BEYOND THE "INFORMATION WALL" WITH DISCRETE INFORMATION PROCESSING

Gérard Pinson

Abstract

In the flowchart : (*Objects* $\rightarrow$) *Symbols* $\rightarrow$ *Words* $\rightarrow$ *Source Texts* $\rightarrow$ *Code Text* , several measures of object content information are possible : Shannon entropy H, Von Neumann entropy S for quantum objects, algorithmic complexity or Kolmogorov entropy K, etc.

Some typical features are appearing by a cross analysis of this quantities. Two fundamental hypothesis are invoked here : 1) the distinction of information elements from each other requires a discernability property, like countability or prefixed strings ; 2) shared information content of two objects r and s is symmetrical.

On one hand, the mutual prefix algorithmic complexity $K(s{:}r)$ is a countable quantity but not symmetrical. To achieve symmetry, we must explicitly carry out the program v^* which encodes r to calculate $K(s|v^*) = k(s|r)$. This is the Chaitin complexity or "explicit conditional prefix complexity", with $k(s|r) < K(s|r)$: the compressibility is maximal, but $k(s|r)$ is not countable.

On the other hand, because of quantum entanglement (violation of Bell inequalities), the Von Neumann conditional entropy $S(\rho_\xi|\rho_\eta)$ can be negative (on non-orthogonal states), while $S(\rho_\xi|\rho_\eta) < H(\xi|\eta)$. The compressibility of quantum information is maximal, but information in quantum states is a continuum, while measurement remains discrete.

Extending information quantities to continuous sample spaces becomes arbitrary difficult and needs a "sampling" where is lost information on order, i.e. on origin and position of data, which are defined within a recursive permutation. Underlying any decipherable alphabetic communication process (i.e. distinguishable, discrete, transmitted), there is a non-countable, non-transmitted background. This makes a meaningful difference between *"a priori"* and *"accessible"* information.

But the translation of continuous frames into discrete signals is a well known topic of Digital Signal Processing (DSP). We show that similar principles would be possible on "Discrete Information Processing", based on the same principles, with few modifications.

It is possible to define properties of a “Super-Boole” computer matching this two kinds of information, combining bits and forms, i.e. arithmetic calculus and Boole algebra principles with harmonic analysis and Lie algebra.

In fact, this principles point out quantum calculus. But in the face of hardware difficulties making quantum computer, and software difficulties (calculation time) simulating with boolean

computer, we suggest, to study this systems, a circuit diagram with DSP processors. Feasibility evaluation is proposed from TI's IC fixed-point TMS320Cxx.

Independance scale principle may be applied : this principle of calculus does not modify the shape of a form, only its amplitude. So normalized FFT F_1 is seen as NOT operator, convolution as OR, product as AND, normalized Dirac distibution $\delta_1(x)$ and $\mathbf{1}(x)$ as boolean Ø and 1 respectively.

Hardware applications like timing, computational speed, wiring are examined. For instance, boolean operation XOR here means a new DSP operation, where signal and spectrum are multiplied. It is a new device for programming.

From separability of M-dimension FT property, it is possible to construct arithmetic acting on M-dimension numbers whose "bits" are themselves signals of N-samples, with a very important new feature : there are two “zero”, the real number “0” and the “super-boolean” zero normalized distribution δ_1. Other extensions of boolean calculus, as non-commutability and Lie algebra, modulation and modal logic, etc, are to be considered. The main advantage in relation to quantum computer is here it is extremely easy keeping coherence, with SYNC inputs.

****

1. NOTATIONS, DEFINITIONS AND CLASSIFICATION OF SOURCES OF INFORMATION

Notations :

$\leftrightarrow$	bijection
ε	empty string
$a_1\ a_2\ a_3$	concatenation of a_1, a_2 and a_3
$< s , w >$	pairing function
qlog x	quasilogarithm function, defined as : $\text{qlog}\, x = \begin{cases} \log_2 x & \textit{if}\ \ x > 1 \\ 0 & \textit{if}\ \ x \le 1 \end{cases}$
$\text{qlog}^{(k)}\, x$	iterations of qlog x : $\text{qlog}^{(1)} x = \text{qlog}\ x$; $\text{qlog}^{(k+1)}\, x = \text{qlog}\ (\text{qlog}^{(k)}\, x)$
λ	number of iterations : $\lambda = \max \{ k \ge 1 : \text{qlog}^{(k)}\, x > 0\}$
$Q_\lambda(x)$	sum of iterated quasilogarithms : $Q_\lambda(x) = \text{qlog}^{(1)} x + \text{qlog}^{(2)} x + ... + \text{qlog}^{(\lambda)} x$
$\lvert \mathbf{a} \rangle$	vector
$\langle \mathbf{a} \lvert \mathbf{b} \rangle$	scalar product
Tr	Trace operator

Let the general diagram :

(*Objects* →) *Symbols* → *Words* → *Source Texts* → *Code Texts*

1.1. OBJECTS

- Elements O_i
- **Object** : $\boldsymbol{O}$ = set { $O_i \mid O_1 , O_2 , ... , O_i , ... , O_N , ...$ }

1.2. SYMBOLS – Description of objects by symbols.

Subsequently, we will not need objects any more : information theories are theories of symbols, whatever the nature of objects that, in any case, does not appear in the equations.

- $O_i \leftrightarrow$ **symbol** χ_i

Note this symbols are discernible *by definition.*

- **Library** : Ξ = set of symbols { χ_i | χ_1 , χ_2 , ... , χ_i , ... , χ_N , ... }

  N = card (Ξ) , finite or infinite

  Sign = variable ξ ; $\xi \in \Xi$

- **Message** : $\sigma = \xi_1 \xi_2 \dots \xi_k \dots \xi_m$ (concatenation of symbols)

  $\sigma \in \Xi^m$

  length : $m = l(\sigma)$

  Concatenation $\mathcal{E}$ is defined as : $\forall$ ordered $(\rho,\sigma) \in \Xi^* \times \Xi^*$, $\exists\ \rho\sigma = \mathcal{E}(\rho,\sigma) \in \Xi^*$.

  This operator is associative and ε is a neutral element. Ξ^* is a closed set for this relation.

- **Dictionary** : $\Xi^* = \bigcup_{m \geq 1} \Xi^m$

- **Symbolic source** : $\Sigma \subset \Xi^*$

  Symbols being discernible by definition, note that a symbolic source acts as a prefixed alphabetic source (see later) : the distinction between two symbols in a text is always possible. Each symbol is like a string of one character chosen in an infinite (and countable) alphabet.

1.3. WORDS – Transcription from symbols to words

- **Letters** X_l

- **Alphabet** : $\boldsymbol{A}$ = finite set of letters { X_l | X_1 , X_2 , ... , X_l , ... , X_Q }

  Q = card ($\boldsymbol{A}$) , finite

  Character = variable x ; $x \in \boldsymbol{A}$

- **Word** or **String** : w or $s = x_1 x_2 \dots x_j \dots x_n$

  $s \in \boldsymbol{A}^n$

  length : $n = l(s) = \log_Q s + O(1)$

- **Vocabulary** : $A^* = \bigcup_{n \geq 1} A^n$

- **Alphabetic source** or **Language** : $\boldsymbol{L} \subset \boldsymbol{A}^*$

  $\boldsymbol{A}^*$ is a free monoid under concatenation (the empty string is the null element).

  A transcription $\mathcal{T}$ is a coding relation from Ξ to $\boldsymbol{A}^*$ defined on a subset of Ξ, called the d*omain* of $\mathcal{T}$ (write : *dom*($\mathcal{T}$)). We assume that *dom*($\mathcal{T}$) = Ξ, and we say that $\mathcal{T}$ is *total :*

$$\forall \xi \in \Xi \xrightarrow{\mathcal{T}} \exists \mathcal{T}(\xi) \in \boldsymbol{A}^*$$

Conversely, we define a language $\boldsymbol{L}$ for a given transcription $\mathcal{T}$ by the following statement :

$$\boldsymbol{L} = \{s \mid s \in \boldsymbol{A}^* \, ; \, \exists \xi \, : \, s = \mathcal{T}(\xi)\}$$

- sign $\xi_k \leftrightarrow$ word s_k ; message $\sigma \leftrightarrow$ **Text** $t = s_1 \, s_2 \ldots s_k \ldots s_m \in \boldsymbol{L}$

1.4. ALPHABETIC SOURCES – Coding.

- **weak source** :

A *weak source* is a singular code where the transcription $\mathcal{T}$ is not injective :

$$\xi_1 \neq \xi_2 \;\not\Rightarrow\; s_1 \neq s_2$$

or : $\exists \, \xi_1 , \xi_2 : \; \xi_1 \neq \xi_2$ and $\mathcal{T}(\xi_1) = \mathcal{T}(\xi_2)$

So (as $\mathcal{T}$ is total) :

$$\text{card}\,(\boldsymbol{L}) < \text{card}\,(\Xi)$$

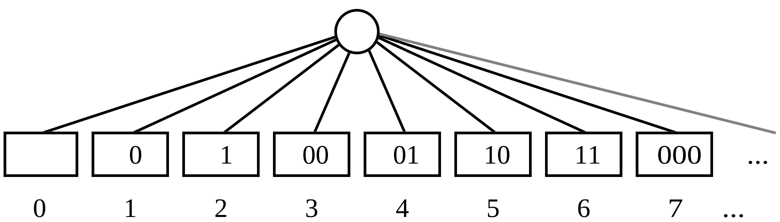
There are not enough words to describe the set of symbols !

From a general point of view, if we suppose the value of an information content I depends on the number of possibilities of designating it (i.e. labeling, marking up, ...), this inequality means that :

$$I_L < I_\Xi$$

i.e., average information content of a word (a label, a mark...) is smaller than average information content of a symbol (a form, an object, ...).

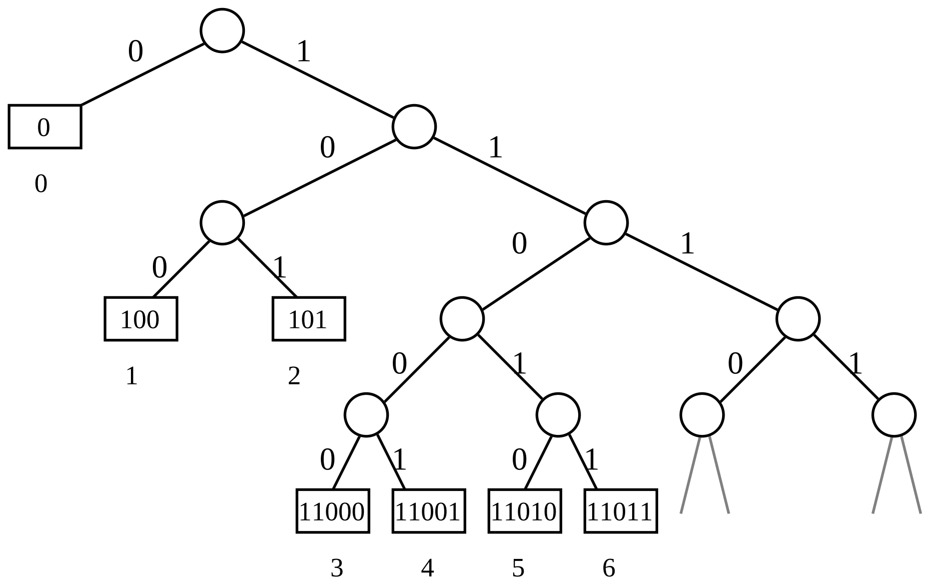
- Regular source :
  - ◇ **Lexical source**. Words are not delimited. (= not decipherable code). Notation : **C**



- *Fig. 1* : The space structure of **C** is a *bunch* or a *star* : no element is "more important" than an other. Partial ordering is =, i.e. $s \leq w$ iff $s = w$. [Uspensky, 1992].

