A theory of experiment

Pierre Albarède*

July 22, 2013

Abstract

This article aims at clarifying the language and practice of scientific experiment, mainly by hooking observability on calculability.

1 Motivation

Scientific knowledge is traditionally based on logic and experiment. However, I will defend that experiment involves a lot of logic, if not only logic.

As an experimentalist, I have found out, which is not easy to admit, that experiments do fail logically as well as technically. "Experiment, observable, error, interpretation, constant…" are ill-defined. Every experimental constant hides a potential variable. Technology and jargon are often hopeless attempts to escape a logical quagmire. The very difficulty of discussing experiment indicates a need of logic, where it has been too quickly eliminated, to explain and prevent failures, and eventually to sharpen our understanding of nature, including ourselves.

"Analog computing, numerical experiment, thought experiment, experimental program, object programming, computing hardware, natural or real number" implicitly relate computing and experiment. Could the art of experiment develop somehow like the art of computer programming, at least for the sake of their cooperation? I will defend that experiment is computing, by hooking observability on calculability, a well-established mathematical concept [1].

*rés. Valvert, 12 rue de la Fourane, 13090 Aix-en-P., France, pierre.albarede@free.fr

A theory of experiment should describe: - objects, with just as much detail as needed, symbolically, - relations, between a user and objects, - interactions, between objects, - the user's action, to extract information from objects. A duplicity of relation and action is a feature of both experiment and functional programming. Both symbolism and functional programming are supported by *Mathematica* [2, 3].

2 Formal system

2.1 Calculability

The user expresses y in a natural lexicon by x in a formal lexicon, $y \triangleright x, x \triangleleft y$. The formal expresses the natural: time$\triangleright t$, let t be time, for time t. An object is nothing but what the user expresses, an object of expression. An *atom* is a symbol, like π, z, δ, or a numeral, like $3, 22/7, 3.14$; although π expresses a number (the half-circle perimeter), it is not a numeral. The formal lexicon is not outside the natural lexicon; on the contrary, as we speak naturally of π, the natural tends to encompass the formal.

An expression immediately does or does not match a pattern (not a set), expressed by _. For example, _ matches any expression, x_ matches any expression, to be named x, _Integer matches any integer numeral.

A formal system rewrites or evaluates any expression given by the user, according to rules, possibly using patterns. Although an expression may mean something for the user, meaning must not affect evaluation. A rule is potential or delayed evaluation. For y in the formal lexicon, as there is no better expression of y than y itself, $y \triangleright x$ is expressed by a replacement rule, $x \to y$, so that the formal system will evaluate $x \mapsto y$. $\triangleright, \triangleleft, \mapsto$ are not in the formal lexicon, we use them to speak about the user and the formal system; they are meta-symbols (like meta-characters in a typesetting system).

A program is an assembly of rules (obeying meta-rules). The user can rule and interrupt evaluation. Without him, the formal system does nothing, or runs in circles: HOMO EX MACHINA prevents IN GIRUM IMUS NOCTE. This also sounds like the principle of inertia in mechanics.

I take for a definition of recursivity and calculability [3, 3.1.2]. Calculability means evaluating into _Integer or any other numerical pattern, like _Real. For example, π is calculable, but Ω defined in [4] is not.

An expression is – like a tree, with a trunk, branches and leaves, or a body, with a head, members and fingertips – a headed sequence of atoms or, recursively, expressions, like $x[y, z[1, 2]]$ or $x + y$, standing for $\text{Plus}[x, y]$. The depth of an expression is its recursion number, plus one, $\text{Depth}[x[y, z[1, 2]]] \mapsto 3$.

Definition 2.1. *A formal system is universal if it can be ruled to evaluate for any other formal system.*

A Turing machine is a universal formal system. Many microprocessors are Turing machines. The complexity of an expression, whatever the controversial definition of complexity [1], depends on the formal system.

Definition 2.2. *For a formal system f, let $c[x, f] \in \mathbb{R} \cup \{\infty\}$ be the complexity of x in f. $c[x, f] = \infty$ if x is non-calculable. For two formal systems f, g, f is specialized in x, more than g, if $c[x, f] \leq c[x, g]$.*

Specialization is reusable complexity, packaged as a subroutine or a black box, with public (non-private) rules, that are all what the user needs to know.

- A microprocessor is specialized in floating-point arithmetic by a so-called floating-point unit.
- Computational fluid dynamics aims at specializing a formal system in solving fluid-dynamics equations. The public rules are boundary conditions and thermodynamic laws, like an equation of state.
- A set of wires, resistors, generators and gauges can contribute to a formal system, specialized in solving linear algebraic equations with real coefficients. The public rules are (Ohm and Kirchhoff) laws of electrokinetics.
- A primitive idea of geometry is that a material system may be ruled to partly evaluate itself, as the universe is ruled by the gnomon to evaluate star positions [5].

Analog computing is when private rules are discovered, digital computing is when private rules are invented. However, the difference between discovery and invention (or analysis and synthesis, after Kant) is thin, because both are interwoven. Anyway, the user is mostly concerned by computing power, not the status of private rules.

