multiplicative fragment of logic. When a slice and an anti-slice match, the identification between opposite actions induces a *maul*, which is the intrinsic temporality of execution.
See: **Action, Design, Maul, Temporal logics, Time in logic**.

• Sokal
The pamphlet of Sokal and Bricmont (Sokal and Bricmont 1999) was a healthy thing in the sense that some French thinkers are using science, not just as an occasional metaphor, but as a real allegory: think of Lacan and his notorious (ab)use of knot theory – one comparison in this style is fine, two are already too much,... and what can we say about those psychoanalysts who (tried to) learn knot theory?

The problem is the lurking positivism at work in this criticism. It is easy to catch the philosopher or psychoanalyst who makes an irrelevant scientific quotation. But what about those AI people, who have in each decade since 1950 (got largely funded for and) predicted that in the following one they would construct complete theorem provers, automatic translators between any two languages, complete cooking robots (well, they have been implemented at McDonald's)... when they are not independently rediscovering Kepler's laws on their computer?

The mechanistic and formalist philosophies in Science have caused much greater damage than any 'scientific' hand-waving in the humanities.
See: **Allegory, Artificial Intelligence, Kepler, Metaphor, Nostradamus**.

• Solvable
A λ-term is solvable when it has a head normal form, see, for example, Barendregt (1984). This head normal is the exact analogue of the first positive rule of a design.
See: **Böhm tree, Faith**.

• Soundness
In the syntax/semantics schizophrenia, soundness is the converse of completeness: 'what is provable is true'. Soundness, which is not limited by any restriction is one of the Tarpeian rocks of Broccoli logics: whatever crazy interpretation you devise, the free structure made from syntax will give you completeness – by standard techniques, independent of

the intrinsic qualities of your system (think of the package `Broccoli`). When one tries to prove soundness, one may be asked to restrict oneself to structures enjoying certain properties that are so artificial that only one such structure can be exhibited: the free structure, that is, syntax itself.
See: **Behaviour, Broccoli logics, Completeness (external), Gesticulation, Schizophrenia**.

• SPECIFICATION
In real life, this is the 'how-to' – the guide to 'plug and play'. Specifications reassure you about the eventual behaviour of the product you just bought[†]. Type theory identifies specifications with types, that is, logical formulas. Specifications have so far been combined by means of spiritual connectives.
See: **Abstraction, Subtyping**.

• SPIRITISM
If you are surprised to hear that spiritism originates in the most stubborn positivism, think of the 'magical' deviances of formalism. Anyway, Allan Kardec, the pope of spiritism is known for his 'positive' theory of ghosts.
See: **Intensional, Numerology, Sherlock Holmes**.

• SPIRITUAL LOGIC
In other words, the usual logic, which speaks of some extraneous reality.
See: **Category, Connective, Delocation, Locative logic, Reservoir, Spiritualism**.

• SPIRITUALISM
The locative/spiritual opposition emerged during the composition of this monograph – and became a great divide. Roughly speaking, spirituality is a very important principle, which guarantees modularity – for example, deductive principles – and is very well expressed by category theory

Everything is up to isomorphism.

For instance, in the Hilbert hotel, all rooms are the same, so you cannot tell the difference between $13, 39, 40$. Spiritualism says that mathematics refer to abstractions, invariants *etc*. The value of this principle is immense, but it is completely wrong. There is a physical analogy: when you enter a plane, you are supposed to turn off your cellular phone, because of interferences: that is a smart spiritual move... but interferences do exist – it is precisely the reason behind the interdiction! In the same way, spiritualism avoids logical interferences, but this does not mean that they do not make sense.

Category-theory is entirely concerned with spiritual operations, as is traditional logic, although the idea of a spiritual second-order quantifier is almost an oxymoron. Spirituality induces good properties, typically completeness, and although wrong, remains the most important guideline we can imagine.

However, in real life, interferences are not always that bad. In the same way, spiritualism made us considerably weaken our logic principles, since it was impossible to exploit

[†] For instance: `We guarantee that our car does not use fuzzy logic`.

https://doi.org/10.1017/S096012950100336X

something like the *identity* of objects. Prenex forms, and more generally what I call ‘shocking commutations’ are the positive output of locativism.

See: **Abstraction, Category, Completeness (internal), Delocation, Geometry of interaction, Hilbert hotel, Illusions, Interference, Locative logic, Oxymoron, Prenex form, Spiritual logic**.

• Square wheels

The only way to react to obfuscation is to abstract from petty technical details. Let me tell a story: at the end of a talk about yet-one-more axiomatisation of negation as failure, I explained to the orator that his system was inconsistent, but since the audience was logically illiterate, it was easy for him to get the last word: ‘I say that my axiom system proves ‘. . . ’ and you say it is inconsistent, so it is even better.’ If you keep the discussion on technical grounds, only a couple of people will understand, no way. But you can try to transpose: ‘Pr. Berlusconi once remarked (Berlusconi 1992) that classical wheels are not aerodynamic, and proposed to square them. Somebody objected that a car cannot move with square wheels. . . But Berlusconi said: it is even better, no resistance from the air!’

See: **Joke, Metaphor, Negation as failure, Obfuscation**.

• Stability

The main breakthrough after Scott domains was *stability*, which was discovered by Berry (Berry 1978) and is the basic ingredient of the coherent spaces of Girard (1987a), which led to linear logic. The basic novelty was the discovery, in addition to the usual *extensional order*, of a coarser *stable* ordering. The extensional order is deeply linked (at least in spirit, we have not checked the details) to our $\preceq$, for example, compare

$$f \preceq g \Leftrightarrow \forall a\ f(a) \preceq g(a) \tag{222}$$

with

$$\mathfrak{D} \preceq \mathfrak{D}' \Leftrightarrow \forall \mathfrak{E} \quad \ll \mathfrak{D} \mid \mathfrak{E} \gg \preceq \ll \mathfrak{D}' \mid \mathfrak{E} \gg \tag{223}$$

Stability (written as $f(a \cap b) = f(a) \cap f(b)$ as soon as $a \cup b$ is a clique), says that, to any $z \in f(a)$, we can associate a well-defined finite $a_0 \subset a$ that is ‘responsible’ for the fact that $z \in f(a)$. In other words, the map $z \rightsquigarrow a_0$ defines an adjoint map, a feedback. When these things are put together in the right framework (coherent spaces), this adjoint map is just contraposition, that is, stability leads to linear negation.

See: **Coherent space, Denotational semantics, Gustave function, Linear logic, Linear negation, Parallel or, Pull-back, Scott domain, Sequentiality**.

• Stoup

In sequent calculi, this is a specific place where formulas are baptised. The stoup is a formal approach to focalisation.

See: **Atomic proposition, Focalisation, Sequent calculus**.

• Strategy

A design can be viewed as a strategy, provided we remember that we are playing a game by consensus. Indeed the strategy is rather the *incarnation* of the design.

See: **Atomic weapon, Behaviour, Design, Consensus, Game semantics, Incarnation**.

https://doi.org/10.1017/S096012950100336X

• Stream

A stream can be seen as an infinite sequents of digits that come one after the other, with delays of arbitrary duration, to the extent that we are not sure that the next digit will eventually arrive. The correct approach to designs is to see them as *streamlike*. In the case of a positive base, one is expecting a (preferably, proper) first action. As we are waiting, we can see something like Ω written. Then the answer may come, either something like 'end of stream', that is, $\maltese$, or more interestingly (ξ, I). In the latter case, the stream will proceed, but we must move to one of the $\{\xi * i; i \in I\}$, and ask for a specific ramification J, so we have been replacing ξ with $\xi * i * J$, *etc*. If we think of several parallel channels waiting for various values of J, we are in fact building a negative rule. When we start, all channels display 'Ω', that is, we start with $\mathscr{N} = \emptyset$, but $\mathscr{N}$ keeps growing and growing. In other words, the streamlike approach is about infinite sequences of designs, increasing with respect to inclusion: a design is given progressively, one must think of it as 'in formation'.

All operations on designs must be continuous: this is why normalisation is streamlike, that is, 'proceeds from the conclusion'; this is also why Ω is not a rule. This is not an absolute novelty in proof-theory, see Kreisel *et al.* (1975), see also Girard (1987b), where all cut-elimination is done in this style.

As for the interpretation of streams, proof-normalisation acts as if the proofs were already there, and only given in small bits, due to our limited buffers, and also because our data may be created by normalisation; proof-search presents another interpretation, the proof-searcher is the Sibylla, and surely does not know in advance what the next action will be, she can for instance toss a coin *etc*.

See: **Computer science, Daimon, Dessein, Expansive, Falsifiable, Faith, Objects and properties, Pauperism, Potential, Proof-search, Recessive, Reification, Sense of rules, Sequential algorithm, Σ and Π formulas**.