  - ◇ Decipherable source : words are delimited, with :

- special character (as "#" for instance...)
- constant length *n*
- prefix code (or instantaneous, irreducible code) => **Prefixed source** : words are autodelimited. Notation : **K**



- *Fig. 2* : The space structure of **K** is a *tree*, with Kraft inquality : $\sum 2^{-l(s)} \leq 1$
- Partial ordering ≤ is defined as follows : $s \leq w$ iff *s* is a prefix of *w*, i.e. $\exists\, t : st = w$

1.5. ENCODING – Coding from source-texts to code-texts

In the framework of Turing machines, giving a program is giving a machine. Let $C_\Phi(s)$ the *Kolmogorov complexity* defined as the length of a smallest program that halts and ouputs *s* on an empty input [Li, Vitànyi, 1997]. A program Φ has *additive optimality* property for a class of programs *P* iff :

$$\forall \Phi' \in P,\ \exists c_{\Phi'},\ \forall s,\ C_\Phi(s) \leq C_{\Phi'}(s) + c_{\Phi'}$$

Let $C_{\Phi 1}(s)$ and $C_{\Phi 2}(s)$ Kolmogorov complexities defined for two different additively optimal programming languages. Then :

$$\forall \Phi_1, \Phi_2, \exists c\ \ \forall s\ \ |\, C_{\Phi 1}(s) - C_{\Phi 2}(s)\, | \leq c$$

So, Kolmogorov complexity is defined within a $O(1)$ for such a language.

- source **C** → code **C** : **algorithmic complexity** or **simple entropy**. Notation : C_C

- Definition : $C_C(s) = C_\Phi(s) = \min\{l(w) : \Phi(w) = s\}$
- Order of magnitude : $C_C(s) \approx l(s) + O(1)$
- C_C is denoted K_A in [Kolmogorov 1965], $\mathbb{BB}$ or KS in [Uspensky 1992], **KS** in [Dubacq 1998], C in [Li et Vitányi 1997]

• source **K** → code **C** : **uniform complexity** or **decision entropy**. Notation : C_K

- Definition : $C_K(s) = C_\Phi(s;l(s)) = \min\{l(w) : \Phi(l(s),w) = s_{1:k} \ \ \forall\ k \le l(s)\}$
- Order of magnitude : $C_K(s) \approx l(s) + O(1) \quad \forall s$
- This complexity is called *"uniform complexity"* in [Loveland 1969], *"decision complexity"* in [Zvonkin et Levin, 1970]. C_K is denoted $K_A(y^n;n)$ in [Loveland 1969], KR in [Zvonkin et Levin 1970], $\mathbb{BT}$ or KD in [Uspensky 1992], $C(x;l(x))$ in [Li et Vitányi 1997].

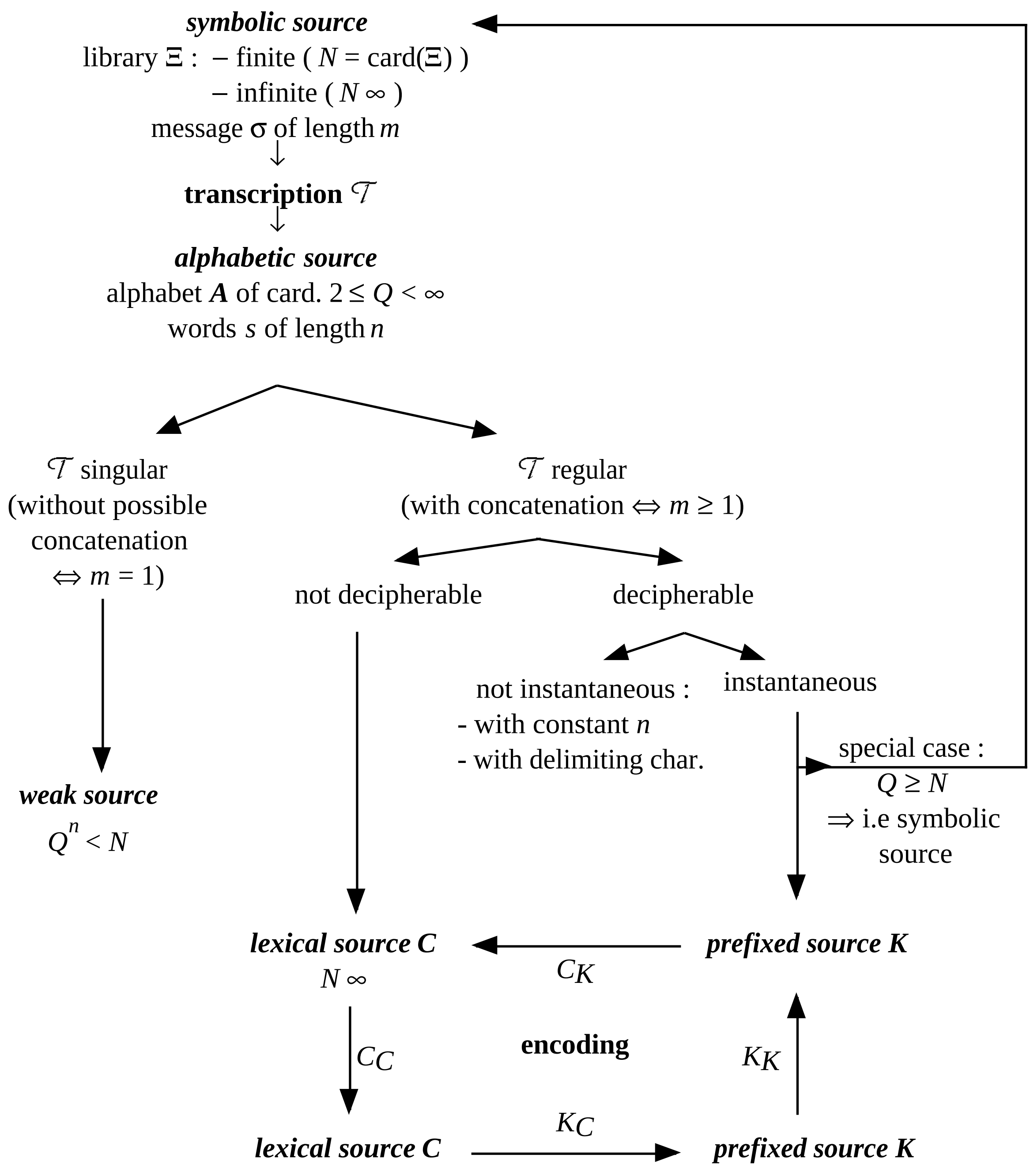
• source **C** → code **K** : **algorithmic prefix complexity** or **prefix entropy**. Notation : K_C

- Definition : $K_C(s) = C_\Phi(s) = \min\{l(w) : \Phi(w) = s\}$ with prefixed w
- Order of magnitude : $K_C(s) \approx l(s) + Q_\lambda(l(s)) + O(1) \approx Q_\lambda(s) + O(1) \ \ \forall s$
- K_C is denoted KP in [Levin 1976], $\mathbb{TB}$ or KP in [Uspensky 1992], KP in [Dubacq 1998], K in [Li et Vitányi 1997].

• source **K** → code **K** : **monotonic complexity** or **monotonic entropy**. Notation : K_K

- Definition : $K_K(s) = C_\Phi(s) = \min\{l(w) : \Phi(w) = s\}$ with prefixed s and w
- $K_K(s) \approx l(s) + \mathrm{O}(1) \ \ \forall s$
- K_K is denoted $\mathbb{TT}$ in [Uspensky 1992], *KM* in [Li et Vitányi 1997].

To sum up, see the following diagram. Note that, making distinction between symbolic source, whose library may be infinite, and alphabetic source, whose alphabet is always finite by definition, is equivalent – in the standard information theory – to make the distinction between encoding with finite or infinite alphabets. The reason of this distinction is very simple : from a finite set, we can construct a countable infinite set only, while from an infinite countable set, we can construct a non countable infinite set. So the information contents of the elements chosen among this sets are quite different.

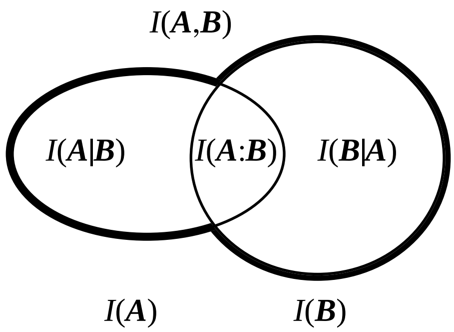


• *Fig. 3* : General diagram of sources of information

2. ENTROPIES : MAIN FEATURES

2.1. GENERAL DIAGRAM

Let two objects A and B, and their information contents $I\ (A)$ and $I\ (B)$. Then entropy Venn diagrams are well known (*Fig. 4*) :



Let x and y two sets of symbols which describe A and B. Information contents are functions of x and y. There are several kinds of functions :

$I\ (A) = H\ (x)$: classical or Shannon entropy

$I\ (A) = H_d\ (x)$: differential entropy (Shannon entropy for continuous distributions)

$I\ (A) = S\ (x)$: quantum or Von Neumann entropy

$I\ (A) = C_C\ (x)$: simple algorithmic complexity or Kolmogorov entropy

$I\ (A) = C_K\ (x)$: uniform complexity

$I\ (A) = K_C\ (x)$: prefix algorithmic complexity

$I\ (A) = k_C\ (x)$: explicit prefix algorithmic complexity or Chaitin entropy

$I\ (A) = K_K\ (x)$: monotonic complexity

From a general point of view, let $I(A)$ and $I(B)$ the measures of information contents or *entropies*, $I(A,B)$ information linked to $A \cup B$, $I(A{:}B)$ mutual information jointly shared by A and B in $A \cap B$. conditional entropy $I(A|B)$ (whose set representation is $A \cap \neg B$) is the information content of A knowing B.

Definition : in $I(A|B)$, call A the "proper" information and B the "context information" or *identification content* of A. Here, B is an external knowledge, a kind of "catalyst" that is used to calculate $I(A)$, but the information content $I(B)$ is not included in the information content $I(A)$.