Noise seems to be a particular drawback of analog computing. However, an analog system evaluating x with noise can be considered as evaluating the probability density function of x without noise. In digital computing, apart from purposely introduced Monte-Carlo noise and large thermodynamic fluctuations, floating point arithmetic makes only round-up errors. But round-up errors depend on random hardware, precision, units and programming: for example, numerically (N), $N[\log[3/2]] \neq N[\log[3]] - N[\log[2]]$, so that log can be considered as noisy. Finally, many techniques of numerical evaluation are available, and no one is perfect.

2.2 Logic and algebra

The formal system should support not only replacement rules, but also relational rules _! (not Factorial), and evaluate accordingly, as in

$$x \in A!\ x \in A \mapsto \text{True}\,.$$

Gődel's completude and incompletude theorems set limits on logical evaluation, independently of technique: a formal system supporting the rules of logic but not arithmetic ($\mathbb{N}$), or finite logic, can evaluate all its relational expressions, while a formal system supporting the rules of both logic and arithmetic, or infinite logic, cannot. Nevertheless, the user can always rule what a formal system cannot evaluate, at his own risk of inconsistency or redundancy.

Mathematica partly supports some semantic patterns, like $x\_/; x \in \mathbb{R}$, trying to match any expression of a real number, whatever its syntax. *Semantica* [6], a *Mathematica* add-on, based on Solve, supports more semantic patterns, like $(2x\_)_s$, matching any expression with the double of its half. Semantic pattern matching makes *Mathematica* look like it evaluates according to its own understanding of sets. However, internally, a semantic pattern reduces to syntactic patterns, covering limited instances of the intended semantic pattern. The incompletude theorem implies that no formal system can completely support semantic pattern matching [6]. The ultimate responsibility for semantic pattern matching belongs to the user.

Although an object is identical only to itself [7, 5.5303], the user may want to rule identities between expressions of one object, as in

$$\pi \quad \triangleright \quad 3,\ 3.14,\ \frac{22}{7},\ (\sum_{n=1}^{\infty} \frac{6}{n^2})^{1/2} = 2\,\text{ArcSin}[1] = \oint \frac{d\theta}{2} = \pi.$$

$x = y$ means $x \triangleright y, y \triangleright x$. The formal system should be able to evaluate some identities, if not all. $x = y$ legitimates both $x \to y, y \to x$, but, in general, which one is more useful cannot be decided once for ever. A solution, suggested by Burindan's donkey, the wise animal, would be to keep x, y under the constraint $x = y$.

Definition 2.3. $f[\#]$ *conveys identity, if* $f[x] = f[y]$ *whenever* $x = y$.

$$\begin{array}{ccc} x & = & y \\ \downarrow & f & \downarrow \\ f[x] & = & f[y] \end{array}$$

After $y = f[x]$!

$$\begin{array}{ccc} y & = & f[x] \\ \downarrow & \text{Depth} & \downarrow \\ 1 & \neq & 2 \end{array}$$

Depth[#] does not convey identity.

Definition 2.4. *A* function f *from a set* A *to a set* B *relates every element* $x \in A$ *to one element* $y \in B$*, or no one. Let* $\mathbf{F}[A, B]$ *be the set of functions from* A *to* B*.*

Actually, (def. 2.4) is so informal that a formal system can hardly support it. What *Mathematica* readily supports is pure function, Function,

$$f[\#]\& = \text{Function}[x,\ f[x]],\ f[\#]\&[x] \mapsto f[x],\ \text{a function } f\ \triangleright\ f[\#]\&\ \triangleright\ f.$$

Definition 2.5. *For* $f \in \mathbf{F}[A, B]\ \triangleright\ f[\#]\&$*, and* $x \in A$*, related to* $y \in B$ *by* f*,* $y = f[x]$!

Theorem 2.1. $f[\#]\&$ *expresses a function iff (if and only if)* $f[\#]$ *conveys identity.*

Proof. Only if ($\Rightarrow$): a set, as opposed to a pattern, does not depend on syntax (formalities); $x, x^*, x = x^*$, expressing the same element of A, are related to $y = f[x] = f[\widetilde{x}]$. If ($\Leftarrow$): construct the quotient of f modulo identity. $\square$

As $\#[[1]]\& = \mathrm{Part}[\#, 1]\&$ [2] expresses a projection, $\#[[1]]$ conveys identity. Conversely, as Depth[#] does not convey identity, Depth (syntax-sensitive) expresses no function.

Definition 2.6. *f depends on x or x is implicit in f, $\partial_x f \neq 0$, if f matches $g\_[x]_s$.*

No formal system can completely support $g\_[x]_s$. *Mathematica* does not support semantic functional patterns, neither *Semantica*, because Solve does not support functional equations. For f depending on x, $f = g[x]$, $f[x]$ is not a value of the function f, but a value of a function h, $h[x] = g[x][x]$.