• Strictness

Category theory provides us with a lot of properties such as commutativity, associativity, and so on, *but only up to isomorphism*. These isomorphisms are never equalities, and, by the way, the mere idea of equality is foreign to categories. In ludics, the basic connectives are strict, and when they are, say, commutative, this means plain equality. These strict connectives admit delocated versions, which are still, say, again, commutative, but only up to isomorphism. Of course category theory requires more than mere isomorphisms, they must be *canonical*... Whatever this means, you will agree that the delocation of an equality could hardly induce a non-canonical isomorphism.

Phil Scott points out that strict categories arise from the study of braid groups... But I still think that the emphasis on strictness is one of the major novelties of ludics, which is not primarily a category-theoretic approach.

See: **Bergen, Category, Commutativity, Connective, Delocation**.

• Structural rules

These are the rules of *exchange, weakening* and *contraction*, which maintain classical sequent calculus. These rules are problematic to various extents, the most powerful and criticisable being contraction. Linear logic is based on the banishing of weakening and contraction, which become the main logical rules of the connective '?'. Exchange is usually accepted, except in non-commutative logic, see Abrusci and Ruet (2000).

The notion of a structural rule does not make sense in the absence of cut-elimination: for instance, you can add the axiom scheme $A \multimap A \otimes A$ so as to obtain the effects of contraction, without declaring contraction! This is the Tarpeian Rock of the bleak area known as 'substructural logics' (Schroeder-Heister and Došen 1993): these systems usually do not enjoy cut-elimination...

See: **Contraction, Linear logic, Non-commutative logic, Substructural logics, Tartuffe, Weakening**.

• Subformula property

The rules of sequent calculus all proceed from simple to complex – with the notorious exception of the cut-rule, which is the point of cut-elimination. In particular cut-free proofs of a formula A are entirely circumscribed in the narrow circle of subformulas of A. The subformula property induces such a drastic simplification of proof-search that proof-search becomes, if not decidable, tractable – this is logic programming.

The subformula property actually states that the possible proofs (cut-free ones, of course) are already 'listed somewhere', that is, there is no way to add new proofs. This is the sense that we give to completeness, namely that 'nothing is missing'. To be more precise, observe that external completeness is traditionally restricted to $\mathbf{\Pi}^1$ formulas, which is the same as the domain of validity of the subformula property – since Takeuti's conjecture extends the scope of cut-elimination to second-order logic, but without keeping the subformula property outside the class $\mathbf{\Pi}^1$.

In system $\mathbb{F}$, current data types are encoded by $\mathbf{\Pi}^1$ formulas, for example, natural numbers by $\mathbf{int} = \forall X(X \Rightarrow X) \Rightarrow (X \Rightarrow X)$. The subformula property enables one to characterise all closed normal terms of this type: they basically are in a natural correspondence with natural numbers (Girard *et al.* 1990). This is why a term of system $\mathbb{F}$ of type $\mathbf{int} \Rightarrow \mathbf{int}$ induces, through normalisation and cut, a recursive function from $\mathbb{N}$ to itself.

See: **Completeness (internal), Cut-elimination, Logic programming, Proof-search, Sequent calculus, System $\mathbb{F}$, Takeuti's conjecture**.

• Substructural logics

The question is whether or not linear logic, which has good properties such as cut-elimination, should be styled 'substructural': yes, linear logic is a substructural logic, but a *degenerated* one.

See: **Affine logic, Algebraic logic, Artificiality, Broccoli logics, Linear logic, Phase semantics, Relevance logics, Structural rules, Tartuffe**.

• Subtyping

If a type is a specification, it is clear that the same object may receive several types: inclusion between types (subtyping) is a natural operation (Abadi and Cardelli 1996). It is important to observe that subtyping is plain inclusion and not some form of isomorphism of one type into another one. In ludics subtyping corresponds to the inclusion of behaviours: when the type increases the incarnation decreases. See the detailed example of Subsection 5.4.2.

See: **Affine logic, Incarnation, Intersection type, Specification, Torino School, Winning**.

https://doi.org/10.1017/S096012950100336X

• Syllogism

Syllogisms are one of the purest, even if very old, creations of the logical tradition, and their father was Aristotle. A syllogism cannot be explained by anything, except another syllogism. This was the activity, now considered as minor, of the scholastic philosophers. An interesting work by Abrusci (Abrusci 2000) interprets the medieval figures in terms of linear logic (more precisely, proof-nets).

See: **Aristotle, Linear logic, Proof-net, Scholastics, Trinity**.

• Syntax

From the Greek τάξις, syntax classifies, puts into order *etc*. Observe that all mathematical activity eventually ends with syntax, that is, syntax only interacts with syntax. Semantics was introduced in order to minimise what could be arbitrary in a purely internal interpretation: syntax should correctly maintain semantics. However, the development of proof-theory since Gentzen shows that syntax has its own regularity, its own immanence (which is not restricted to mere consistency), typically the Church–Rosser property, normalisation. Another style of semantics is therefore necessary, that is, the study of syntax as syntax, and not of syntax as speaking of the world. But the tendency of these second-generation semantics, typically denotational semantics, has been to become another form of reality, though subtler perhaps.

Eventually, ludics has denied the syntax/semantics distinction, by producing artifacts (designs, behaviours) in-between: from syntax it keeps dynamics, finitism (expressed through streamlike aspects), but rejects bureaucracy, codings *etc*. From semantics it keeps the idea of objectivity, plain set-theoretic definitions and geometrical intuitions; it rejects any form of gesticulation.

Of course when I reject syntax, I only reject it as an important philosophical category. Practically speaking, syntax can be extremely important. The building of a useful syntax for ludics is a very interesting *practical* question.

See: **Behaviour, Completeness (external and internal), Consistency, Design, Gesticulation, Ludics, Natural deduction, Realism, Schizophrenia, Semantics, Trinity, Truth**.

• Synthetic connective

It seems that any formula $\Phi(P,Q,R,\ldots)$ defines a connective, but this works only when there is no change of polarity inside the formula. Typically $A\oplus(B\otimes C)$ defines a connective, whereas $A\oplus(B\,\&\,C)$ does not: it is not possible to write complete rules for that connective. Logical time corresponds to the necessary alternation of synthetic connectives.

See: **Focalisation, Invertibility, Locus, Sequentiality, Time in logic**.

• System $\mathbb{F}$

My first work in logic (Girard 1971) is a typed λ-calculus, extending the Dialectica interpretation to full second-order, but better known for the definite solution to Takeuti's conjecture. The structure of system $\mathbb{F}$ is still present in ludics, for example, in the soundness theorem.

See: **Constructions, Dialectica interpretation, Forgetful interpretation, Formalisable, λ-calculus, Laplace, Martin-löf system, PER-model, Subformula property, Takeuti's conjecture**.

• Takeuti's conjecture

Takeuti introduced his sequent calculus $\texttt{G}^1\texttt{LC}$ for second-order logic in 1953 (Takeuti 1953), with a cut-elimination procedure, and he conjectured cut-elimination. The conjecture was first proved by Schütte (Schütte 1960b) and Tait (Tait 1966), in the weak form 'If A is provable, it is also cut-free provable'. I gave the definite answer 'the cut-elimination procedure converges' in (Girard 1971), by means of the *candidats de réductibilité*. The existence of Takeuti's second-order sequent calculus is very important, since one sees that the subformula property, which is the real content of completeness, only holds for $\mathbf{\Pi}^1$ formulas.

See: **Completeness (internal), Laplace, Logical relation, Normalisation, Subformula property, System $\mathbb{F}$**.

• Tarskian semantics

This is a theory of truth:

— $A \wedge B$ is true when A is true *and* B is true.
— $A \vee B$ is true when A is true *or* B is true.
— $\neg A$ is true when A is *not* true.
— $\forall x A[x]$ is true when $A[c]$ is true *for all* c.

and so on... In other words, truth is the quality of what is true. The first time one hears this nonsense, one finds it stupid, but after one learns about a subtle point, namely the distinction between $\wedge$ and *and*: 'you know, *and* is meta'; the truth of A is no longer A, it is in fact meta-A. Tarskian semantics is as exciting as a carrot diet... But after a few weeks of such a diet, nothing tastes better than a carrot: you get addicted, and you dispraise the vulgar minds that say that the King is naked... Tarski's semantics represents the most unimaginative expression of Western rationalism, in sharp contrast to Brouwer's approach. The usual schizophrenic presentation of logic must be ascribed to his influence.

See: **Broccoli logics, Brouwer, Dupond et Dupont, Kreisel, La Palice, Meta, Pleonasm, Schizophrenia, Scholastics, Semantics, Truism**.