Then the usual relationships (but not always true, as it is well known) are :

- *symmetry* : the measure of mutual content of information is a commutative operation :

$$I(A{:}B) = I(B{:}A) \qquad (2.1)$$

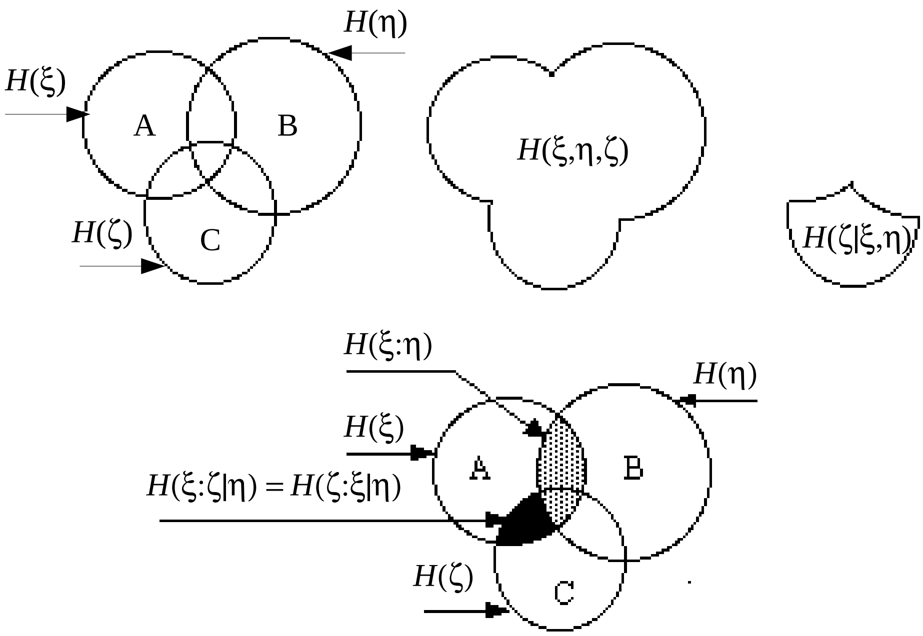
- *additivity* : the information contents add :

$$I(A,B) = I(B) + I(A|B) = I(A) + I(B|A) \quad (2.2)$$

$$I(A) = I(A{:}B) + I(A|B) \quad (2.3)$$

$$I(B) = I(A{:}B) + I(B|A) \quad (2.4)$$

And, from entropy Venn diagrams, it is possible to derive classical entropic inequalities for more than two variables. For instance, with three sets (*Fig. 5*) :



$$H(\xi,\eta,\zeta) \leq H(\xi) + H(\xi) + H(\xi) \quad (2.5)$$

$$H(\zeta|\xi,\eta) \leq H(\zeta|\eta) \quad (2.6)$$

$$H(\xi{:}\eta,\zeta) = H(\xi{:}\eta) + H(\xi{:}\zeta|\eta) \quad (2.7)$$

$$H(\eta,\zeta{:}\xi) = H(\eta{:}\xi) + H(\zeta{:}\xi|\eta) \quad (2.8)$$

From what one derives the "strong" additivity property of H :

$$H(\xi,\eta,\zeta) + H(\eta) \leq H(\xi,\eta) + H(\eta,\zeta) \quad (2.9)$$

And inequalities on mutual entropies (in the case of uniform distributions, one has $H(\xi) = H(\eta) = H(\zeta) = 1$) :

$$H(\xi{:}\eta) + H(\xi{:}\zeta) - H(\eta{:}\zeta) \leq H(\xi) \quad (2.10)$$

$$H(\xi{:}\eta) - H(\xi{:}\zeta) + H(\eta{:}\zeta) \leq H(\eta) \quad (2.11)$$

$$- H(\xi{:}\eta) + H(\xi{:}\zeta) + H(\eta{:}\zeta) \leq H(\zeta) \quad (2.12)$$

These results can be extended to more than three sets. A resulting inequality for four sets becomes, for instance :

$$H(\xi'{:}\eta) + H(\xi{:}\zeta) - H(\eta{:}\zeta) + H(\xi{:}\xi') \leq 2 \quad (2.13)$$

2.2. GENERAL OVERVIEW OF ENTROPIES AND COMPLEXITIES : tab. 1 *(next page)*

1	*Features*	*Description*	*Relationships*	*H*	*Hd*	*S*	*CC*	*CK*	*KC*	*kC*	*KK*
2	*Enumerability*	The set of elementary information contents is countable		**Y**	**N**	**N**	**Y**	**Y**	**Y**	**N**	**Y**
3	*Minimum*	Entropy is equal to zero iff the message is known	$I(0) = I(1) = 0$	**Y**	**N**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**
4	*Maximum*	There is an upper bound function of entropy	$I(x) \le f(x)$	**Y**	**N**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**
5	*Concavity*	Entropy of a mean is greater than mean of elementary entropies	$I(\sum(kx)) \ge k\sum I(x)$	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**
6	*Monotony*	Entropy of the whole is greater than entropy of the parts	$I(x,y) \ge I(x)$ $I(x,y) \ge I(y)$	**Y**	**N**	**N**	**N**	**Y**	**Y**	**Y**	**Y**
7	*Additivity*	Let two objects A and B : infomation on A only reduces uncertainty on B	$I(x \mid y) \le I(x)$ $I(y \mid x) \le I(y)$ $I(x,y) = I(x) + I(y \mid x)$	**Y**	**Y**	**Y**	**N**	**Y**	**N**	**Y**	**?**
8	*Subadditivity*	Correlations between two parts of an object only reduce the whole entropy	$I(x,y) \le I(x) + I(y)$	**Y**	**Y**	**Y**	**N**	**N**	**Y**	**Y**	**?**
9	*Strong subadditivity*	If two objects AB and BC (union ABC) have a common intersection B, then :	$I(x,y,z) + I(y) \le I(x,y) + I(y,z)$	**Y**	**Y**	**Y**	**?**	**?**	**?**	**?**	**?**
10	*Sign of I(x)*	Entropy is positive or equal to zero	$I(x) \ge 0$	**Y**	**N**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**
11	*Sign of I(x,y)*	id.	$I(x,y) \ge 0$	**Y**	**N**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**
12	*Sign of* $I(x \mid y)$	Conditional entropy is positive or equal to zero	$I(x \mid y) \ge 0$	**Y**	**N**	**N**	**Y**	**Y**	**Y**	**Y**	**Y**
13	*Sign of I(x : y)*	Mutual entropy is positive or equal to zero	$I(x : y) \ge 0$	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**
14	*Symmetry*	Information content of A about B is the same as information content of B about A	$I(x : y) = I(y : x)$	**Y**	**Y**	**Y**	**N**	**N**	**N**	**Y**	**?**
15	*coordinates system independance*	There is no variability of the entropy under transformations of coordinate systems	$x \longrightarrow x' : I(x') = I(x)$	**Y**	**N**	**Y**	**N**	**N**	**N**	**N**	**N**
16	*Randomness*	There is a probability measure	p(x) is defined	**Y**	**Y**	**Y**	**N**	**N**	**Y**	**Y**	**Y**

Tab 1, notes :

Y : true ; N : false ; ? : not examinated in this study.

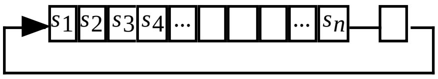
For recent overview on entropy, see for instance : [Wehrl, 1978], [Preskill, 1998], [Li, Vitànyi, 1997], [Uspensky, Shen, 1996],

Row 2 : *S :* the sample space is not countable (Hilbert space), but the measure space is.

Row 15 : *S :* basis change

C_C *:* recursive permutation ; *but* Yes for $C_C(s \mid l(s))$

C_K , K_C , k_C , K_K : recursive permutation :



Row 16 : H_d *:* subject to convergence of integrals

K_K *:* extending to continuous sample space is arbitrary difficult (Ackermann function)

3. INEQUALITIES FROM ALGORITHMIC THEORY

3.1. EXPLICIT CONDITIONAL PREFIXED COMPLEXITY

Some typical features are appearing by *cross analysis* of this quantities, and are giving a better understanding of the concept of content information. Two fundamental hypothesis are invoked here, distinction and identification operators :

1°) Distinction of information elements from each other requires a discernability property. This property can be seen from different point of view. The basic statement is following : a set is countable if it exists a bijection with $\mathbb{N}$, whose elements are separated (and ordered) by definition. An other example is discernability of encoded strings, which requires for instance a prefix-code to split a whole text into separated words. In any case, distinction operator is a disjunction, that breaks up an object representation into its component parts. Call this principle *discernability.*

2°) Mutual information $I(\ :\ )$ between to objects is a *shared* information content, that invokes a symmetrical pairing $I(A{:}B) = I(B{:}A)$, where B works as a "reference" to identify A and vice versa. For instance, if $A = C \cup E$ and $B = D \cup E$, with $C \neq D$, then $I(E)$ is a common content information shared by B and A, or by A and B. So $I(E)$ is the same, we don't care about the order in the pair (A,B) or (A,B). Identification operator is a conjunction, that shares two object representations in a symmetrical relationship. Call this principle *information relativity* or *bipolarity.*

Now, let the following argument :

1°) Outside any hypothesis on objects, no symbol, no number includes "more information", is "more significant" than an other : all elements stand up at the same level, as in a "star" or a "bunch" diagram. Hence source texts are *a priori* lexical.

2°) But to transmit more than one information element between two sources, the communication process must be performed from discernible symbols to discernible strings. So code texts are prefixed. Therefore the measure of transmitted information contents is prefix complexity K_C.

The function $K_C(s)$ is not partial recursive (Noncomputability Theorem). We cannot calculate $K_C(s)$ given s. But given $<s,K_C(s)>$, we can enumerate all shortest programs for s, with a diagonal flowchart : first step of program 1, first step of program 2, second step of program 1, first step of program 3, etc. Let w^* the first one we find. From w^* we can compute $s = \Phi(w^*)$ and $K_C(s) = l(w^*)$. So w^* and $<s,K_C(s)>$ contain the same information although they are not identical strings (resp. v^* and $<r,K_C(r)>$).

3°) We consider now the relationships between two objects at least. So we must consider the quantities : $K_C(s)$, $K_C(r)$, $K_C(s,r)$, $K_C(s|r)$, $K_C(r|s)$, $K_C(s{:}r)$ and $K_C(r{:}s)$. Let $<s,K_C(s|r)>$ (resp. $<r,K_C(r|s)>$) the string that contains the same information than $<u^*,r>$ (resp. $<t^*,s>$), where u^* is the shortest program which computes s given r.

4°) Considering the relationships between two objects, bipolarity principle requires symmetry of mutual information : $I(r{:}s) = I(s{:}r)$.

5°) But it is well known that Kolmogorov complexity is not additive :

$$\text{(a) } K_C(s,r) \leq K_C(s) + K_C(r|s) \qquad (3.1)$$

To obtain an exact additivity property, we must replace the conditional s by $<s, K_C(s)>$ (equivalently, by w^* the shortest program for s) :

$$\text{(b) } K_C(s,r) = K_C(s) + K_C(r|<s, K_C(s)>) = K_C(s) + K_C(r|w^*) + O(1) \qquad (3.2)$$

It is very important to see this relationships are quite different : in the first one (a), we calculate the information content of r given s. In the second one (b), we calculate this information content of *r given not only s, but also the program that calculates s.* It is not just a data flow, it is a know how.