Definition 2.7. *x is completely explicit in $f[x]$ if f does not depend on x.*

Formalization (to express the elements without the sets), carried out to algebra, yields formal algebra. In particular, some basic (relational) rules of pure function algebra are

$$
\begin{aligned}
(f\_ + g\_)[x\_] &= f\_[x\_] + g\_[x\_]! && (2.1)\\
(f\_g\_)[x\_] &= f\_[x\_]g\_[x\_]! && (2.2)\\
x\_ \circ y\_ = \mathrm{Circle}[x\_, y\_]!\ f\_ \circ g\_[x\_] &= f\_[g\_[x\_]]! && (2.3)
\end{aligned}
$$

As for a vector space, in order to reuse existing rules, while discerning field elements, I propose to express, on one hand, both internal and external multiplicative laws by Times, orderless,

$$x\_y\_ = \mathrm{Times}[x\_, y\_]!\ x\_y\_ = y\_x\_!$$

on the other hand, a field element by $x \in K!$

$$-1 \in K! \qquad x\_ - y\_ = x\_ + (-1)y\_!$$

Formal function algebra is specialized in formal function K-algebra by

$$f\_[\_]/; f \in K \to f.$$

Definition 2.8. *A function f is a K-linear , $f \in \mathrm{LFs}[K]$, if*

$$f[x\_ + y\_] = f[x\_] + f[y\_], \qquad f[\lambda\_x\_] = \lambda\_f[x\_]/; \lambda \in K.$$

For $f \in \mathrm{LFs}[K]$, $f[x\_ - y\_] = f[x\_] - f[y\_]$.

2.3 Perturbation

Definition 2.9. *The perturbation and error symbols are* Star, δ*. For* x, x^* *expressing elements of an additive group,* $x \triangleright x^*$*, with the error*

$$\delta[x] = x^* - x.$$

For x, x^* *expressing elements of a real vector space, and a perturbation amplitude* $\epsilon \in [0,1]$*,* $x \triangleright x^*_\epsilon$*, with the error*

$$\delta_\epsilon[x] = x^*_\epsilon - x.$$

$$\begin{aligned} x^* &= \mathrm{Star}[x], & x^*_\epsilon &= \mathrm{Star}[\epsilon, x], \\ & & \delta_\epsilon[x] &= \delta[\epsilon, x], \\ x^*_0 &\to x, & x^*_1 &\to x^*, \\ f\_[x]^* &= f\_{}^*[x^*], & f\_[x]^*_\epsilon &= f\_{}^*_{\epsilon}[x^*_\epsilon]. \end{aligned}$$

- x *is* constant *(non-variable),* $x \in \mathrm{Cst}$*, if* $\delta[x] = \delta_{\_}[x] = 0$.
- x *is* unshielded *(non-shielded),* $x \in \mathrm{Uns}$*, if* $\delta_{\epsilon\_}[x] = \epsilon\_\delta[x]$.

$\delta_\epsilon[x] = \epsilon\delta[x]$ for $\epsilon \in \{0,1\}$ and even for $\epsilon \in [0,1]$, if $x \in \mathrm{Uns}$.
From (def. 2.9), we infer

$$\begin{aligned} \delta_0[\_] &\to 0, & \delta_1 &\to \delta, \\ x\_{}^*/; x \in \mathrm{Cst} &\to x, & x\_{}^*_{\epsilon}/; x \in \mathrm{Cst} &\to x. \end{aligned}$$

As $\mathrm{Cst}, \mathrm{Uns}$ express no sets outside the formal system, I call them pseudo-sets. If $x \in \mathrm{Cst}$, then $x \in \mathrm{Uns}$. The formal system should at least partly support the pseudo-inclusion $\mathrm{Cst} \subset \mathrm{Uns}$, as in

$$x \in \mathrm{Cst}!\ x \in \mathrm{Uns} \mapsto \mathrm{True}\,.$$

The interest of $x \triangleright x^*$ without $x^* \to x$ is when x^* matches a simple pattern, that x does not match. Typically, $x \in \mathbb{R}$ and x^* is a floating-point numeral, matching _Real.

$\delta[\#]$ is not meant to convey identity. For example, let x, y be the lengths of both branches of the upright-faced letter V; although ideally $x = y$, there

Table 1: compound error (chain rule)

$$\begin{array}{ccccc} & & (f+\delta_\epsilon[f])[y+\delta_\epsilon[y]] & & \\ & & \parallel & & \\ \delta_\epsilon[f[x]] & = & f[x]^*_\epsilon & - & f[x] \\ \triangledown & & \triangledown & & \triangledown \\ \Delta_\epsilon[f] & = & f^\star_\epsilon & - & f \end{array}$$

are drawing errors $\delta[x], \delta[y]$, that need not be identical to zero nor to each other.

$$\begin{array}{ccc} x & = & y \\ \downarrow & \delta & \downarrow \\ \delta[x] & \neq & \delta[y] \end{array}$$

A physicist would say (privately, so that only another physicist can understand): perturbation breaks the Curie principle. From (the. 2.1), δ expresses no function. If you reject the pure function δ, then you must consider δx as an atom, and you cannot speak of error, abstractly or symbolically.

Perturbation and error expressions are naturally and informally abbreviated using total perturbation and total error (like "total derivation") $\Delta, \star$ (tab. 1).