• Tartuffe

The incredible success of sequent calculus can be measured by the fashion to formulate junk logic (without cut-elimination) in sequent calculus. The authors forget (or rather know too well) that the heavy straightjacket of sequent calculus is only justified by the beautiful reward of cut-elimination. The austerity of the style implicitly suggests cut-elimination, without stating it, so there is no possible complaint. This sort of attitude is reminiscent of Molière's Tartuffe: a crook disguised as a monk.

See: **Cut-elimination, Do-it-yourself, Sequent calculus, Structural rules, Substructural logics**.

• Temporal logics

This is the bureaucracy of time so to speak, useful, but so bleak... For the temporal logician, time is a secretion of clocks.

See: **Locative logic, Slice, Time in logic**.

• Tensor product

In the case of overbooking at the Hilbert hotel – one group reserving rooms $1, 12, 13$, the other reserving rooms $7, 13, 21$ – ludics considers four locative protocols:

https://doi.org/10.1017/S096012950100336X

$\mathfrak{D} \olessthan \mathfrak{E}$: Rooms $7, 13, 21$ are given to $\mathfrak{E}$, and $\mathfrak{D}$ gets $1, 12$.
$\mathfrak{D} \ogreaterthan \mathfrak{E}$: Rooms $7, 21$ are given to $\mathfrak{E}$, and $\mathfrak{D}$ gets $1, 12, 13$.
$\mathfrak{D} \odot \mathfrak{E}$: Due to the conflict for room 13, the hotel is closed.
$\mathfrak{D} \obar \mathfrak{E}$: Rooms $7, 21$ are given to $\mathfrak{E}$, and $\mathfrak{D}$ gets $1, 12$; room 13 becomes a common room, and, by the way, the hotel keeper installs the most generic inhabitant, the Skunk.

Of course, these four protocols differ only in the maintenance of the disputed room 13. When there is no conflict, the four definitions collapse into a single case, the tensor product $\mathfrak{D} \otimes \mathfrak{E}$, which can be, for instance, achieved with delocation… but nobody will get the beautiful room 13.

These four tensors are associative, but this is not the point: they do have adjoints, $\mathfrak{F}[\mathfrak{A}], [\mathfrak{A}]\mathfrak{F}, (\mathfrak{F})\mathfrak{A}, \{\mathfrak{F}\}\mathfrak{A}$, respectively, which is why the behaviours constructed from the four protocols are associative.

See: **Adjunction, Associativity, Boots, Hilbert hotel, Locative product, Money, Non-commutative logic, One.**

• Test

How can we change plain realisability into an interactive version? The answer is simple, just say that a Ⓡ A exactly when a passes a certain number of *tests*.

Implication: In order to test that f Ⓡ $A \Rightarrow B$ I must produce a realiser a Ⓡ A, together with a test θ for B: $f(a)$ must pass the test θ. In other words, a test for $A \Rightarrow B$ is the pair of a realiser for A and a test for B.

Conjunction: If I want to test a would-be realiser $c = (\pi_1 c, \pi_2 c)$ of $A \wedge B$, I can either test the first component, or test the second component. I conclude that a test for $A \wedge B$ is a test for A *or* a test for B. Observe that the notion of test is subtler than the classical notion of refutation (counter-model): a counter-model refutes A or B, that is, may refute both, whereas the test attacks A (left) or B (right), but not both of them… In particular, a test for $A \wedge A$ is a pair (θ, i), where θ is a test for A to be applied against $\pi_1 c$ if $i = 1$, against $\pi_2 c$ if $i = 2$.

Disjunction: If I want to test a would-be realiser $i * c$ of $A \vee B$, I have to prepare two tests: one in case c pretends to realise A $(i = 1)$ and one in case it pretends to realise B $(i = 2)$. A test for $A \vee B$ is therefore a pair (θ_1, θ_2) of a test for A and a test for B.

We can summarise our analysis by means of the (temporary) symbol A^t for tests on A:

$$(A \wedge B)^t = A^t \vee B^t \qquad (A \vee B)^t = A^t \wedge B^t \qquad (A \Rightarrow B)^t = A \wedge B^t \tag{224}$$

and these formulas are reminiscent of the familiar De Morgan laws of classical logic:

$$\neg(A \wedge B) = \neg A \vee \neg B \quad \neg(A \vee B) = \neg A \wedge \neg B \quad \neg(A \Rightarrow B) = A \wedge \neg B \tag{225}$$

This analogy suggest an identification between A^t and $\neg A$, together with a duality based on the analogy:

Test for A $\sim$ ***Realiser for*** $\neg A$**.**

In fact this does not work, since two readings of $\wedge$ are competing, and only linear logic can separate them. Eventually, everything is fixed – I mean looks plausible – when the connectives are given their linear meaning:

$$(A \,\&\, B)^t = A^t \oplus B^t \qquad (A \oplus B)^t = A^t \,\&\, B^t \qquad (A \Rightarrow B)^t = !A \otimes B^t \tag{226}$$

Of course A^t is bound to become $A^\perp$, and our realisers/tests duality now becomes a proofs/counter-proofs duality. The only problem is to formulate 'a passes test θ' in the right way, and, if possible, symmetrically. This is all the achievement of designs: $\mathfrak{D} \perp \mathfrak{E}$ precisely means that $\mathfrak{D}$ passes test $\mathfrak{E}$ or that $\mathfrak{E}$ passes test $\mathfrak{D}$, a consensus.

See: **Consensus, Correctness criterion, Interactivity, Logical relation, Linear negation, Orthogonality, Realisability**.

• Time in logic

The intrinsic temporality of logic lies in the study of the permutations of rules. Now a cluster of rules of the same polarity can be performed as a single rule, by invertibility in the negative case, by focalisation in the positive case. This is still true of a positive cluster followed by a negative cluster, but not the other way around. By symmetry we must consider that the intrinsic clock of logic is the change of polarity (our connective $\updownarrow$).

Concretely speaking, a double tensor product $A \otimes (B \otimes C)$ can be considered as a single ternary connective. But a connective like $A \parr (B \otimes C)$, which involves a change of polarity, cannot receive sequential (that is, timable) rules, in other words, it is not synthetic. To see this, consider the proof

$$\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A} \qquad \dfrac{\dfrac{\dfrac{}{\vdash B^\perp, B} \qquad \dfrac{}{\vdash C^\perp, C}}{\vdash C^\perp, B^\perp, B \otimes C}}{\vdash C^\perp \parr B^\perp, B \otimes C}}{\vdash A^\perp \otimes (C^\perp \parr B^\perp), A, B \otimes C}}{\vdash A^\perp \otimes (C^\perp \parr B^\perp), A \parr (B \otimes C)}$$

which admits no essential permutation. From below, the rules are done in the order $\parr, \otimes, \parr, \otimes$, and right, left, left, right: the two right rules cannot be performed 'at the same time'. In other words, the only compound formulas that define (synthetic) connectives are those without change of polarity.

See: **Focalisation, Invertibility, Maul, Polarity, Slice, Sequentiality, Shift, Synthetic connective, Temporal logics**.

• To know not and not to know

This is the basic distinction coming from incompleteness, undecidability. This distinction is negated by paralogics and metaphors like the story of the prisoners.

The difference between to-know-not and not-to-know is precisely the difference between negative information and the absence of positive information, that is, the difference between $\mathfrak{Dai}$ and $\mathfrak{Fid}$. Since everything interesting lies in between, one can imagine the bleakness of paralogics.

See: **Daimon, Faith, Non-monotonic logics, Paralogics, Prisoners**.

• Torino School

The Torino School (Coppo, Dezani, Honsell, Ronchi and Venneri) introduced intersection types in order to improve the logical analysis of λ-calculus. However, these types always stayed apart from logic, for want of a clear status, typically the absence of categorical semantics *etc*. Ludics gives a status to intersection types, moreover, in the negative case,

https://doi.org/10.1017/S096012950100336X

intersection types include the additive conjunction &. Intersection types are the most general form of a quantifier.
See: **Category, Intersection type, λ-calculus, Locative logic, Ludics, Subtyping**.

• TRADITION
Even if this looks like a provocation, I would like to stress my respect for tradition; which is surely not the respect due to the dead, but the respect due to the living[†]. The situation in proof-theory is delicate, since on the one hand the fossil old guard refuses to update, while on the other hand a bunch of hooligans tries to radically change logic without even knowing the basics. Between Predicatology and Abductology, it should be possible to find a moderate position 'Se vogliamo che tutto rimanga come è, bisogna che tutto cambi[‡].'
See: **Abduction, Jurassic Park, Kreisel, Paralogics, Predicativity**.