Chaitin [Chaitin 1969, 1990] defines the conditional complexity :

$$k_C(r|s) = K_C(r \mid \langle s,K_C(s)\rangle) + O(1) = K_C(r|w^*) + O(1) \quad (3.3)$$

$$k_C(s|r) = K_C(s \mid \langle r,K_C(r)\rangle) + O(1) = K_C(s|v^*) + O(1) \quad (3.4)$$

Call this *explicit prefixed complexity*. Let u^{**} the shortest program which computes s given v^* (resp. t^{**} for r given w^*). Of course, $k_C(s) = K_C(s)$ and $k_C(s,r) = K_C(s,r)$. We can formulate the additivity property as :

$$k_C(s,r) = k_C(s) + k_C(r|s) + O(1) \quad (3.5)$$

and so we obtain exact symmetry of mutual information :

$$k_C(s{:}r) = k_C(r{:}s) \quad (3.6)$$

<u>*But*</u> :

a) $k_C(r|s)$ is not co-enumerable (unlike the other algorithmic information measures) [Li, Vitànyi, 1997.]. Then $k_C(s{:}r) = k_C(r) - k_C(r|s)$ is not enumerable.

That is to say : *making quite explicit some implicit identification content (i.e : substituting x^* for x) leads to non-enumerability.*

b) It ensues from the definition of k_C the following inequality :

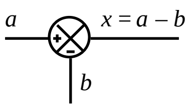
$$\left.\begin{aligned} K_C(s,r) &= K_C(s) + k_C(r \mid s) + O(1) \\ K_C(s,r) &\le K_C(s) + K_C(r \mid s) + O(1) \end{aligned}\right\} \Rightarrow k_C(r \mid s) \le K_C(r \mid s) + O(1) \quad (3.7)$$

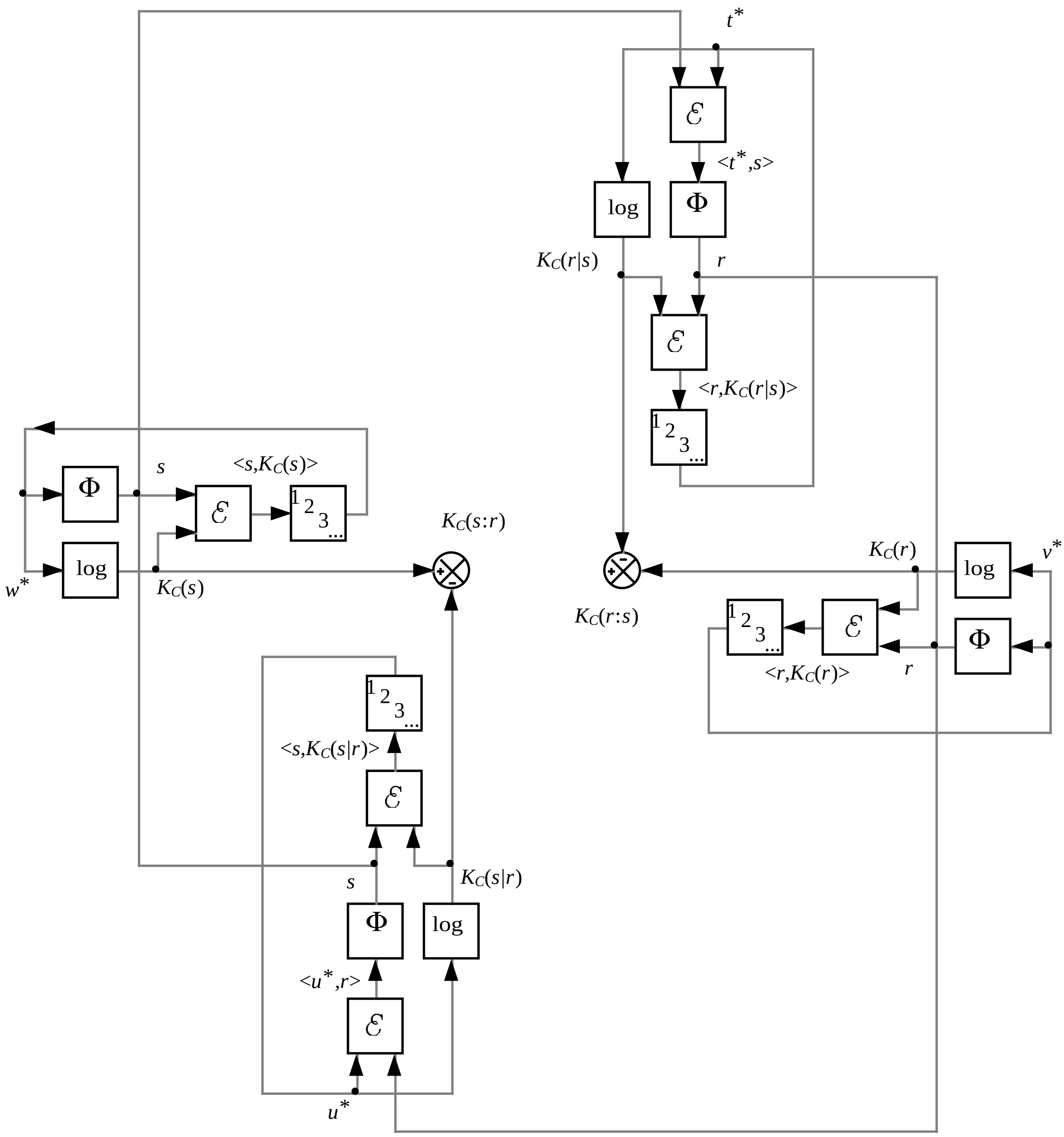
6°) We summarize the different relationships in the following figures, which describe the different calculus underlying this measures.

For prefixed complexity : see fig. 6

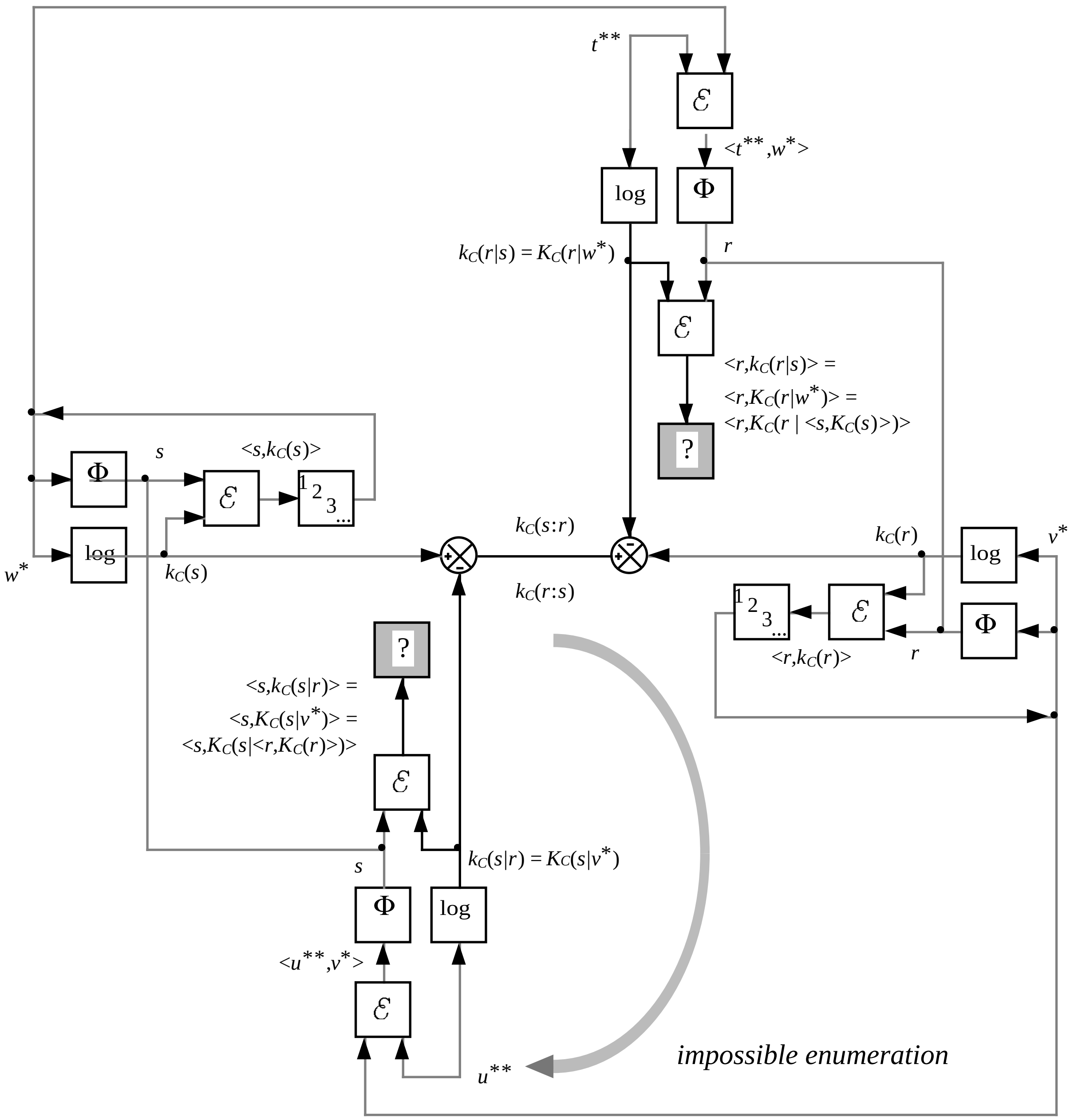
For explicit prefixed complexity : see fig. 7

Legend :

----------	*Discrete variable*	———	*Continuous variable*
	*subtraction*		
log	*Logarithm or length function*	Φ	*Turing machine*
$\mathcal{E}$	*Concatenation*	1 2 3 ...	*Diagonal Enumeration*


t*
<t*,s>
log
Φ
$K_C(r|s)$
r
ℰ
<r,$K_C(r|s)$>
s
<s,$K_C(s)$>
$K_C(s:r)$
w*
$K_C(s)$
$K_C(r:s)$
$K_C(r)$
v*
<r,$K_C(r)$>
<s,$K_C(s|r)$>
$K_C(s|r)$
<u*,r>
u*

• *Fig. 6*



•

• *Fig. 7*

To calculate *s* given *r* with the program *u***, we need to know *v**, the shortest program which computes *r*. But then we cannot perform an enumeration leading to such a *u*** program. There is no more equivalence between *u*** and $\langle s,k_C(s|r)\rangle$. We are facing a kind of "information wall", because :

a) on account of non recursivity of K_C function, we cannot calculate *u*** and thus $l(u^{**}) = k_C(s|r) = K_C(s|v^*) = K_C(s \mid \langle r,K_C(r)\rangle)$ from *s* given *r* and $K_C(r)$,

b) but we cannot enumerate *u*** given $l(u^{**})$.

3.2. MONOTONIC COMPLEXITY ON A CONTINUOUS SAMPLE SPACE

Definition : call "effective" a communication process between two sources where any *external* information content needed to calculate the information content *shared* by the sources is *really* available.