Theorem 2.2.

$$\Delta_\epsilon[f] \triangleleft \delta_\epsilon[f[x]] = f[x^*_\epsilon] - f[x] + \delta_\epsilon[f][x^*_\epsilon].$$

Proof. Use (2.3). $\square$

Like motion, perturbation is relative to a steady frame, made of constant symbols:

$$\text{Plus} \in \text{Cst}! \;\; \text{Times} \in \text{Cst}! \;\; -1 \in \text{Cst}! \;\; \text{Power} \in \text{Cst}! \;\; \text{Circle} \in \text{Cst}! \qquad (2.4)$$

Theorem 2.3. *For $f \in \text{Cst}$, Star, f commute,*

$$f[x]^*_\epsilon = f[x^*_\epsilon], \qquad \delta_\epsilon[f[x]] = f[x^*_\epsilon] - f[x].$$

If moreover $f \in \mathrm{LFs}[K]$*, then* δ, f *commute,*

$$\delta_\epsilon[f[x]] = f[\delta_\epsilon[x]].$$

Theorem 2.4 (product error).

$$\delta_\epsilon[yx] = y\delta_\epsilon[x] + x^*_\epsilon \delta_\epsilon[y].$$

If moreover $x, y \in \mathrm{Uns}$*, then*

$$\delta_\epsilon[yx] = \epsilon(y\delta[x] + x\delta[y]) + \epsilon^2 \delta[x]\delta[y].$$

Proof. Use (the. 2.3), with $f \to \mathrm{Times}$; $\mathrm{Times} \in \mathrm{Cst}$ (2.4) and Times is orderless. $\square$

Theorem 2.5. *For* $x, y \in \mathrm{Cst}$*,* $x + y, xy, x^{-1} \in \mathrm{Cst}$*,* Cst *is an algebra. Moreover,* $x^y, x[y], x \circ y \in \mathrm{Cst}$*.* $\mathrm{Star} \in \mathrm{LFs}[\mathrm{Cst}]$*,* $\delta \in \mathrm{LFs}[\mathrm{Cst}]$*.* Cst *is the kernel of* δ*.* Uns *is a* Cst*-vector space. If* $f \in \mathrm{Uns}$ *and* $x \in \mathrm{Cst}$*, then* $f[x] \in \mathrm{Uns}$*.* Uns *is stable for* $f \in \mathrm{Cst}$*,* $f \in \mathrm{LFs}[K]$*.*

Proof. Firstly, use (2.4). Lastly, use (the. 2.3),

$$\forall x \in \mathrm{Uns},\ \delta_\epsilon[f[x]] = f[\delta_\epsilon[x]] = f[\epsilon\delta[x]] = \epsilon f[\delta[x]] = \epsilon\delta[f[x]],\ f[x] \in \mathrm{Uns}\,.$$

$\square$

Attention: from (the. 2.4), the product of unshielded expressions is in general shielded, Uns is not an algebra.

3 Experiment

3.1 Observability

Definition 3.1. *The world is a real vector space* $\mathcal{T}$*. A* state *is* $T \in \mathcal{T}$*. For* $p \in \mathbb{N}$*,*

$$\mathcal{T}_p \to \mathrm{If}[p = 0, \mathcal{T}, \mathbf{F}[\mathbb{R}^p, \mathcal{T}]].$$

A material system *is* $T \in \mathcal{T}_p$*. If* $p \neq 0$*, then an* input *of* T *is* $z \in \mathbb{R}^p$*,* $\exists T[z] \in \mathcal{T}$*. The input number of* T *is* p*.*

Definition 3.2. *For a material system T, an expression x is abstract (non-concrete), $x \in \mathrm{Abs}$, if it does not depend (def. 2.6) on T.*

The complexity of concrete expressions decreases with concrete specialization (def. 2.2).

Definition 3.3. *For $n, p \in \mathbb{N}$, $M \in \mathbf{F}[\mathcal{T}_p, \mathbb{R}^n]$, $T \in \mathcal{T}_p$, the* material property *$M[T]$ is* observable, *as the output of the* experiment *$\{M, T\}$, and n is the output number of the* experiment, *if $M[T]$ is calculable. For $p = 0$, M is a state function and the experiment is static.*

The experiment aims at partly evaluating T, which is not required to be observable a priori. Let M be a state function: for $T \in \mathcal{T}$, $\{M, T\}$ is a static experiment; for $p \neq 0$, $T \in \mathcal{T}_p$, $\{M \circ \#\&, T\}$ is an experiment, and its output is the function $M \circ T$.

Definition 3.4. *For $n, p \in \mathbb{N}$, $M \in \mathbf{F}[\mathcal{T}_p, \mathbb{R}^n]$, $T \in \mathcal{T}_p$, Γ is a* gauge *of $M[T]$, sensitive to the* flux *$\Phi[T]$, if $M[T] = \Gamma[\Phi[T]]$ and Γ is calculable.*

Γ aims at making $\Gamma[\Phi[T]]$ less complex that $\Phi[T]$, which is not required to be observable a priori. Γ, Φ express laws of physics (or any experimental science).

Theorem 3.1. *For $M[T] \in \mathbb{R}^n$, $M[T]$ observable, and $P \in \mathbf{F}[\mathbb{R}^n, \mathbb{R}]$, P calculable, $P[M[T]]$ is observable, as the the output of the experiment $\{P \circ M, T\}$.*

Proof. The compound of calculable functions is calculable. $\square$

For example, $n \to 1$, $f \to \log$, if $M[T] > 0$, then $\log[M[T]]$ is observable.