• TREASON
This is the only reasonable meaning of 'semantics'. It consists of devious interpretations; classical model-theory has been excellent at this. Phase semantics, coherent semantics, Geometry of Interaction are semantics in that sense: they help to understand, but they are not the real thing. Ludics discloses the real thing (designs), hence ludics is no longer semantics.
See: **Coherent space, Geometry of interaction, Phase semantics, Semantics**.

• TRINITY
$A \wedge B$ is true when A is true *and* B is true. $\wedge$ is the Syntax, '*and*' is the semantics, and since you might imagine that there is nothing in this definition, we have the Meta: *and* is not quite $\wedge$, it is meta-$\wedge$... Just like the Christian God comes as three-in-one, Logic has its own Trinity, namely, Semantics (the Father), Syntax (the Son or Verb) and Meta (the Holy Ghost). Many logical papers look like a religious service.
See: **Black Mass, Completeness (external), Jurassic Park, Meta, Schizophrenia, Syntax, Semantics**.

• TRUISM
According to Tarski, 'true' is the essence of *truth*. And *vice-versa* truth is the quality of what is true.
See: **La Palice, Pleonasm, Tarskian semantics, Truth**.

• TRUTH
This is usually defined by reference to an external preexisting reality. But the truth of A turns out to be A, and we need all the perversity of Metatarski to get something out of it. In the twenties, people had a nice definition of truth, namely A is true when it is formally provable. And when Gödel realised that he could formalise this notion in arithmetic, he attempted to prove a contradiction, by means of a mere imitation of Cantor's diagonal argument. By the way he succeeded, but this was not a contradiction, just the fact that truth could not be defined that way, in other words, incompleteness.

However, I think that this idea from the twenties is correct, provided we replace 'formal provability', which refers to something too narrow (syntax) with provability, but not in a

[†] Kreisel used to contrast rigour with *rigor mortis*.
[‡] Tomasi di Lampedusa, *Il Gattopardo*.

https://doi.org/10.1017/S096012950100336X

given system, but in a system that we do not yet know, that is, in an expanding formalism. Of course, the expansion could turn out badly, and we have to find some guidelines; consistency (the poor man's immanence) is definitely not enough, something as strong as cut-elimination should be required. It is impossible to describe this expanding formalism, and surely the old-style '*Panzerdivisionen* of theories' miss the point. Behaviours, defined by orthogonality, are completely alien to all the recursive pathologies of the progressions of our (grand-)fathers. So a formal proof is defined as an inhabitant of a behaviour. However, one should pay attention and not admit any design to the status of would-be formal proof, which is the point of winning. A behaviour is true when it harbours a winning design, false when its negation is true. In particular, a behaviour can be neither true nor false.

Although I define truth in this text, and, moreover, internally, I consider it as a way of speaking: truth is no more a true notion than reality is real.

See: **Behaviour, Consistency, Cut-elimination, Incompleteness, Realism, Syntax, Truism, Gödel, Realism, Winning**.

• Twins

Are twins distinct persons or distinct occurrences of the same person? Anyone who professes the latter opinion is likely to be kicked in the ass – two kicks, not two occurrences of the same kick.

See: **Atomic proposition, Delocation, Fax, Identity axiom, Occurrence**.

• Type

If designs were just old λ-terms, the only possibility would be to define a behaviour as a set of designs, and this sort of definition would leave an incredible freedom to the referee, with the result that not a single non-trivial result on behaviours could be stated. But what is a logical formula, or, better, a type? When I give type $\mathbf{A}$ to a design $\mathfrak{D}$, I mean that, as long as I follow the typing rules, everything will work nicely: in particular, if $\mathbf{A}$ is $\mathbf{B} \multimap \mathbf{B}'$, then $\mathfrak{D}$ applied to any $\mathfrak{E}$ of type $\mathbf{B}$ yields through normalisation a design $(\mathfrak{D})\mathfrak{E}$ of type $\mathbf{B}'$. The most important thing is not that $(\mathfrak{D})\mathfrak{E}$ is of the right type, but that it has a normal form. For, if $B' = C \multimap C'$, we can, in turn, analyse the property that $(\mathfrak{D})\mathfrak{E}$ is of type $\mathbf{B}'$, and we see that we are bound to look at all $((\mathfrak{D})\mathfrak{E})\mathfrak{F}$ *etc.* We are trying to feed $\mathfrak{D}$ with all its arguments. This is not usually possible, but imagine that we can actually reach the point where all arguments have been given, then we should understand the ultimate mystery of typing. Technically speaking, the only property that all applications $(\ldots(\mathfrak{D})\mathfrak{E}_1\ldots)\mathfrak{E}_n$ share is the existence of a normal form. So let us use our linear logic background, and write $\mathbf{A}$ as $\mathbf{A}' \multimap \perp$, where $\mathbf{A}'$ is the type corresponding to the negation of $\mathbf{A}$: the condition ensures that, given $\mathfrak{E}$ in $\mathbf{A}'$ (whatever that means) $\mathfrak{D} \perp \mathfrak{E}$, and nothing more. Hence, if I fill the type $\mathbf{A}$ with all designs in $\mathbf{A}'^{\perp}$, this bigger type will enjoy all constraints already satisfied by $\mathbf{A}$. Since *nature abhors a vacuum*, we must consider that $\mathbf{A}'^{\perp}$ is the real type. Concretely, this mean that each type is the orthogonal of something (a set of designs, seen as *constraints*), and, using the common background of elementary mathematics, that a type is a set of designs equal to its biorthogonal. By the way, this forces $\mathbf{A}' = \mathbf{A}^{\perp}$, which the notation was heavily suggesting anyway.

See: **Behaviour, Design, Orthogonality, Referee**.

• Unbounded operator

In functional analysis, an unbounded operator is a partial operator whose graph is closed. By the closed graph theorem, if the operator happens to be total, it becomes an ordinary (bounded) operator. In particular, a really unbounded operator (that is, one whose norm is infinite) is partial, not out of a forgotten totality, but intrinsically.

See: **Gödel's incompleteness, Halting problem, Paralogics**.

• Unfalsifiable

Very recently (March 2000) I received a copy of an Email from a professional AI man, who was so happy, so very happy, he could not help it... to learn that Gödel's theorem had eventually been shown to be wrong. Of course mathematics was immediately inconsistent because of this discovery... The poor chap had not noticed that Gödel's theorem cannot be disproved, since it is stated as 'If Peano's arithmetic is consistent, then...'.

There is a big difference between a refutation of Fermat and a refutation of Gödel: in both cases mathematics becomes inconsistent for these theorems have been proved. But Fermat's theorem becomes definitely wrong, whereas Gödel's theorem is proved on even more solid grounds. This is an example of an unfalsifiable result, and, according to Pauperism is of no scientific value. But perhaps it is Pauperism that has no value at all...

See: **Armaggedon, Artificial Intelligence, Dissensus, Falsifiable, Fermat, Gödel's incompleteness, Incompleteness, Inconsistency proof, Pauperism**.

• Uniformity

This winning condition expresses alikeness, for example, invariance under permutation. The subtle point is that uniformity relies on non-uniform objects, just like virtue relies on sinners.

See: **Barbichette, Bihaviour, Dog, First-order quantifier, Laüchli semantics, Loser, Winning**.

• Variables

These are the typical locative artifacts, which have been completely mistreated by the logical tradition. The Tarskian credo says that a variable refers to some object, whereas it is only a location – typically the location of its binder. Computer scientists were more perspicuous than logicians, since they have spent a lot of energy on the problem of renaming variables, think of De Bruijn indices: in computer science, a variable is an address in memory, and no spiritual principle can make two addresses equivalent (even if some commands like `defrag` are of a spiritual nature, and user interfaces try to minimise the allocation problems). The typical interference of variables is known as *capture*, see the difference between $\sum_i a_i.b_i$ and $\sum_{ij} a_j b_i$. The renaming of i into j avoids an interference.

The theory of free and bound variables is still a typical topic for the early morning lecture the day following a meeting banquet. But this is a deep subject, provided one accepts capture as a natural phenomenon like interference. By the way, interference occurs in a computer.

See: **Atomic proposition, Delocation, Interference, Tarskian semantics**.

https://doi.org/10.1017/S096012950100336X

• View

Views were introduced in the $\mathtt{H}^2\mathtt{O}$-*games* of Hanno (Nickau) (Nickau 1994) and Hyland and Ong(Hyland and Ong 2000).

See: **Chronicle, Game semantics, Sequential algorithm**.