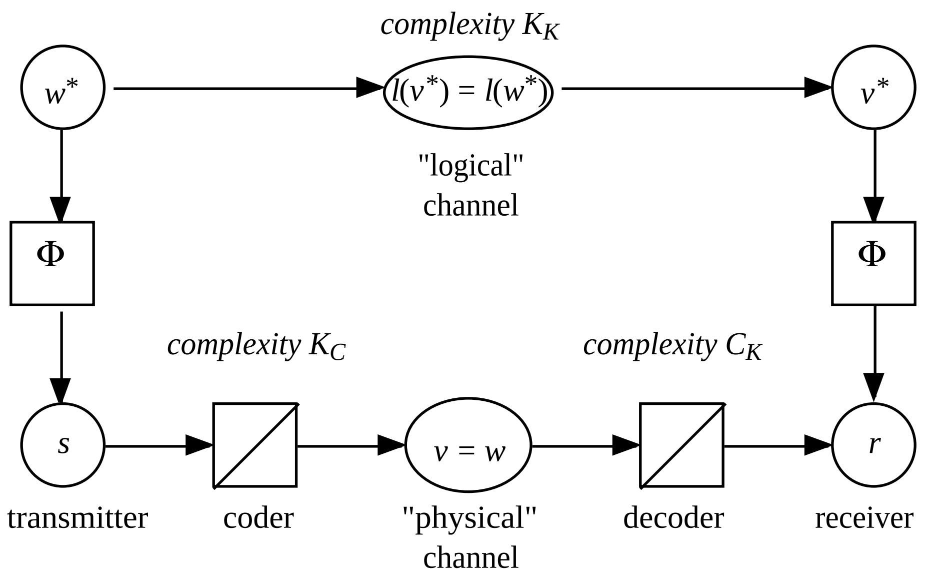
In the diagram of figure 6, calculating $K_C(s,r)$ then $K_C(s{:}r)$ requires knowing r and u^* to calculate $s|r = \Phi(<u^*,r>)$. So the measure of this knowledge itself is a prefixed complexity $K_C(<u^*,r>)$. (NB : it is easy to separate u^* from r in the pair $<u^*,r>$, as u^* is prefixed by definition, while r is not).

But the strings w^* and v^* in the diagram of figure 7 are prefixed, whereas s and r resp. are not in figure 6. Sharing this contents needs substituting : lexical → prefixed sources, for : prefixed → prefixed sources encoding diagrams. We have, substituting (ordinary information) → (explicit information) :

$$s|r = \Phi(<u^*,r>) \rightarrow s|r = s|v^* = \Phi(<u^{**},v^*>) \quad (3.8)$$

$$r|s = \Phi(<t^*,s>) \rightarrow r|s = r|w^* = \Phi(<t^{**},w^*>) \quad (3.9)$$

So the measures of this external knowledge are monotonic complexities $K_K(<u^{**},v^*>)$ and $K_K(<t^{**},w^*>)$ now (because $<u^{**},v^*>$ and $<t^{**},w^*>$ are prefixed strings). We may resume the communication diagram like this :



• *Fig. 8* : Modified Shannon diagram of communication

But in this case, u^{**} and t^{**} are underlying an extension of complexity to continuum, that lies in the pairs $<s,K_C(s|v^*)>$ and $<r,K_C(r|w^*)>$. As we are in the continuous case, the complexities may be defined on a binary tree, or on the tree consisting of all words in an infinite countably alphabet, like $\mathbb{N}$, if one takes $\mathbb{N}$ as an alphabet [Uspensky, 1992, p 101. See also Uspensky, Shen, 1996, for a more detailed demonstration]. That is to say : according to the definitions we wrote in § 1.1, this is no more an alphabetic source under the meaning we have laid out (alphabetic transcription based upon a finite alphabet), but we have to deal now with a *symbolic source,* whose symbols are the programs themselves (running on the reference monotone machine *U*).

They are two ways to calculate information content of an object, from the length of the shortest program, or from the negative logarithm of the universal probability. In the discrete case, it turns out it is the same (for prefix machines). Not in the continuous case.

On one hand, define *a priori complexity* K_{K1} , based upon a semi-measure of probability on a continuous sample space :

$$K_{K1}(s) = -\log(\Pi(s)) \quad (3.10)$$

This is the evaluation *a priori* of all possibilities we have, in practice, to designate a particular item in such a set, i.e. a symbolic source.

On an other hand, define *monotonic complexity* $K_{K0}(s)$, based upon an algorithmic probability $\pi(s)$ obtained from coding theorem :

$$K_{K0}(s) = -\log(\pi(s)) \quad (3.11)$$

But it is well known that :

$$K_{K1}(s) < K_{K0}(s) \quad (3.12)$$

i.e. : entropy of alphabetic source < entropy of symbolic source. and :

$$K_{K0}(s) - K_{K1}(s) = \mathrm{Ack}^{-1}(l(s)) + O(1) \quad (3.13)$$

(Ack^{-1} : inverse of the Ackermann function) [Gàcs, 1983]

i.e. : the *effective* transcription of a message between symbols and an alphabetic code becomes arbitrarily difficult.

3.3. LENGTH-CONDITIONAL COMPLEXITY OF A GRAPH

Comparing the complexity of a recursive function and the complexity of its graph could give us an example of the difficulty to translate a symbolic source into an alphabetic one [Durand, Porrot, 1998]. Let :

- a recursive function *f*
- its graph $G_f = \{<u,s>,\ s = f(u)\}$

• encoding initial parts G_f^n of G_f by $<f(0), f(1),..., f(n)>_n$

• the complexity C_G of initial parts of the graph of *f*, defined by the length-conditional Kolmogorov complexity :

$$C_G(f) = \limsup_{n\to\infty} C_C\left(G_f^n \middle| n\right) \qquad (3.14)$$

• Let $\boldsymbol{F} = \{f_i\}_{i \in \mathbb{N}}$ any recursive family of distinct recursive functions.

Then :

$$\forall\, c\ \exists\, f_i \in \boldsymbol{F} :\quad C_C(f_i) - C_G(f_i) > c \qquad (3.15)$$

And :

C_C is infinitely often of order of k when C_G is of order of $\log k$. It is true for any recursive function. Therefore one may be of order of *x* while the other one may be of order of *x^x^x^x^x^x^x....*

So :

$$C_C(f_i) > C_G(f_i) \qquad (3.16)$$

example :

For the family *F* defined by

$$\forall\, n \le i :\ f_i(n) = 1 \ \text{ et } \ \forall\, n > i :\ f_i(n) = 0$$

one cannot bound by a constant the difference between the complexities of *f* and its graph.

In other words : the recursive function *f* is the whole information, given *a priori*, but its graph is the information content which we really deal with. So $C_G < C_C$. If we have no other information, we must calculate G_f^n for $i = 0$ to n, in this order, so $C_C(G^n{}_f|n)$ becomes a uniform complexity. As Loveland said, there is a further important comparison to be noted between the length-conditional complexity and the uniform complexity. For any recursive permutation (noted $\bar{s}$) of *s* (string *s* of length $l(s)$), there exists a constant *c* such :

$$|\, C_C(s|l(s)) - C_C(\bar{s}\,|l(s))\,| < c \qquad (3.17)$$

So the length-conditional complexity is invariant under any recursive permutation. It may be regarded, said Loveland, «as bounded by the conditional complexity of its least complex permutation plus the cost of the permutation. This is certainly a natural property for a measure of "information content" to possess when evaluating complete units of patterns». But it is easily observed that the uniform complexity lacks this property.

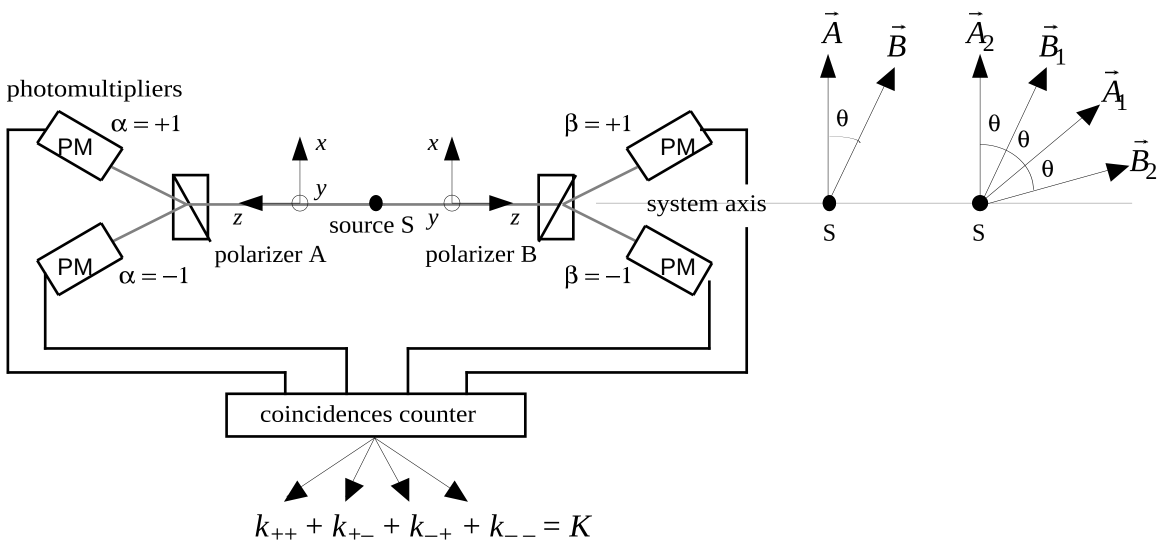
To conclude, it turns out the loss of information which is appearing when we measure information contents from a given *a priori* pattern (i.e. a symbolic source) to its really encoded description (i.e. an alphabetic source), is lying into the *order* of data, which implicitly exists inside the pattern but which is not given to the observer first : this one *must* construct an order to describe the world.

4. INEQUALITIES FROM QUANTUM THEORY

4.1. ENTROPIC BELL INEQUALITIES

It is well known the uncertainty principle implies that the simultaneous perfect knowledge of two conjugate observables is impossible : this is at the root of the violation of Bell inequalities [Bell, 1964] and their generalization CHSH [Clauser *et al.*, 1969]. In recent papers, Cerf and Adami [Cerf, Adami, 1997, 1999] have shown that entropic Bell inequalities can be derived from Venn diagrams, but are violated for a quantum EPR pair [Einstein *et al*, 1935]. This authors have studied the case of three variables.

Here we examine, for instance, the case of four variables, according to experiment of Aspect [Aspect *et al*, 1982] with two widely separated entangled photons in a singlet state $|\Phi^+\rangle$ which leads to such an inequality :



- Fig. 9 9.b 9.c
- Let $\vec{A}$ and $\vec{B}$ analysis axis of detectors : $\theta = \left(\vec{A},\vec{B}\right)$. Each detector has two possible orientations : $\vec{A}_1$, $\vec{A}_2$ (response $\alpha_{1,2} = \pm 1$) and $\vec{B}_1$, $\vec{B}_2$ ($\beta_{1,2} = \pm 1$).
- For each pair (α_i, β_j) and K simultaneous counting, let

$<\alpha_i\beta_j> = (k_{++} - k_{+-} - k_{-+} + k_{--})/K = p_{++} - p_{+-} - p_{-+} + p_{--}$, the mean value of frequencies $k_{\pm\,.}$, which is such that : $-1 \le <\alpha_i\beta_j> \le +1$.