Definition 3.5. *For an experiment $\{T, M\}$, and a calculable function P of $M[T]$, P interprets $M[T]$ to $\widetilde{M}[T] = P[M[T]]$. The interpretation P is* tautologic *if $P = \mathrm{Identity}$,* private *if $\widetilde{M} \notin \mathrm{Abs}$.*

$\#[[1]]\&$ interprets $\{M_1[T], M_2[T]\}$ to $M_1[T]$, yielding a new experiment $\{M_1, T\}$.

Theorem 3.2. *For two experiments $\{M, T\}$ and $\{\widetilde{M}, T\}$, and a calculable function P of $M[T]$, P interprets $M[T]$ to $\widetilde{M}[T]$ iff P is a gauge of $\widetilde{M}[T]$, sensitive to $M[T]$. Moreover, for $M \in \mathrm{Abs}$, the interpretation P from $M[T]$ is private (def. 3.5) iff the gauge P of $\widetilde{M}[T]$ is concrete (def. 3.2).*

Proof. Use (def. 3.4, def. 3.5, the. 3.1) and $\widetilde{M}[T] = P[M[T]]$. $\square$

A private interpretation should be presented as a concrete gauge.

3.2 Controllability

Definition 3.6. *Let* $S \in \mathbf{F}[\mathbf{F}[\mathbb{R},\mathbb{R}],\mathbb{R}]$, *S calculable. The set of material systems* T, controllable *by the servo-function* S, *according to the state function* $R \in \mathbf{F}[\mathcal{T},\mathbb{R}]$, *is*

$$\mathcal{C}[R,S] = \{T \in \mathcal{T}_1,\ \exists!\rho[T] \in \mathbb{R},\ R[T[\rho[T]]] = 0,\\ R \circ T \ \textit{observable},\ \rho[T] = S[R \circ T]\}. \quad (3.1)$$

For $T \in \mathcal{C}[R,S]$, *the* eigeninput *of* T *is* $\rho[T]$.

For $T \in \mathcal{T}_1$, S interprets $R \circ T$ to $\rho[T]$. For example, let $S[f]$ be the root of $f \in \mathbf{F}[\mathbb{R},\mathbb{R}]$, f existing everywhere, continuous, strictly monotonous and crossing 0. $S[f]$ is calculable by dichotomy. For $T \in \mathcal{T}_1$, $T \in \mathcal{C}[R,S]$ if $R \circ T$ is like f before.

Definition 3.7. *The constraint pure function is* C, $\underline{x\_} = C[x\_]$,

$$\underline{(M\_[T\_][\#]\&)_s}/;(T \in \mathcal{C}[R,S], M[T] \notin \mathcal{C}[R,S]) \to M[T][\rho[T]].$$

Theorem 3.3. *For* $P \in \mathrm{Abs}$ *(def. 3.2),* C, P *commute.*

Proof.

$$\underline{P[M[T][\#]\&]} = P[M[T][\rho[T]]] = P[\underline{M[T][\#]\&}].$$

For $M[T]$ matching semantically a function, like the derivative $M[T] = T'$, shortly,

$$\underline{P[M[T]]} = P[M[T][\rho[T]]] = P[\underline{M[T]}].$$

□

3.3 Perturbation under constraint

Theorem 3.4. *If* $\rho \in \mathrm{Cst}$ *and* $\mathrm{Star} \in \mathrm{Abs}$, $\epsilon \in \mathrm{Abs}$, *then* C, Star *commute on* $M[T]$*: perturbation under constraint is the same as constraint under perturbation.*

Proof. Thanks to $\rho \in \text{Cst}$ (in the lower horizontal branch) and $\text{Star} \in \text{Abs},\ \epsilon \in \text{Abs}$ (in the right vertical branch),

$$\begin{array}{rcl} M[T] & \overset{\text{Star}[\epsilon,\#]\&}{\mapsto} & M^*_\epsilon[T^*_\epsilon] \\ C\downarrow & & C\downarrow \\ M[T][\rho[T]] & \overset{\text{Star}[\epsilon,\#]\&}{\mapsto} & M^*_\epsilon[T^*_\epsilon][\rho[T^*_\epsilon]] \end{array}$$

$\square$

Moreover, $C \in \text{Cst}$ iff $\rho \in \text{Cst}$ if $S \in \text{Cst},\ R \in \text{Cst}$ (def. 3.6). To satisfy the hypothesis of (the. 3.4),

$$\text{Star} \in \text{Abs}!\ \delta \in \text{Abs},\ \epsilon \in \text{Abs}! \tag{3.2}$$

$$S \in \text{Cst}!\ R \in \text{Cst}!\ \rho, C \in \text{Cst}\,. \tag{3.3}$$

Theorem 3.4 is implicitly used in thermodynamics, as a principle, asserting that, in a state to state transformation, like a monothermal compression, the final value of a state function, like energy, does not depend on the order of perturbation (compression) and constraint (constant temperature). $C \in \text{Cst}$ means that compression does not affect the reservoir temperature.