• Weakening

Contraction contradicts linearity by introducing quadratic dependencies. Weakening

$$\frac{\vdash \Gamma}{\vdash \Gamma, A} \tag{227}$$

introduces fake dependencies, that is, affine functions, which are almost linear. In denotational semantics, weakening is not compatible with the existence of an involutive negation… but in ludics, weakening works without the slightest problem, which is a miracle due to polarisation (the weakening rule just written can be restricted to *positive* formulas A). The possibility of weakening is essential in the completeness proof of the tensor product, especially the Projection Lemma (Theorem 19, p. 354). Weakening is eventually declared losing: parsimony is against weakening.

See: **Affine logic, Contraction, Critical pair, Dog, Dualiser, Exactness, Leakage, Material implication, Mix rule, Parsimony, Polarity, Relevance logics, Structural rules, Xenoglossy**.

• Weak logics

People sometimes think that constructive logics (intuitionistic, linear) are weaker, since 'they prove less'. As Kreisel remarked long ago, this is a complete mistake: intuitionistic disjunction is not a classical conjunction with one hand tied behind its back, it is a different operation. By the way, locative phenomena show that these non-classical connectives are no longer bound to be weaker than the 'corresponding' classical ones, which, by the way, are no longer 'corresponding' at all, think of Equation (101)

$$\forall d(G_d \oplus H_d) = (\forall dGd) \oplus (\forall dHd).$$

See: **Affine logic, Classical logic, Linear logic, Intuitionistic logic, Prenex forms**.

• Winning

A behaviour is, as we have said, the ludic equivalent of a formula. In logic formulas are valid (that is, assuming external completeness, provable or not), but it is never the case that both $\mathbf{G}, \mathbf{G}^{\perp}$ are valid. How can we speak of validity, that is, the *truth* of a behaviour? If truth is, by analogy with completeness, the existence of a proof, the truth of a behaviour is something like the existence of a design in the behaviour… But, unfortunately, behaviours are never empty. Hence the idea is to distinguish between the first and second-class citizens among designs. First-class designs will be styled *winning*, which corresponds, in decreasing order of importance, to being uniform, stubborn and parsimonious – what a programme indeed! Now let us go to subtle points:

— Winning must be introspective: just because the output of an interaction cannot be analysed. Hence winning should be a general property of designs. A typical want of taste would be to introduce 'candidates of winning', by saying that a behaviour is given by means of a set of designs together with a distinguished subset (likely to be empty), the *winning* ones; this style of extrospective definition would be a major flaw in the construction, a hidden way to reintroduce the referee.

In fact, *subtyping*, which is inclusion between behaviours $\mathbf{G} \subset \mathbf{H}$, implicitly requires that a design which is winning in $\mathbf{G}$ must remain winning in $\mathbf{H}$… and the logical relation style is incompatible with such a brutal simplicity.

- In a behaviour, a design should be replaceable by its incarnation. If the design is winning, the incarnation should be also. The only way to ensure this property is to require that the winning of $\mathfrak{E}$ implies the winning of $\mathfrak{D}$ as soon as $\mathfrak{D} \subset \mathfrak{E}$.
- The logical relation style '$\mathfrak{F}$ of type $\mathbf{G} \multimap \mathbf{H}$ is winning iff for all winning $\mathfrak{A} \in \mathbf{G}$, the design $(\mathfrak{F})\mathfrak{A}$ is winning' is forbidden. However, it will be true that if $\mathfrak{F}, \mathfrak{A}$ are winning, then $(\mathfrak{F})\mathfrak{A}$ is winning. To sum up, winning is closed under normalisation, that is, a net formed with winning designs must normalise into a winning net… provided it normalises.
- Finally, not every design should be winning. Typically, it should not be possible to find $\mathfrak{D} \in \mathbf{G}$ and $\mathfrak{E} \in \mathbf{G}^{\perp}$ that are both winning. If this were the case, the normal form $\mathfrak{Dai}$ of the net $\{\mathfrak{D}, \mathfrak{E}\}$ would be winning too. Hence the simplest choice is to require daimons to be losing, which is ensured by obstination.

See: **Admissible rule, Barbichette, Completeness (external), Consensus, Dualiser, Incarnation, Intensional, Introspective, Loser, Obstination, Parsimony, Referee, Subtyping, Truth, Uniformity, Xenoglossy**.

• Xenoglossy

I give this rather artificial word, which refers to a *medium* speaking a language that he does not understand, to stress a methodological point, which will be my conclusion, namely the importance of the *cracks in the building*.

The explicit aim of the ludic programme is to give *the* interpretation of logic, and not yet-one-more semantics, natural or not. Where can we find the right ideas? Surely not in our intentions or basic intuitions, even if they are pure: plenty of people have had the same and did not make it. The methodology I use is to start with something rather standard, which has proved to be useful, and to examine the leakage, preferably the small leakage: this is the crack in the building – the building must be safe and the crack almost invisible. You may think that this is obvious, but think again, and look at the tendency of authors to minimise their (small) failures, by tampering with definitions.

Once the small crack has been identified, one should try to enlarge it; it is most likely that some interesting animal lives there. To be concrete, let me take a couple of examples from my personal experience:

- Scott semantics was, in its day, a remarkable achievement. I once remarked that Scott domains are not direct limits of finite ones; this small crack, which some authors filled by styling finite a finitely generated Scott domain, eventually led to the discovery of coherent spaces.
- The interpretation of intuitionistic logic in coherent spaces worked smoothly, apart from the intuitionistic disjunction: given stable maps F from X to Z and G from Y to Z, the union $F \cup G$ does not define a stable map from $X + Y$ to Z, since $F(\varnothing)$

may differ from $G(\emptyset)$. This small crack can be fixed by some complication of the definition, for example, replacing coherent spaces with 'dI domains'. By sticking with coherent spaces I was led to use a linearisation technique, which was the key to linear logic.

— Full completeness theorems work with a limited amount of leakage, provided one excludes the additive neutrals: they are associated with 'zero spaces', which make the interpretation collapse. This is why additive neutrals are absent from all works concerned with full completeness. I realised that this problem had to be solved anyway... The answer led to the recognition of polarity as the major divide of logic, and thus to ludics.

— Of course, some cracks did not reveal anything, for example, the leakage of proof-nets with respect to multiplicative neutrals was too complex to lead to anything.

The problem is to go on, and to know what to do in, say, the delicate problem of exponentials. A few months ago, instead of Chapter 8, there was a chapter about exponentials – and it worked! Yes it worked, but not with the miraculous mutual accommodation of the concepts that you can find in the present manuscript. Hence I decided that there was something I had not understood, and removed the chapter from the final version. Something is missing, but exponentials form too big a problem to be attacked directly within a reasonable time.

Fortunately, ludics, as it is, is not perfect – even if it is more satisfactory than any previous explanation. There is even one small crack, namely the mismatch between parsimony and exactness. Remember that I proved full completeness with respect to exactness, but that I consider parsimony to be the correct notion, since exactness is not interactive. All existing full completeness results are based on non-interactive definitions, for example, some external uniformity; hence, why not change my definitions and call 'winning' a strategy that comes from an exact design? Just because it would refer to some 'moral' principle (exactness), which can lead to Tarskian regressions. Something is winning when it wins all particular plays (disputes), period. Changing the definition of winning from parsimonious to exact is typical of the mentality of a *bounty hunter*, which I consider adverse to science.

We are left with two possible *scenarios*:

Pessimistic version: Parsimony cannot make its way. I am sorry, but then we should accept weakening as a correct logical principle, whether we like it or not.

Optimistic version: Parsimony is the same as exactness, provided the notion of design is liberalised. This liberalisation consists of proving a strengthened form of separation:

If $\mathfrak{D} \in \mathbf{G}^p$ is finite and material, then there exists $\mathfrak{E} \in \mathbf{G}^\perp$ such that the normalisation of $\{\mathfrak{D}, \mathfrak{E}\}$ consumes exactly $\mathfrak{D}$.

Moreover, the main highways, stability, associativity, *etc.* should remain.

I do believe on the optimistic scenario, and I have started to work on it. This global consumption requires both a desequentialisation of designs (several actions in parallel) and a possibility of reusing foci, which will produce superimpositions (up to non-determinism).

Then the requirement of stability should involve the use of real coefficients, leading to a probabilistic approach, or to a new version of GoI…

Chi vivrà vedrà…

See: **Affine logic, Artificiality, Boots, Daimon, Dualiser, Exactness, Exponentials, Full completeness, Geometry of interaction, Leakage, Linear logic, Ludics, Naturality, Non-determinism, Parsimony, Perishable, Polarity, Scott domain, Semantics, Weakening, Winning**.

NON SI NON LA

Appendix B. Bestiary

The essential designs-dessins have been listed below, with, depending on their polarity, positive or negative logos.