- Let $<\gamma> = <\alpha_1\beta_1> + <\alpha_1\beta_2> + <\alpha_2\beta_1> - <\alpha_2\beta_2>$, which is such that, in the *classical case*, $-2 \le \gamma \le +2$ (CHSH inequality), according to the "*H*" hypothesis : *within the framework of a separable theory, response of the detector* B *in orientation* $\vec{B}_1$ ***must be independent*** *of the orientation* $\vec{A}_1$ *or* $\vec{A}_2$ *of the detector* A.

In the system of reference *Oxyz* of the laboratory, the state of photon *a* crossing A is represented by the vector :

$$\left|\psi_A\right\rangle = \frac{1}{\sqrt{2}}\left(\left|x_A\right\rangle + \left|y_A\right\rangle\right) \tag{4.1}$$

A measurement of polarization of *a* is the projection of this vector onto the basis $\{|X_A\rangle, |Y_A\rangle\}$ associated with the detector A. Let φ_A the angle between the axis (*Ox, Oy*) and (*OX, OY*) :

$$|x_A\rangle = \cos\varphi_A\,|X_A\rangle - \sin\varphi_A\,|Y_A\rangle \tag{4.2}$$

$$|y_A\rangle = \sin\varphi_A\,|X_A\rangle + \cos\varphi_A\,|Y_A\rangle \tag{4.3}$$

The same for *b*. The pair of photons, before any measurement, is an inseparable whole represented by :

$$\left|\varphi\right\rangle = \frac{1}{\sqrt{2}}\left(\left|x_A, x_B\right\rangle + \left|y_A, y_B\right\rangle\right) \tag{4.4}$$

The state vector in the representation φ is :

$$\left|\varphi\right\rangle = \frac{1}{\sqrt{2}}\ [\ \cos(\varphi_B - \varphi_A)\left|X_A, X_B\right\rangle - \sin(\varphi_B - \varphi_A)\left|X_A, Y_B\right\rangle + \sin(\varphi_B - \varphi_A)\left|Y_A, X_B\right\rangle + \cos(\varphi_B - \varphi_A)\left|Y_A, Y_B\right\rangle\ ] \tag{4.5}$$

From that vector, one calculates for instance the probability to detect simultaneous *a* polarized under φ_A and *b* polarized under φ_B angles :

$$p_{++} = \tfrac{1}{2}\cos^2(\varphi_B - \varphi_A) \qquad \text{(responses } |X_A\rangle,|X_B\rangle : \alpha = +1,\ \beta = +1 \Rightarrow \alpha\beta = +1) \tag{4.7}$$

then :

$$p_{+-} = \tfrac{1}{2}\sin^2(\varphi_B - \varphi_A) \qquad \text{(responses } |X_A\rangle,|Y_B\rangle : \alpha = +1,\ \beta = -1 \Rightarrow \alpha\beta = -1) \tag{4.8}$$

and so on. Repeating this calculus for each orientation :

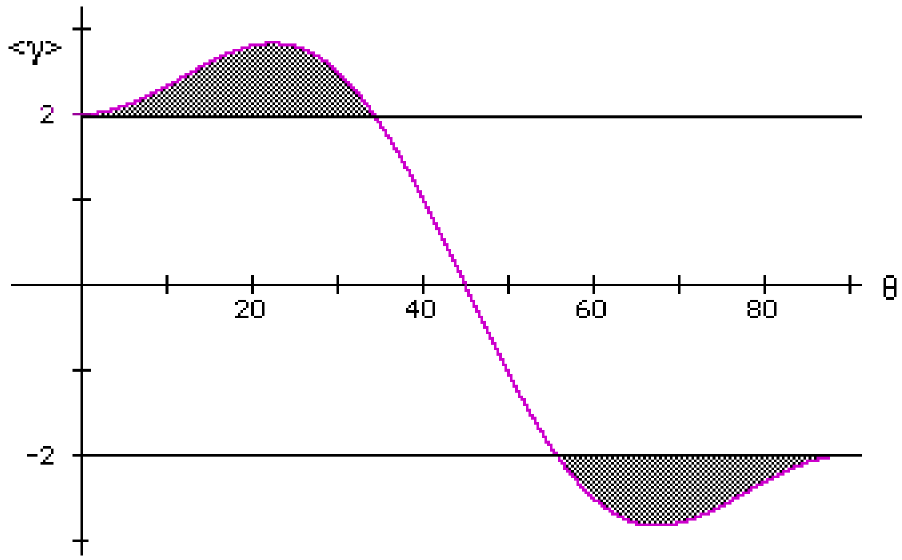
$$<\alpha\beta> = p_{++} - p_{+-} - p_{-+} + p_{--} = \cos[\,2\,(\varphi_B - \varphi_A)\,] \tag{4.9}$$

$$\Rightarrow <\alpha_1\beta_1> = <\alpha_1\beta_2> = <\alpha_2\beta_1> = \cos 2\theta \text{ and } <\alpha_2\beta_2> = \cos 6\theta \tag{4.10}$$

So :

$$<\gamma> = <\alpha_1\beta_1> + <\alpha_1\beta_2> + <\alpha_2\beta_1> - <\alpha_2\beta_2> = 3\cos 2\theta - \cos 6\theta \tag{4.11}$$

The following graph shows the areas where there is violation of Bell inequality (maximum for $\theta = \pi/4 = 22{,}5°$) :



• *Fig. 10* : results of Aspect experiment. Grey areas point out the values of θ where there is entanglement between photons *a* and *b*. *So the "H" hypothesis is false.*

It is possible to interpret the CHSH inequality within the framework of information theory . Let p_{ij} the probability of correlation ($p = p_{++} + p_{--}$) and $1 - p_{ij}$ the probability of anticorrelation ($1 - p = p_{+-} + p_{-+}$) :

$$<\alpha_i\beta_j> = p_{++} - p_{+-} - p_{-+} + p_{--} = p_{ij} - (1-p_{ij}) \Rightarrow p_{ij} = (1 + <\alpha_i\beta_j>) / 2 \quad (4.12)$$

$$\Rightarrow \quad p_{11} = p_{12} = p_{21} = \frac{1+\cos 2\theta}{2} \quad (4.13)$$

$$\text{and} \quad p_{22} = \frac{1+\cos 6\theta}{2} \quad (4.14)$$

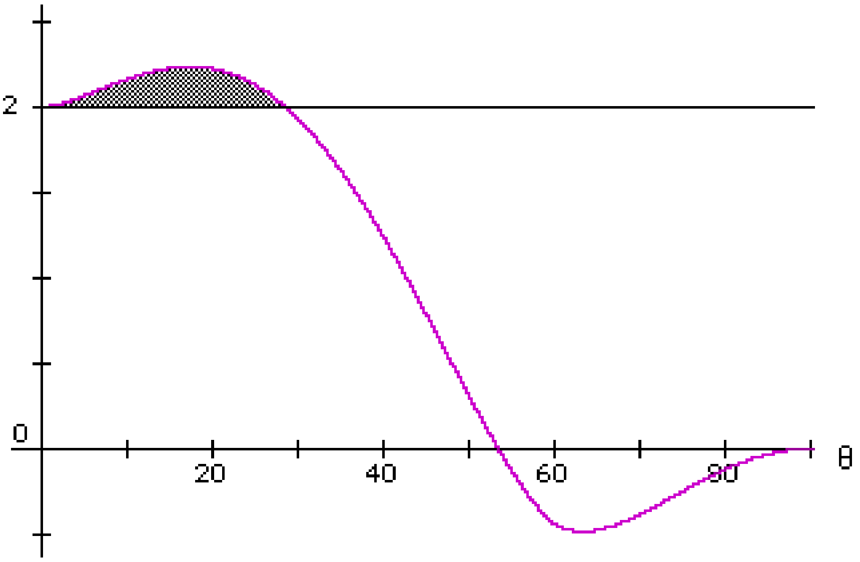
So :

$$H(\vec{A}_i{:}\vec{B}_j) = 1 - H_2(p_{ij}) \quad \text{with} \quad H_2(p_{ij}) = -\,p_{ij}\log p_{ij} - (1-p_{ij})\log(1-p_{ij}) \quad (4.15)$$

The CHSH inequality for mutual entropies can be derived using the inequality (2.13), similar in form to the traditional CHSH inequality, which implies Schumacher's quadrilateral inequality [Schumacher, 1991] :

$$H(\vec{A}_1{:}\vec{B}_1) + H(\vec{A}_2{:}\vec{B}_1) + H(\vec{A}_1{:}\vec{B}_2) - H(\vec{A}_2{:}\vec{B}_2) \le 2 \quad (4.16)$$

The calculus of the left part of this inequality provides the following results :



• *Fig. 11* : entropic equivalent of figure 10.

One sees that such an inequality on mutual entropies shows a domain of violation, that differs from the original one, yet. Here the graph is not symmetric : there is only violation for $\theta \in [0, \approx 30°]$. Maximal violation is ≈ 2,22 for $\theta \approx 17{,}4°$.

What does it means ? Von Neumann entropy *S* (quantum entropy) is defined as :

$$S(\rho) = -\mathrm{Tr}\,(\rho \log \rho) \quad (4.17)$$

with : a library of mixed state $[\Xi\rangle = \{ |\chi_1\rangle, |\chi_2\rangle, \ldots, |\chi_i\rangle, \ldots, |\chi_N\rangle \}$

$p_i = \mathrm{prob}\,(|\chi_i\rangle)$

density matrix $$\rho = \sum_{i=1}^{N} p_i |\chi_i\rangle\langle\chi_i| \quad \text{with} \quad \sum_{i=1}^{N} p_i = 1 \quad (4.18)$$

with the equivalencies (Cerf and Adami give a more general definition extended to non commutative matrix) :

$H(\xi) = -\sum_i p(i) \log p(i)$ $\qquad$ $S(\xi) = -\mathrm{Tr}\,(\rho_\xi \log \rho_\xi)$ (4.19)

$H(\xi,\eta) = -\sum_{i,j} p(i,j) \log p(i,j)$ $\qquad$ $S(\xi,\eta) = -\mathrm{Tr}\,(\rho_{\xi\eta} \log \rho_{\xi\eta})$ (4.20)

$H(\xi|\eta) = -\sum_{i,j} p(i,j) \log p(i|j)$ $\qquad$ $S(\xi|\eta) = -\mathrm{Tr}\,(\rho_{\xi\eta} \log \rho_{\xi|\eta})$ (4.21)

with $p(i|j) = p(i,j) / p(j)$ $\qquad$ with $\rho_{\xi|\eta} = \rho_{\xi\eta} / (\mathbf{1}_\xi \otimes \rho_\eta)$