If $\text{Plus} \in \text{Abs},\ R \in \text{Abs}$, then, considering that Plus, $R \circ T$ semantically match functions,

$$\underline{M_1[T] + M_2[T]} = \underline{M_1[T]} + \underline{M_2[T]}, \qquad \underline{R \circ T} = R[\underline{T}]$$

and, with perturbation,

$$(\underline{R \circ T})^*_\epsilon = \underline{(R \circ T)^*_\epsilon} = \underline{R \circ T^*_\epsilon} = R \circ T^*_\epsilon[\rho[T^*_\epsilon]] = R[\underline{T^*_\epsilon}] \triangleright \underline{R}^\star_\epsilon .$$

Let $T, T^*_\epsilon \in \mathcal{C}[R,S],\ \delta[T] \notin \mathcal{C}[R,S]$. By (def. 3.7),

$$\begin{array}{rcl} T^* & = & T + \delta[T] \\ \downarrow & C & \downarrow \\ T^*[\rho[T^*]] & \neq & (T + \delta[T])[\rho[T]] = T^*[\rho[T]] \end{array}$$

$\underline{\#}$ does not convey identity, C expresses no function (the. 2.1). Equivalently, $\underline{T^* - T}$ matches the semantic pattern of (def. 3.7) in two ways, $T_{\_} \rightarrow T$ or $T_{\_} \rightarrow T^*$ (a donkey problem again).

Table 2: constraint and perturbation shorthands

$x_{\_}$	$\underline{x} =$	$\underline{x}\triangleright$
	$\rho[T]$	ρ
T	$T[\rho]$	$\underline{T}$
T'	$T'[\rho]$	$\underline{T}'$
	$\rho[T^*_\epsilon]$	$\rho^\star_\epsilon$
T^*_ϵ	$T^*_\epsilon[\rho^\star_\epsilon]$	$\underline{T}^*_\epsilon$
$R \circ T^*_\epsilon$	$R[\underline{T^*_\epsilon}]$	$\underline{R}^\star_\epsilon$
$\delta[T]$	$\delta[T][\rho]$	$\underline{\delta[T]}$

4 Design

4.1 Complexity reduction

Let $T \in \mathcal{T}$. The user-experimentalist wants to observe (to make observable) not the whole of T, but some less complex material property $\sigma[T] \in \mathbb{R}$. He finds out an input in T, on which $\sigma[T]$ depends,

$$\exists\,\text{Function}[\epsilon \in [0,1], T^*_\epsilon \in \mathcal{T}] \triangleright T^*_\#\& \in \mathcal{T}_1,\ T = T^*_0,\ \partial_\epsilon \sigma[T^*_\epsilon] \neq 0,$$

and a state function $R \in \mathbf{F}[\mathcal{T}, \mathbb{R}]$, such that $R[T^*_\epsilon]$ is observable for $\epsilon \in [0,1]$, hence the synthetic experiment $\{(R \circ \#)\&, T^*_\#\&\}$. The problem is, to find an interpretation P from $R \circ T^*$ to $\sigma[T]$, yielding the experiment $\{\sigma[\#[0]]\&, T^*_\#\&\}$, of output $\sigma[T^*_0] = \sigma[T] = P[R \circ T^*]$. The pure function $\#[0]\&$ expresses the Dirac measure, thus linked to analytic experiment.

I assume a linear gauge $\sigma[T]\#\&$ of $R[T]$, sensitive to $\Phi[T]$, and

$$\sigma \in \text{LFs}[\mathbb{R}],\ \sigma \in \text{Cst},\ \delta[\sigma[T]] = -\sigma[T],\ T \in \text{Uns}\,.$$

$$\begin{aligned}
\forall \epsilon \in [0,1],\ \sigma[T^*_\epsilon] &= \sigma[T] + \epsilon\sigma[\delta[T]] = \sigma[T](1-\epsilon), \\
R[T^*_\epsilon] &= \sigma[T]\Phi[T^*_\epsilon](1-\epsilon), \\
(R \circ T^*)'[0] &= \partial_\epsilon R[T^*_\epsilon]/.\epsilon \to 0 \\
&= \sigma[T](d\Phi[T][\delta[T]] - \Phi[T]).
\end{aligned}$$

If $d\Phi[T][\delta[T]] = 0$ and $\Phi[T]$ is observable, then $-\Phi[T]^{-1}\#\&$ interprets $(R \circ T^*)'[0]$ to $\sigma[T]$, as expected.

In the case [8], $\sigma[T]$ cannot be so easily evaluated; $\sigma[T]$ is a component of T:

$$T = \{\sigma[T], \widetilde{T}\},\ \widetilde{T} \in \mathrm{Cst}!$$
$$\delta[T] = \{-\sigma[T], 0\},\ T^*_\epsilon = \{\sigma[T](1-\epsilon), \widetilde{T}\}.$$
$$\forall \epsilon \in [0,1],\ R[T^*_\epsilon] = \sum_{t=0}^{\infty} r_t[\sigma[T], \widetilde{T}]\epsilon^t.$$

Instead of $\sigma[T]$, $r_t[\sigma[T], \widetilde{T}],\ t = 0, 1\ldots$ are observable (after another interpretation, that is interpolation). The user has not only an inverse problem, but also the problem that $r_t[\#, \widetilde{T}]\&$ is a private interpretation from $\sigma[T]$.