FAITH

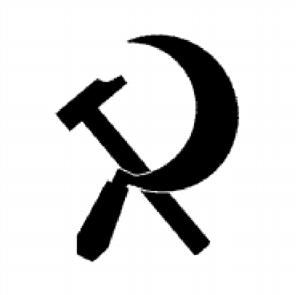

Fig. 1. $\mathfrak{Fid}$

$$\frac{}{\vdash \Lambda}\ \Omega$$

$\mathfrak{Fid}$ is not a design, since it denotes the absence of design, in general because of the divergence of normalisation. $\mathfrak{Fid}$ is a *partial* design, indeed the only quite partial design of a given base.

DAIMON

Fig. 2. $\mathfrak{Dai}$

$$\dfrac{}{\vdash \Lambda}\,\maltese$$

$\mathfrak{Dai}$ belongs to all positive behaviours, in particular $\mathbf{0} = \{\mathfrak{Dai}\}$.

DAIMON (negative)

Fig. 3. $\mathfrak{Dai}^-$

$$\dfrac{\ldots \qquad \dfrac{}{\vdash \xi * I, \Lambda}\,\maltese \qquad \ldots}{\xi \vdash \Lambda}\,(\xi, \wp_f(\mathbb{N}))$$

$\mathfrak{Dai}^-$ belongs to all negative behaviours, in particular $\mathbf{0}^- = \{\mathfrak{Dai}^-\}$.

SKUNK

Fig. 4. $\mathfrak{Sk}$

$$\cfrac{}{\xi \vdash \Lambda}\,(\xi,\emptyset)$$

The principal behaviour of $\mathfrak{Sk}$ is the largest one, $\top$, whose incarnation is so small $|\top| = \{\mathfrak{Sk}\}$. The Skunk is orthogonal to the sole Daimon, hence $\mathfrak{Sk} \in \mathbf{G} \Rightarrow \mathbf{G} = \top \ldots$ and $|\mathbf{G}| = \{\mathfrak{Sk}\}$.

SKUNK (positive)

Fig. 5. $\mathfrak{Sk}_{(\lambda,I)}$

$$\cfrac{\ldots \quad \cfrac{}{\lambda * i \vdash}\,(\lambda{*}i,\emptyset) \quad \ldots}{\vdash \Lambda}\,(\lambda,I)$$

With $\lambda \in \Lambda$. The incarnation of the greatest positive behaviour $\top^+$ is equal to $\{\mathfrak{Sk}_{(\lambda,I)}\,;\lambda \in \Lambda, I \in \wp_f(\mathbb{N})\} \cup \{\mathfrak{Dai}\}$.

RAMIFICATION

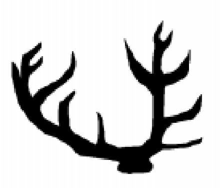

Fig. 6. $\mathfrak{Ram}_{(\lambda,I)}$

$$\dfrac{\cdots \qquad \dfrac{\cdots \dfrac{}{\vdash \lambda * i * J}\,\maltese \quad \cdots}{\lambda * i \vdash}\,{\scriptstyle(\lambda*i,\wp_f(\mathbb{N}))} \qquad \cdots}{\vdash \Lambda}\,{\scriptstyle(\lambda,I)}$$

$\P\mathbf{G} = \{I\,;\mathfrak{Ram}_{(\lambda,I)} \in \mathbf{G}\}$ indexes the connected components of $\mathbf{G}$.

DIRECTORY

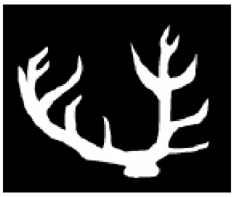

Fig. 7. $\mathfrak{Dir}_{\mathcal{N}}$

$$\dfrac{\cdots \qquad \dfrac{}{\vdash \xi * I, \Lambda}\,\maltese \qquad \cdots}{\xi \vdash \Lambda}\,{\scriptstyle(\xi,\mathcal{N})}$$

$\mathfrak{Dai}^-$, $\mathfrak{Sk}$ and $\mathfrak{Boots}$ correspond to the cases $\mathcal{N} = \wp_f(\mathbb{N}), \emptyset, \{\emptyset\}$. $\P\mathbf{G}$ is defined by $|\mathfrak{Dai}^-| = \mathfrak{Dir}_{\P\mathbf{G}}$.

https://doi.org/10.1017/S096012950100336X

ONE

Fig. 8. $\mathfrak{One}$

$$\cfrac{}{\vdash \xi}\,(\xi,\emptyset)$$

This design is the unit of the tensor product $\olessthan, \odot, \obar$. The behaviour $\mathbf{1} = \{\mathfrak{Dai}, \mathfrak{One}\}$ is the unit of the tensor products $\olessthan, \odot, \obar$.

BOOTS

Fig. 9. $\mathfrak{Boots}$

$$\cfrac{\cfrac{}{\vdash}\,\maltese}{\xi \vdash}\,(\xi,\{\emptyset\})$$

The dual neutral element $\perp = \mathbf{1}^{\perp}$ is defined by its incarnation $|\perp| = \{\mathfrak{Boots}\}$.

FAX

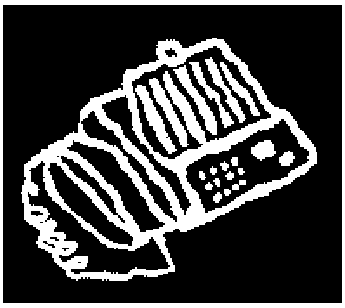

Fig. 10. $\mathfrak{Fax}_{\xi,\xi'}$

$$\cfrac{\dots \qquad \cfrac{\begin{array}{c}\vdots\ \mathfrak{Fax}_{\xi'*i,\xi*i}\\ \dots\ \xi' * i \vdash \xi * i \dots\end{array}}{\vdash \xi', \xi * I}\,(\xi', I)}{\xi \vdash \xi'}\,(\xi, \wp_f(\mathbb{N}))$$

The fax is the most important design, since it implements the 'identity axiom', which is, indeed, a *delocation axiom*, since it relates two disjoint *loci* ξ, ξ'.

References

Abadi, M. and Cardelli, L. (1996) *A theory of objects*, Monographs in Computer Science, Springer.

Abramsky, S. and Jagadeesan, R. (1994) Games and Full Completeness for Multiplicative Linear Logic. *Journal of Symbolic Logic* **59** (2) 543–574.

Abramsky, S., Jagadeesan, R. and Malacaria, P. (2000) Full Abstraction for PCF. *Information and Computation* **163**.

Abramsky, S. and McCusker, G. (1999) Game semantics. In: Berger, U. and Schwichtenberg, H. (eds.) *Computational Logic*, Springer-Verlag, NATO series F **165** 1–56.

Abrusci, V. M. (2000) Syllogisms and linear logic. Preprint, Dipartimento di Informatica, Università Roma III.

Abrusci, V. M. and Ruet, P. (2000) Non-commutative logic I: the multiplicative fragment. *Annals of Pure and Applied Logic* **101** 29–64.

Altenkirch, T. (1993) *Constructions, Inductive Types and Strong Normalization.* Ph. D. thesis, University of Edinburgh.

Amadio, R. and Curien, P.-L. (1998) *Domains and Lambda-Calculi*, Cambridge Tracts in Theoretical Computer Science **46**, Cambridge University Press.

Andreoli, J.-M. and Pareschi, R. (1991) Linear objects: logical processes with built-in inheritance. *New Generation Computing* **9** (3-4) 445–473.

Asperti, A. (1987) A logic for concurrency. Technical report, Dipartimento di Informatica, Pisa.

https://doi.org/10.1017/S096012950100336X

Asperti, A. (1998) Light affine logic. In: *13th Annual IEEE Symposium on Logic in Computer Science*, IEEE Computer Society Press 300–309.

Bainbridge, S., Freyd, P. J., Scedrov, A. and Scott, P. J. (1990) Functorial polymorphism. *Theoretical Computer Science* **70** 35–64.

Barendregt, H. P. (1984) *The Lambda Calculus: its Syntax and Semantics*, revised edition, North-Holland.

Barr, M. (1979) $*$-autonomous categories. *Springer-Verlag Lecture Notes in Mathematics* **752**.

Barr, M. (1991) $*$-autonomous categories and linear logic. *Mathematical Structures in Computer Science* **1** (2) 159–178.

Berlusconi, S. (1992) Logical aspects of aerodynamics. Technical report, Forza Logica preprint series, Milan.

Berry, G. (1978) Stable models of typed lambda-calculi. In: Proceedings of the 5th International Colloquium on Automata, Languages and Programming. *Springer-Verlag Lecture Notes in Computer Science* **62**.