$H(\xi{:}\eta) = -\sum_i p(i,j) \log p(i{:}j)$ $\qquad$ $S(\xi{:}\eta) = -\mathrm{Tr}\,(\rho_{\xi\eta} \log \rho_{\xi:\eta})$ (4.22)

with $p(i{:}j) = p(i).p(j) / p(i,j)$ $\qquad$ with $\rho_{\xi:\eta} = (\rho_\xi \otimes \rho_\eta) / \rho_{\xi\eta}$

To summarize, a calculus with the matrix $\rho_{\xi|\eta}$ is the same as in the classical case if this one is diagonal. But the eigenvalues of $\rho_{\xi|\eta}$ can be greater than one, so this matrix is not a density matrix. :

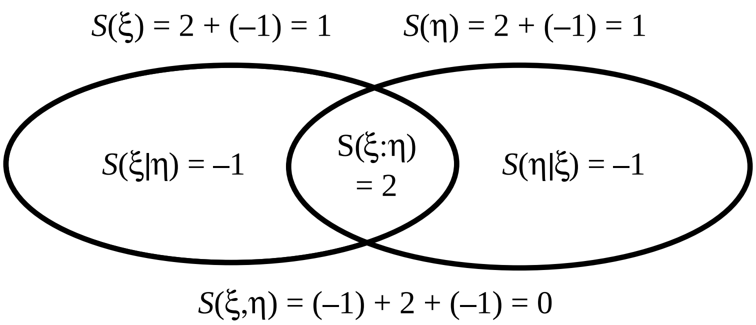
$S(\xi|\eta)$ can be negative. This consequence of non-monotonicity of Von Neumann entropy is an intrinsically quantum property, independent of the fact that classical differential entropy (i.e. classical entropy in continuous case) can be negative. While in the (discrete) classical case we always have :

$$H(\xi:\eta) \leq \min [\, H(\xi), H(\eta)] \tag{4.23}$$

In the quantum case, we have :

$$S(\xi:\eta) \leq 2 \min [\, S(\xi), S(\eta)] \tag{4.24}$$

For example, if $S(\xi) = S(\eta) = 1$ and $S(\xi:\eta) = 2$, then $S(\xi|\eta) = S(\eta|\xi) = -1$ (fig. 12). See also the former result about Aspect experiment.



• *Fig. 12* [From Cerf and Adami]

It ensues that, in the extreme case of a bipartite pure quantum state, we may have $S(\xi) = S(\eta) \neq 0$, while $S(\xi,\eta) = 0$: the information content lies in the nonlocal quantum entanglement, so it is not possible to know anything about the states which was prepared by observing the two subsystems separately (the measurement outcome is random).

More generally, if we measure the observable $M = \sum_{i=1}^{N} |m_i\rangle\, m_i \,\langle m_i|$ with $p(m_i) = \langle m_i|\rho|m_i\rangle$

in the state ρ , then the Shannon entropy of the set of measurement outcomes (source $\boldsymbol{M}$, library $\{m_i\}$) is such that [Preskill, 1998, eq. 5.52]:

$$H(\boldsymbol{m_i}) \geq S(\rho) \tag{4.24}$$

In the same way, distinguishability is lost when we mix nonorthogonal pure states :

$$H(|\chi_i\rangle) \geq S(\rho) \tag{4.25}$$

with equality if the signal states $|\chi_i\rangle$ are mutually orthogonal. *But* nonorthogonal pure quantum states cannot be perfectly distinguished (in this case, the density matrix ρ has also off-diagonal elements). "We can't fully recover information about which state was prepared, because the information gain attained by performing a measurement cannot exceed $S(\rho)$" [Preskill, 1998]. That is to say : the number of available letters, written by physical states, is less than the number of bits which could constitute the symbolic description of the system, i.e. the message. Words are lacking to describe the world...

4.2. HOLEVO BOUND

On an other hand, it can be shown that :

$$S(\xi{:}\eta) \geq H(\xi{:}\eta) \tag{4.26}$$

This is another way to write the Holevo bound, which is the greater *classical* information quantity that is possible to transmit across a quantum binary channel (recall : capacity of a classical BSC : $H = 1 - H_2(p)$). Let two binary sources Σ and Γ, resp. transmitter and receiver, and a single qubit as a signal between Σ and Γ. The library of Σ is two nonorthogonal (equiprobably prepared) states [Holevo, 1973 ; Preskill, 1998] :

$$\xi \in \Xi = \{\chi_1 = |0\rangle, \chi_2 = |1\rangle\}, \quad \text{with :}$$

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} ; \ |1\rangle = \begin{pmatrix} \cos\frac{\theta}{2} \\ \sin\frac{\theta}{2} \end{pmatrix} \quad \text{with} \quad 0 < \theta < \pi \quad et \quad p(0) = p(1) = \frac{1}{2} \tag{4.27}$$

Since both signal states are pure states, Holevo bound reduces to $S(\rho)$, which we may calculate by diagonalization :

$$\rho = \tfrac{1}{2}\,|0\rangle\langle 0| + \tfrac{1}{2}\,|1\rangle\langle 1| = \frac{1}{2}\begin{pmatrix} 1+\cos^2\frac{\theta}{2} & \cos\frac{\theta}{2}\sin\frac{\theta}{2} \\ \cos\frac{\theta}{2}\sin\frac{\theta}{2} & \sin^2\frac{\theta}{2} \end{pmatrix} \tag{4.28}$$

whose eigenvalues are :

$$\det(\rho - \lambda\,\mathbf{1}) = 0 \quad \Leftrightarrow \quad \lambda^2 - \lambda\,\mathrm{tr}\,\rho + \det\rho = 0 \tag{4.29}$$

$$\Leftrightarrow \lambda^2 - \lambda + \frac{1}{4}\left(1+\cos^2\frac{\theta}{2}\right)\sin^2\frac{\theta}{2} - \frac{1}{4}\cos^2\frac{\theta}{2}\sin^2\frac{\theta}{2} = 0 \tag{4.30}$$

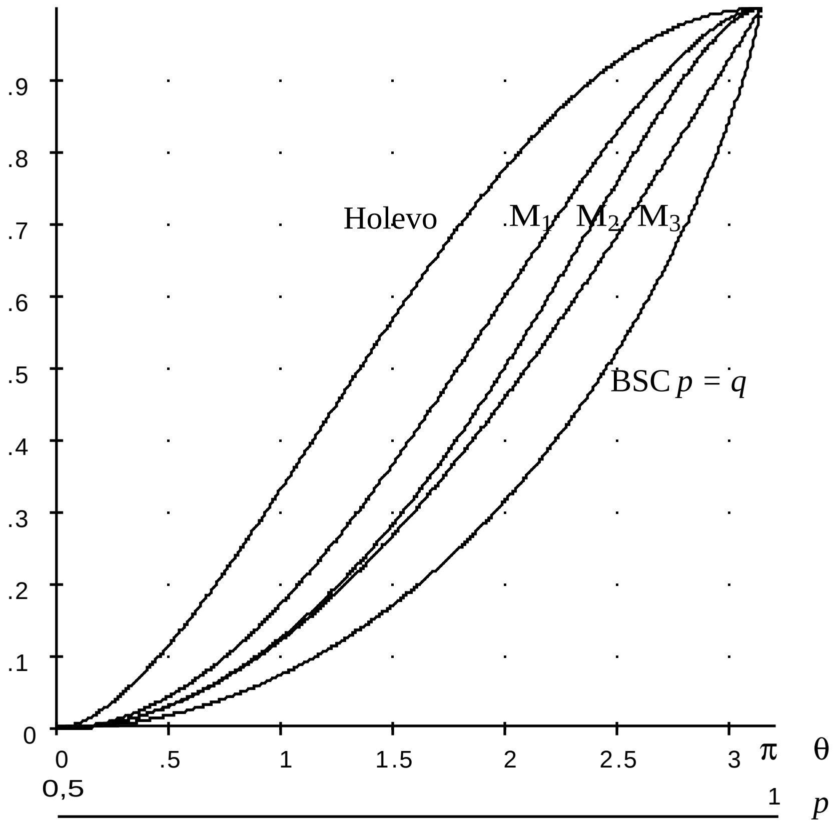
$$\Leftrightarrow \lambda = \frac{1}{2} \pm \frac{1}{2}\sqrt{1-\sin^2\frac{\theta}{2}} = \frac{1}{2} \pm \cos\frac{\theta}{2} = \sin^2\frac{\theta}{4}, \ \ \cos^2\frac{\theta}{4} \tag{4.31}$$

$$\Leftrightarrow \rho = \begin{pmatrix} \sin^2\frac{\theta}{2} & 0 \\ 0 & \cos^2\frac{\theta}{2} \end{pmatrix} \tag{4.32}$$

$$\Rightarrow H(\eta:\xi) = S(\rho) = -\mathrm{tr}\ (\rho \log \rho) = -\sin^2\frac{\theta}{4}\log\sin^2\frac{\theta}{4} - \cos^2\frac{\theta}{4}\log\cos^2\frac{\theta}{4} = H_2\left(\cos^2\frac{\theta}{4}\right)$$

(4.33)

See figure 13.



• *<u>Fig. 13</u>* : comparison between classical case (graph marked BSC), with $0{,}5 \leq p \leq 1$, and quantum case. This chart indicate the *classical* transmitted information quantity (i.e. mutual entropy of Σ and Γ) with a classical bit (BSC) or a qubit (with a $\theta \in [0,\pi]$ between the two states). Between Holevo and BSC, several results are performed, according to the kind of measurement (**M** operator applied to the qubit).

5. CONCLUSION

5.1. *A PRIORI* INFORMATION

Is a quantum symbolic source "weak" (in the sense we have defined it in § 1 : a singular code) ? In term of classical information coding, clearly the response is yes. But this is rather a commonplace point of view : encoding classical information in qubits (with not necessary orthogonal states), it is clear that if we code a 0 for instance in an horizontal polarization photon and a 1 in a diagonal polarization one, no measurement allows to retrieve the whole information bit.

A sharper interpretation is to see the entanglement as the result of a quantum information coding in the two photons, where it exists some redundancy inside this bipartite system : the total information quantity is smaller than one, due to non-discernability of non-orthogonal states.

Now we are in position to refine this last remark in the light of Kolmogorov complexity. We have seen the following inequalities (within a $O(1)$ for K-complexity) :

$$K_C(r \mid s) \geq k_C(r \mid s) = K_C(r \mid u^*) \quad (5.1)$$

$$H(\xi|\eta) \geq S(\xi|\eta) = S(\rho_\xi|\rho_\eta) \quad (5.2)$$

In $I(A|B)$, we called A the information content and B the identification content of A. Now, as the complexity of a string is the length of its minimal description (maximal compressibility), we can see that the value of an information content depends upon its identification content, and is minimal if this last one is known (replacing s by u^*). What about quantum information ? In Shannon calculus, we have no idea about the probabilities which appear in the $H(p)$ formula : we must invoke Bayes' principle, or estimate frequencies from preliminary experiments, to perform this calculus ; we hope the past frequency will be the future likelihood. On the contrary, in quantum theory, we prepared the system, so we are able to calculate the density matrix and the pattern of probabilities, as in complexity theory we wrote the u^* program to work out the s string. The results are not provided by a simple, sometimes arbitrary, estimation, but a set of rulers – the rulers of quantum mechanics. Before knowing an information content, we must know how it is produced : where does it come from ?