The signal to noise ratio is $r_1[\sigma[T], \widetilde{T}]/r_0[\sigma[T], \widetilde{T}]$; the shielding ratio is $r_2[\sigma[T], \widetilde{T}]/r_1[\sigma[T], \widetilde{T}]$ (null if $\Phi \in \mathrm{Uns}$, from (4.1)). The user wants to minimize the latter and to maximize the former, while avoiding large variations of R (for private reasons). Therefore, he relaxes $\widetilde{T} \in \mathrm{Cst}$ and finds an input in $\widetilde{T}$, on which $R[T]$ depends,

$$\exists \widetilde{T}^*_{\#}\&,\ \widetilde{T} = \widetilde{T}^*_0,\ \partial_\epsilon R[\{\sigma[T], \widetilde{T}^*_\epsilon\}] \neq 0,$$
$$T_2[x, y] = \{\sigma[T](1-x), \widetilde{T}^*[y]\},\ \partial_{x,y} T_2[x, y] = 0,$$
$$\widetilde{R}[x\_] = R[x\_] - R[T],$$

such that $R[T_2[x, y]]$ is observable for $x, y \in [0, 1]$, hence the experiment $\{\widetilde{R} \circ \#\&, T_2\}$. For $T_2[x, \#]\& \in \mathcal{C}[\widetilde{R}, S]$ (def. 3.7), $\widetilde{R} \circ T_2$ is interpreted by $S[\mathrm{Function}[y, \#[x, y]]]\&$ to $\rho[T_2[x, \#]\&]$:

$$\rho[T_2[x, \#]\&] = S[\widetilde{R} \circ T_2[x, \#]\&] = S[\mathrm{Function}[y, \widetilde{R} \circ T_2[x, y]]].$$

Synthetically, $\widetilde{R} \circ T_2$ is interpreted by $\mathrm{Function}[x, S[\mathrm{Function}[y, \#[x, y]]]\&]$ to $\mathrm{Function}[x, \rho[T_2[x, \#]\&]]$.

$$\forall \epsilon \in [0,1],\ \rho[T_2[\epsilon, \#]\&] = \sum_{t=0}^{\infty} \widetilde{r}_t[\sigma[T], \widetilde{T}^*_{\#}\&]\epsilon^t.$$

Instead of $\sigma[T]$, $\widetilde{r}_t[\sigma[T], \widetilde{T}^*_{\#}\&],\ t = 0, 1\ldots$ are observable, without large variations of R, but with a worse privacy problem, since the private variable is now functional.

Complexity constraints leads to observe another material property than initially contemplated; ignoring this fact is a frequent cause of misunderstanding.

Experiment design follows from calculability. According to (def. 3.3) and [3, 3.1.2], $\sigma[T]$ is observable if the formal system can evaluate $\mathrm{step}_0[T]$,

$$\begin{aligned}
&\mathrm{step}_{t_}[T]/;\mathrm{test}_t[\widetilde{\sigma}_t[T]] \to t, && \mathrm{step}_{t_}[T] \to \mathrm{step}_{t+1}[T],\\
&\widetilde{\sigma}_{t_}[T] \to P_t[\sigma,T][\widetilde{\sigma}_{t-1}[T]], && \mathrm{test}_{t_}[T] \to P_t[\mathrm{test},T][\mathrm{test}_{t-1}[T]].
\end{aligned}$$

This experimental program is such that t is incremented only after the test has failed, so that $\mathrm{step}_0[T]$ returns the minimum t passing the test (also a fixed point of the pure function step). For $t \in \mathbb{N} \smallsetminus \{0\}$, $P_t[\sigma,T]$ is a gauge of $\widetilde{\sigma}_t[T]$, sensitive to $\widetilde{\sigma}_{t-1}[T]$, or an interpretation from $\widetilde{\sigma}_{t-1}[T]$ to $\widetilde{\sigma}_t[T]$, public if $\partial_T P_t[\sigma,T] = 0$.

The user is responsible for the initial idea $\widetilde{\sigma}_0[T]$, for ruling P and for interrupting. He plays a game against nature (as in [9]) and P expresses his strategy. For example, if $\sigma[T]$ is the average of a random variable, then $P_t[\sigma,\#]$ means simply "try again",

$$P_t[\sigma,T][\widetilde{\sigma}_{t-1}[T]] \to \mathrm{Random}[T],$$

while $P_t[\mathrm{test},\#]$ should be some convergence criterion.

$\partial_t P_t \neq 0$ allows dynamic programming, or debugging: to have $\mathrm{step}_0[T]$ evaluated step by step and to decide P_t as late as possible, taking into account past experiments, that is *experience*.