Berry, G. and Curien, P.-L. (1982) Sequential algorithms on concrete data structures. *Theoretical Computer Science* **20** 265–321.

Blass, A. (1972) Degrees of indeterminacy of games. *Fundamenta Mathematicæ* **77** 151–166.

Blass, A. (1992) A game semantics for linear logic. *Annals of Pure and Applied Logic* **56** 183–220.

Blute, R. and Scott, P. (1996) Linear Laüchli semantics. *Annals of Pure and Applied Logic* **77** 101–142.

Blute, R. and Scott, P. (1998) The Shuffle Hopf Algebra and Noncommutative Full Completeness. *Journal of Symbolic Logic* **63** 1413–1436.

Böhm, C. (1968) *Alcune proprietà delle forme β-η-normali nel λ-K-calcolo*, Pubblicazioni dell'Istituto per le Applicazioni del Calcolo.

Bucciarelli, A. and Ehrhard, T. (1993) A theory of sequentiality. *Theoretical Computer Science* **113** 273–291.

Bucciarelli, A. and Ehrhard, T. (2000) On phase semantics and denotational semantics: the exponentials. *Annals of Pure and Applied Logic* (to appear).

Cartwright, R., Curien, P.-L., and Felleisen, M. (1994) Fully abstract semantics for observably sequential languages. *Information and Computation* 297–401.

Cervesato, I. and Pfenning, F. (1996) A linear logical framework. In: *Proc. of 11th IEEE Symposium on Logic in Computer Science, New Brunswick*, IEEE Computer Society Press 264–275.

Church, A. and Rosser, J. B. (1936) Some properties of conversion. *Transactions of the American Mathematical Society* **39** 472–482.

Consel, C. and Danvy, O. (1993) Tutorial Notes on Partial Evaluation. In: *Proceedings of the Twentieth Annual ACM Symposium on Principles of Programming Languages, Charleston, South Carolina*, ACM Press 493–501.

Coppo, M., Dezani-Ciancaglini, M., and Venneri, B. (1981) Functional Characters of solvable terms. *Zeitschrift für Mathematische Logik und Grundlagen Math.* **1** (27) 45–58.

Coquand, T. and Huet, G. (1988) The calculus of constructions. *Information and Computation* **76** 95–120.

Curien, P.-L. (1994) On the symmetry of sequentiality. In: Proceedings of Mathematical Foundations of Programming Semantics 1993. *Springer-Verlag Lecture Notes in Computer Science* **802** 29–71.

Curien, P.-L., Plotkin, G. and Winskel, G. (2000) Bistructure models of linear logic. *Milner's Festschrift*, MIT Press.

Danos, V., Herbelin, H. and Regnier, L. (1996) Games Semantics and Abstract Machines. In: *Proceedings of the 11th Symposium on Logic in Computer Science, New Brunswick*, IEEE Computer Society Press.

Danos, V. and Regnier, L. (1989) The structure of multiplicatives. *Archive for Mathematical Logic* **28** 181–203.

Ehrhard, T. (1995) Hypercoherences: a strongly stable model of linear logic. In: Girard, J.-Y., Lafont, Y. and Regnier L. (eds.) *Advances in Linear Logic*, Cambridge University Press 83–108.

Ehrhard, T. (1999) A relative definability result for strongly stable functions and some corollaries. *Information and Computation* **152** 111–137.

Felscher, W. (1985) Dialogues, strategies and intuitionistic provability. *Annals of Mathematical Logic* **28** 217–254.

Freyd, P.J. and Scedrov, A. (1990) *Categories, Allegories*, North Holland Mathematical Library, Elsevier **39**.

Gentzen, G. (1969a) Investigations into logical deduction. In: Szabo, M.E. (ed.) *The collected works of Gehrard Gentzen*, North-Holland 68–131.

Gentzen, G. (1969b) New version of the consistency proof for elementary number theory. In: Szabo, M.E. (ed.) *The collected works of Gehrard Gentzen*, North-Holland 68–131.

Gentzen, G. (1969c) The consistency of elementary number theory. In: Szabo, M.E. (ed.) *The collected works of Gehrard Gentzen*, North-Holland 68–131.

Gentzen, G. (1969d) The consistency of the simple theory of types. In: Szabo, M.E. (ed.) *The collected works of Gehrard Gentzen*, North-Holland 68–131.

Girard, J.-Y. (1971) 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 (ed.) *Proceedings of the* 2^{nd} *Scandinavian Logic Symposium*, North-Holland 63–92.

Girard, J.-Y. (1972) *Interprétation fonctionnelle et élimination des coupures de l'arithmétique d'ordre supérieur*, Thèse de Doctorat d'Etat, Université Paris VII, Paris.

Girard, J.-Y. (1984) The Ω-rule. In: *Proceedings of the International Congress of Mathematicians*, Warsaw, PWN, Polish Scientific Publishers 307–321.

Girard, J.-Y. (1986) The system $\mathbb{F}$ of variable types, fifteen years later. *Theoretical Computer Science* **45** 159–192.

Girard, J.-Y. (1987a) Linear logic. *Theoretical Computer Science* **50** 1–102.

Girard, J.-Y. (1987b) *Proof-theory and logical complexity I*, Bibliopolis, Naples.

Girard, J.-Y. (1988) Multiplicatives. In: Lolli (ed.) *Logic and computer science: new trends and applications*, Rendiconti del seminario matematico dell'università e politecnico di Torino (special issue 1987) 11–34.

Girard, J.-Y. (1989a) Geometry of interaction I: interpretation of system *F*. In: Ferro, Bonotto, Valentini and Zanardo (eds.) *Logic Colloquium '88*, North-Holland 221–260.

Girard, J.-Y. (1989b) Towards a geometry of interaction. In: *Categories in Computer Science and Logic*, Proceedings of Symposia in Pure Mathematics **92**, American Mathematical Society, 69–108.

Girard, J.-Y. (1991) A new constructive logic: classical logic. *Mathematical Structures in Computer Science* **1** 255–296.

Girard, J.-Y. (1993) On the unity of logic. *Annals of Pure and Applied Logic* **59** 201–217.

Girard, J.-Y. (1995a) Geometry of interaction III: accommodating the additives. In: Girard, J.-Y., Lafont, Y. and Regnier L. (eds.) *Advances in Linear Logic*, Cambridge University Press 329–389.

Girard, J.-Y. (1995b) Linear logic, its syntax and semantics. Girard, J.-Y., Lafont, Y. and Regnier L. (eds.) *Advances in Linear Logic*, Cambridge University Press 1–42.

Girard, J.-Y. (1996) Proof-nets: the parallel syntax for proof-theory. In: Ursini and Agliano (eds.) *Logic and Algebra*, Marcel Dekker, New York.

Girard, J.-Y. (1999a) On denotational completeness. *Theoretical Computer Science* **227** 249–273.

Girard, J.-Y. (1999b) On the meaning of logical rules I: syntax vs. semantics. In: Berger, U. and Schwichtenberg, H. (eds.) *Computational Logic*, NATO series F **165**, Springer-Verlag 215–272.

https://doi.org/10.1017/S096012950100336X

Girard, J.-Y. (2000) On the meaning of logical rules II: multiplicative/additive case. In: Bauer, F. L. and Steinbrggen, R. (eds.) *Foundation of Secure Computation*, NATO series F **175**, IOS Press, Amsterdam 183–212.

Girard, J.-Y., Lafont, Y. and Taylor, P. (1990) *Proofs and types*, Cambridge tracts in theoretical computer science **7**, Cambridge University Press.

Gödel, K. (1931) Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte füt Mathemayik und Physik* **38** 173–198. (English translation (1967): On formally undecidable propositions of *Principia Mathematica* and related systems I. In: van Heijenoort (ed.) *From Frege to Gödel*, Harvard University Press.)

Gödel, K. (1958) Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes. *Dialectica* **12** 280–287.

Grishin, V. N. (1982) Predicate and set-theoretic calculi based on logics without contractions. *Math. USSR Izvestiya* **18** 41–59.

Guerrini, S. (1999) Correctness of multiplicative proof-nets is linear. In: *14th Annual IEEE Symposium on Logic in Computer Science* (*LICS '99*), IEEE Computer Society Press 454–463.

Hamano, M. (2000) Pontrjagin duality and full completeness for multiplicative linear logic (without mix) *Mathematical Structures in Computer Science* **10** 213–259.

Harrington, L. A., Morley, M. D. and Scedrov, A. (eds.) (1985) *H. Friedman's Research on the Foundations of Mathematics*, North-Holland.

Hatcliff, J. and Danvy, O. (1994) A Generic Account of Continuation-Passing Styles. In: *Proceedings of the Twenty-First Annual ACM Symposium on Principles of Programming Languages, Portland, Oregon*, ACM Press 458–471.

Hergé (1950) *Tintin au pays de l'Or Noir*, Casterman, Tournai.

Hilbert, D. (1926) Über das Unendliche. *Mathematische Annalen* **95** 161–190. (English translation (1967): On the infinite. In: van Heijenoort (ed.) *From Frege to Gödel*, Harvard University Press.)

Honsell, F. and Ronchi, S. (1992) An approximation theorem for topological lambda models and the topological incompleteness of lambda calculus. *Journal of Computer and System Sciences* **45** (1) 49–75.

Howard, W. A. (1980) The formulae-as-types notion of construction. In: Hindley, J. P. and Seldin, J. R. (eds.) *To H. B. Curry: Essays on Combinatory logic, Lambda-calculus and Formalism*, Academic Press 479–490.

Hyland, J. M. E. , Robinson, E. P., and Rosolini, G. (1989) Algebraic types in PER models. In: Fifth conference on mathematical foundations in programming semantics. *Springer-Verlag Lecture Notes in Computer Science* **442** 333–350.

Hyland, M. and Ong, L. (2000) On Full Abstraction for PCF. *Information and Computation* **163**.

Kanovitch, M. (1991) The multiplicative fragment of linear logic is NP-complete. Technical report, Institute for language, logic and information, Amsterdam.

Kleene, S. C. (1952) *Introduction to metamathematics*, North-Holland.

Kleene, S. C. (1978) Recursive Functionals and Quantifiers of Finite Types Revisited I. In: Fenstad *et al.* (eds.) *Proc. General Recursion Theory II*, North-Holland.

Kreisel, G. (1958) Mathematical significance of consistency proofs. *Journal of Symbolic Logic* **23** 75–99.

Kreisel, G. (1965) Mathematical logic. In: Saaty, T. L. (ed.) *Lectures in modern mathematics, vol III*, Wiley and Sons 99–105.

Kreisel, G. and Levy, A. (1968) Reflection principles and their uses for establishing the complexity of axiomatic systems. *Zeitschrift für Mathematische Logik* **14** 97–142.

Kreisel, G., Mints, G. and Simpson, G. (1975) The use of abstract language in elementary mathematics: some pedagogical examples. In: Parikh, R. (ed.) Logic Colloquium, Boston. *Springer-Verlag Lecture Notes in Mathematics* **453** 38–131.

https://doi.org/10.1017/S096012950100336X

Krivine, J.-L. (1990) *Lambda-calcul, types et modèles*, Etudes et recherches en informatique, Masson.

Lafont, Y. (1996) The undecidability of second-order linear logic without exponentials. *Journal of Symbolic Logic* **61** 541–548.

Lamarche, F. (1992) Sequentiality, games and linear logic (preprint).

Lambek, J. (1958) The mathematics of sentence structures. *American Mathematical Monthly* **65** 154–169.

Läuchli, H. (1970) An Abstract Notion of Realizability for which Intuitionistic Predicate Calculus is Complete. In: *Intuitionism and Proof Theory*, North-Holland 227–234.

Lincoln, P., Mitchell, J. C., Shankar, J. C. and Scedrov, A. (1990) Decision problems for propositional linear logic. In: *Proceedings of 31st IEEE symposium on foundations of computer science, volume 2*, IEEE Computer Society Press 662–671.

Longley, J. R. (1998) The sequentially realizable functionals. Technical Report ECS-LFCS-98-402, LFCS. (To appear in *Annals of Pure and Applied Logic*.)

Longo, G. (2001) The reasonable effectiveness of Mathematics and its cognitive roots. In: Boi (ed.) *New Interactions of Mathematics with Natural Sciences*, Springer-Verlag.

Lorenz, K. (1968) Dialogspiele als semantische Grundlage von Logikkalkülen. *Archiv für Mathematik und Logik Grundlagenforschung* **11** 32–55, 73–100.

Lorenzen, P. (1960) Logik und Agon. In: *Atti Congresso Internazionale di Filosofia, vol. 4*, Sansoni, Florence 187–194.

Martin-Löf, P. (1984) *Intuitionistic type theory*, Bibliopolis, Naples.

Métayer, F. (1994) Homology of proof-nets. *Archive for Mathematical Logic* **33** 169–188.

Miller, D. (1996) Forum: A Multiple-Conclusion Specification Logic. *Theoretical Computer Science* **165** (1) 201–232.

Milner, R. (1975) Processes, a mathematical model of computing agents. In: *Logic Colloquium, Bristol 1973*, North-Holland 157–174.

Milner, R. (1977) Fully abstract models of typed lambda-calculi. *Theoretical Computer Science* **4** 1–23.

Nickau, H. (1994) Hereditarily Sequential Functionals. In: Nerode, A. and Matiyasevich, Y. V. (eds.) Proc. Symp. Logical Foundations of Computer Science: Logic at St. Petersburg. *Springer-Verlag Lecture Notes in Computer Science* **813** 253-264.

O'Hearn, P. (1999) Resource interpretations, bunched implications and the $\alpha\lambda$-calculus. In: Typed Lambda Calculi and Applications. *Springer-Verlag Lecture Notes in Computer Science* **1581**.

Paiva, V. D. (1989) The Dialectica categories. In: Gray and Scedrov (eds.) *Categories in Computer Science and Logic*, American Mathematical Society 47–62.

Parigot, M. (1992) $\lambda\mu$-calculus: an Algorithmic Interpretation of Classical Natural Deduction. In: Proceedings of International Conference on Logic Programming and Automated Deduction. *Springer-Verlag Lecture Notes in Computer Science* **624** 190–201.

Plotkin, G. D. (1975) Call-by-name, Call-by-value and the λ-calculus. *Theoretical Computer Science* **1** 125–159.

Plotkin, G. D. (1977) LCF considered as a programming language. *Theoretical Computer Science* **5** 223–256.

Prawitz, D. (1965) *Natural deduction, a proof-theoretical study*, Almqvist and Wiksell, Stockholm.

Reynolds, J. C. (1978) Syntactic control of interference. In: *Record of the Fifth Annual Symposium on Principles of Programming Languages, Tucson, Arizona*, ACM Press 297–401.

Reynolds, J. C. (1998) Definitional Interpreters for Higher-Order Programming Languages. *Higher-Order and Symbolic Computation* **11** (4) 363–397. (Reprinted from the proceedings of the 25th ACM National Conference (1972).)

Schroeder-Heister, P. and Došen, K. (1993) *Substructural logics*, Oxford Science Publications.

Schütte, K. (1960a) *Beweistheorie*, Springer-Verlag.

Schütte, K. (1960b) Syntactical and semantical properties of simple type theory. *Journal of Symbolic Logic* **25** 305–326.

Scott, D. (1976) Data types as lattices. *SIAM Journal of computing* **5** 522–587.

Sokal, A. and Bricmont, A. (1999) *Impostures intellectuelles*, Odile Jacob.

Stoy, J. E. (1977) *Denotational Semantics: The Scott–Strachey Approach to Programming Language Theory*, MIT Press.

Tait, W. W. (1966) A non-constructive proof of Gentzen's Hauptsatz for second-order logic. *Bulletin of the American Mathematical Society* **72** 980–983.

Tait, W. W. (1967) Intensional interpretation of functionals of finite type I. *Journal of Symbolic Logic* **32** 198–212.

Takeuti, G. (1953) On a generalized logical calculus. *Japanese Journal of Mathematics* **23** 39–96.

Troelstra, A. S. (1973) Notes on second order intuitionistic arithmetic. In: Mathias and Rogers (eds.) Cambridge Summer School in Mathematical Logic. *Springer-Verlag Lecture Notes in Computer Science* **337** 171–205.

Van Oosten, J. (1997) A combinatory algebra for sequential functionals of finite type. Technical Report 996, University of Utrecht.

Vuillemin, J. (1974) *Syntaxe, Sémantique et Axiomatique d'un Langage de Programmation Simple*, Thèse de doctorat d'état, Université Paris 7.

Waismann, F. (1979) *Wittgenstein and the Vienna circle: conversations recorded by F. Waismann 1929–31*, (English translation), Barnes and Noble, New York.

Weyl, H. (1918) *Das Kontinuum*, Veit, Leipzig.

Wittgenstein, L. (1968) *Philosophical Remarks*, (translated into English by G. E. M. Anscombe), Barnes and Noble, New York.

Yetter, D. N. (1990) Quantales and non-commutative linear logic. *Journal of Symbolic Logic* **55** 41–64.

Zucker, J. (1974) Cut-elimination and normalisation. *Annals of Mathematical Logic* **7** 1–155.

Index