4.2 Error reduction

If $\sigma[T]$ was already known (numerically), then the user could test accuracy by

$$\mathrm{test}_t[x\_] \to (|x - \sigma[T]| \leq e\_\mathrm{Real}).$$

Actually, $\sigma[T]$ is not usually known. test may ensure termination, but not accuracy: this all the ambiguity of "finishing a job". When experimenting has stopped at t, the error is, symbolically,

$$\widetilde{\sigma}_t[T] - \sigma[T] = \sigma^*[T^*] - \sigma[T] = \sigma[T^*] - \sigma[T] + \delta[\sigma][T^*].$$

The realization error is $\sigma[T^*] - \sigma[T]$; the programming error is $\delta[\sigma][T] \approx \delta[\sigma][T^*]$. Good realization is ruined by bad programming, and conversely; both errors ought to be simultaneously small.

Table 3: the lexicon of dependence and independence

non-universal	universal
specialized	unspecialized
implicit	explicit
concrete	abstract
private	public
variable	constant
shielded	unshielded

Table 4: the lexicon of experiment

constraint	controllable	eigeninput	error
experiment	exp. program	gauge	flux
input	interpretation		
mat. property	mat. system	observable	output
perturbation	servo-function	state	state function

5 Conclusion

5.1 Rhetoric of experiment

The understanding of dependence, or semantic pattern matching, is essential in experiment, while no formal system can completely support it. Now that we have got a rather precise language for dependence (tab. 3) and experiment (tab. 4), let us play with it, for public, non-technical, expert-free discussion.

Dependence is not only relation but action, or calculability. Relation without action easily leads to non-sense.

- An electrical engineer needs to evaluate a current i, from the heat ri^2 released by a resistor. The initial experiment is $\{r\#^2, i\}$. Assuming that r is observable (which is another problem), ri^2 is interpreted by $(r^{-1}\#)^{1/2}\&$ to $|i|$. The sign of the current cannot be determined (unless some other kind of gauge is used). But ri^2 also matches $(k\_i)_s$, with $k \to ri$, and ri^2 is interpreted by $k^{-1}\#\&$ to i. This interpretation fails, because k is not observable.

- More generally, any observable $M[T]$ cannot be interpreted to $\sigma[T]$ by P, $P[M[T]] = \sigma[T]$, unless P is calculable.

Implicit inputs can play tricks on the user-experimentalist.

- A heating engineer needs to evaluate a temperature θ in a furnace. He uses as a gauge a thermocouple, the output of which is interpreted by an amplifier P. Unfortunately, the amplifier is directly heated by the furnace, so that its gain depends on θ. P is a private interpretation from the thermocouple output or the gauge of some material property, hardly related to temperature. This can be corrected, by either insulating the amplifier or expliciting temperature in gain.

- Two surveyors, one working near the North pole, the other near the equator, need to evaluate the sum of angles of triangular oceanic plots. They may disagree, until they understand that the laws of geometry, expressed by Φ, depend on curvature, decreasing with latitude. This is a typical problem of hidden variable, or restrictive semantic pattern matching.

- Hidden variables have been similarly suspected in quantum mechanics, in particular by Einstein.

To name a gauge after the material property it was intended to evaluate, instead of the material property it actually evaluates, is to take our desires for realities. Without interpretation, we can hope to observe nothing but natural numbers, as in sheep counting, and, hopefully, real numbers, as read on rules.

- An accelerometer does not evaluate acceleration, but displacement, to be interpreted to acceleration by second derivation.

- A flow meter does not evaluate flow rate, but some concrete material property, like a turbine angular velocity, hence "calibration" problems.

- A hygrometer does not evaluate humidity, but the length of a hair. Of course, hair-length meters would not sell as well as hygrometers.

- A lie detector…

Table 5: formal systems?

	T	$\Phi[T]$	$R[\Phi[T]]$
universal	rules, data	trace	output
quantum	potential, mass	wave function	probability
transport	source, cross section	particle flux	average number
mechanical	initial condition, mass	motion	position
human	intention, knowledge	thought	speech
economic	needs, goods	market	price

5.2 Universality, not reductionism

Many formal systems seem to occur naturally (tab. 5) and to behave similarly. However, their universal reduction (def. 2.1) is far from being effective, because of thick complexity barriers. Universality does not mean reductionism.

Thanks

Thanks to Rémi Barrère, for many fruitful discussions, about mathematics, computer science and philosophy.

References

[1] Jean-Paul Delahaye. *Information complexité et hasard*. Hermes Science, 1999.

[2] Stephen Wolfram et al. *Mathematica*. Int. Thomson Pub., 1997.

[3] Rémi Barrère. *Mathematica, calcul formel et programmation symbolique pour l'informatique scientifique*. Vuibert, 2002.

[4] Gregory J. Chaitin. *The limits of mathematics*. Springer, 1998.

[5] Michel Serres. *Éléments d'histoire des sciences*, chapter Gnomon, les débuts de la géométrie en Grèce. Bordas, 1986.

[6] Jason Harris. Semantica: Semantic pattern matching in mathematica. *The Mathematica Journal*, 7(3), 1999.

[7] Ludwig Wittgenstein. *Tractatus logico-philosophicus.* Routledge & Kegan Paul, 1922.

[8] Pierre Albarède. Weighing operator perturbation from quasi-critical source system response. *http://arxiv.org/abs/math-ph/0007026*, 2000.

[9] Pierre Albarède. Fate stochastic management and policy benchmark in 421, a popular game. *http://arxiv.org/abs/math.OC/0007129*, 2000.