But the price of this knowledge is very expensive : k_C is not countable, $K_{K1}(s) < K_{K0}(s)$ with arbitrary difficult encoding, and there is no more discernability of quantum non-orthogonal states.

5.2. *ACCESSIBLE* INFORMATION

In the opposite case, if we have not this implicit knowledge, i.e. if this implicit is not explicited, we have to deal with a continuum with discrete means, within a kind of "DIP" ("Digital Information Processing"). The quantum measurement is discrete ; the initial n-part of the graph of a recursive function is necessary finite. The alphabetic transcription of a symbolic source is a set of finite strings. So computing data from a continuous substrate (infinite string, wave function) needs a sampling which reduces accessible information, with a loss of information on order, i.e. on origin and position of data (or phase), which are defined within a recursive permutation :

$$C_G < C_C \quad (5.3)$$

$$H(\xi{:}\eta) \leq S(\xi{:}\eta) \quad (5.4)$$

So an hypothesis could be the following : underlying any decipherable alphabetic communication process (i.e. distinguishable, discrete, transmitted), there is a non-countable, non-transmitted background. Writing that, we extent to communication processes the well known fundamental scheme of calculability :

The set of subsets of a countably infinite set is not countable :

COUNTABLE ***E*** NOT COUNTABLE $\mathcal{P}(\boldsymbol{E})$

Limitation Theorems (Gödel, Tarski,...)

↙ ↘

Set of Words | Set of Languages

Calculability

↙ ↘

Recursive Procedures, Algorithms Set | Set of Problems

Communication

Sets of Coded Messages

↙ ↘

from a Finite Alphabet	from a Countably Infinite Alphabet
(Alphabetic Sources)	(Symbolic Sources)
Accessible information :	*A priori information :*
Information Content	{*Information + Identification*} *Contents*

6. DIGITAL INFORMATION PROCESSING *(DIP)*

Underlying any decipherable alphabetic communication process (i.e. distinguishable, discrete, transmitted), there is a non-countable, non-transmitted background. This makes a meaningful difference between *"a priori"* and *"accessible"*, i.e. non-transmitted and transmitted information, which is a well known topic among Digital Signal Processing, dealing with continuous vs discrete signals. So modeling this complementarity between these two information states (like energy exists under several states) by means of DSP seems reasonable.

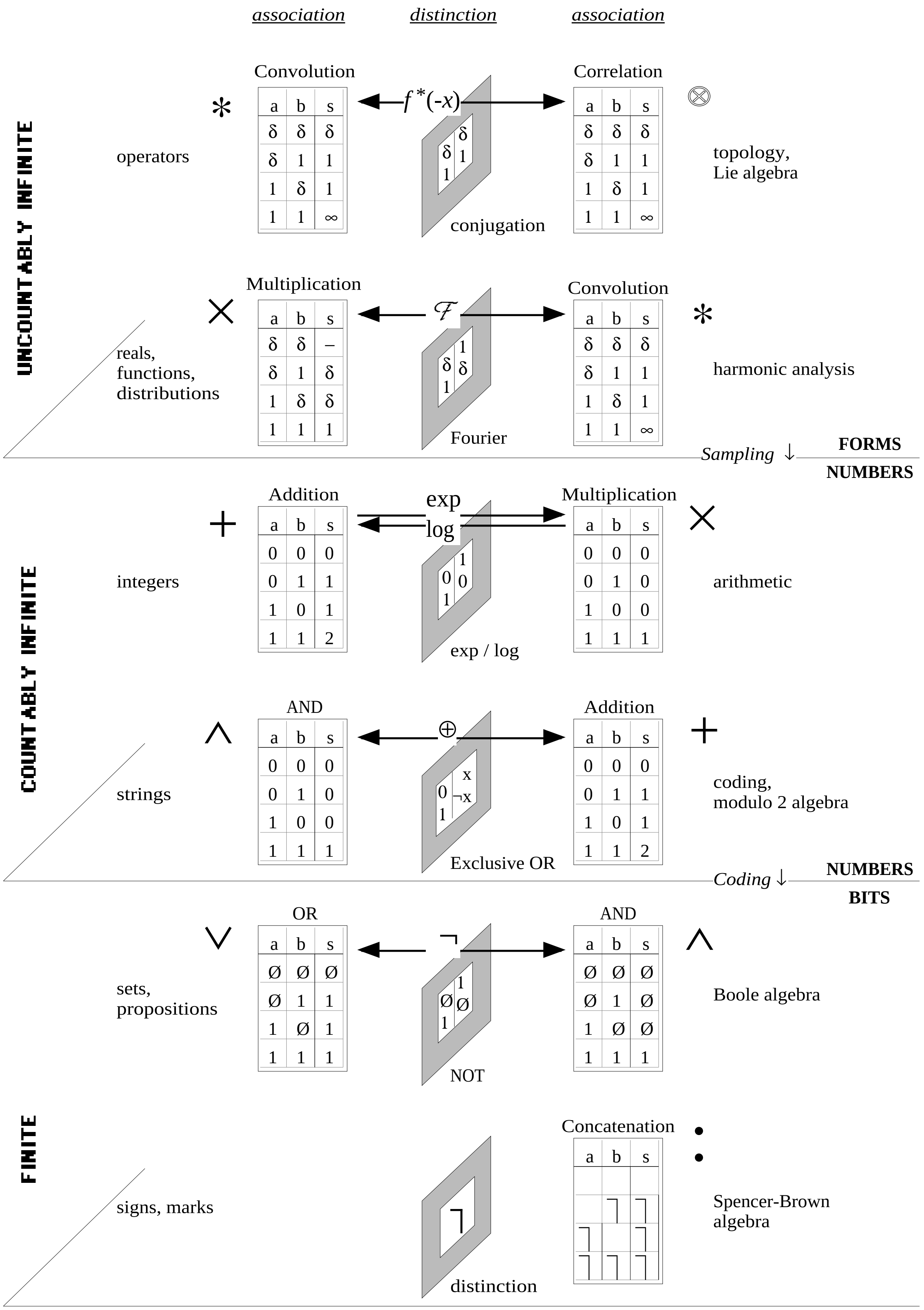

association
distinction
association
UNCOUNTABLY INFINITE
operators
Convolution
a b s
δ δ δ
δ 1 1
1 δ 1
1 1 ∞
f*(-x)
conjugation
Correlation
a b s
δ δ δ
δ 1 1
1 δ 1
1 1 ∞
topology,
Lie algebra
reals,
functions,
distributions
Multiplication
a b s
δ δ –
δ 1 δ
1 δ δ
1 1 1
Fourier
Convolution
a b s
δ δ δ
δ 1 1
1 δ 1
1 1 ∞
harmonic analysis
Sampling ↓
FORMS
NUMBERS
COUNTABLY INFINITE
integers
Addition
a b s
0 0 0
0 1 1
1 0 1
1 1 2
exp
log
exp / log
Multiplication
a b s
0 0 0
0 1 0
1 0 0
1 1 1
arithmetic
strings
AND
a b s
0 0 0
0 1 0
1 0 0
1 1 1
⊕
x
¬x
Exclusive OR
Addition
a b s
0 0 0
0 1 1
1 0 1
1 1 2
coding,
modulo 2 algebra
Coding ↓
NUMBERS
BITS
FINITE
sets,
propositions
OR
a b s
Ø Ø Ø
Ø 1 1
1 Ø 1
1 1 1
¬
NOT
AND
a b s
Ø Ø Ø
Ø 1 Ø
1 Ø Ø
1 1 1
Boole algebra
signs, marks
distinction
Concatenation
a b s
Spencer-Brown
algebra

Fig. 14

6.1. GENERAL DIAGRAM

It is possible to consider the algebraic structures of information processing as making a hierarchic diagram (fig. 14) with three levels : finite (finite sets, boolean logic), countably infinite (integers, arithmetic), non-countable infinite (forms, harmonic analysis). The same operator present at two different hierarchic levels owns a double meaning, as level in which it is. For each algebraic level, a distinction operator (marked Op) links up two associated composition operators which are a matter for a lower algebraic level and an upper respectively :

$$Op(a * b) = Op(a) \bullet Op(b) \qquad (6.1)$$

$$Op^{-1}(a \bullet b) = Op^{-1}(a) * Op^{-1}(b) \qquad (6.2)$$

"Op" is an homomorphism acting on operator marked * and making a new operator marked • at the next level (this process is reversible). Changing level, objects are changing of context, that is to say identification content.

6.2. FORMAL INFORMATION PROCESSING

Formal analogies between boolean calculation and harmonic analysis are well-known : $\{\neg,\vee,\wedge,\varnothing,1\}$ <=> $\{\mathcal{F},*,.,\delta,\mathbf{1}\}$ where $\mathcal{F}$ and * are respectively Fourier Transform and convolution, δ and **1** Dirac and unit functions . This parallel is not exact because of multiplication of distributions impossibility. Therefore, it exists methods of calculation to bypass this difficulty in some particular cases [Colombeau, 1992].

Table 2 : Boole versus Fourier

Distinction	$\neg\neg\, a = a$	$\mathcal{F}[\mathcal{F}[x(t)]] = x(-t)$	(6.3)
Association	$c = a \wedge b$	$z(t) = x(t).\, y(t)$	(6.4)
	$a = a \wedge 1$	$x(t) = x(t)\, .\, \mathbf{1}(t)$	(6.5)
	$c = a \vee b$	$z(t) = x(t) * y(t)$	(6.6)
	$a = a \vee \varnothing$	$x(t) = x(t) * \delta(t)$	(6.7)
Variables	$1 = \neg\, \varnothing$	$\mathbf{1}(t) = \mathcal{F}[\delta(t)]$	(6.8)
	$\varnothing = \neg\, 1$	$\delta(t) = \mathcal{F}[\mathbf{1}(t)]$	(6.9)
Duality	$\neg\, a \vee \neg\, b = \neg\, (a \wedge b)$	$\mathcal{F}[x] * \mathcal{F}[y] = \mathcal{F}[x.y]$	(6.10)
	$\neg\, a \wedge \neg\, b = \neg\, (a \vee b)$	$\mathcal{F}[x]\, .\, \mathcal{F}[y] = \mathcal{F}[x*y]$	(6.11)

Like NOT and exp/log operators, Fourier Transform is a distinction process which can operates between two limits :

F [a.e$^{-bt^2}$] = A.e$^{-B\nu^2}$ (Gauss) ≈ H(0) = 0 (6.12)

F [|_|_|$_1$] = |_|_|$_1$ (Dirac's comb) ≈ H(1) = 0 (6.13)

It means significant information content lies somewhere between desorder of Gauss function (like entropy H of zero-probability), and absolute order of an infinite Dirac's comb (like entropy of probability equal to one).

6.3. SOFTWARE APPLICATION : "SUPERBOOLE" COMPUTER

Specifications

-implementing intelligent calculation functions with signal processing,

-simulating boolean processes by DSP operators,

-extending boolean logic gates properties with DSP features.

Discrete implementation

Physical harmonic analysis of forms follows limitations : quantization, windowing, sampling, holding. This physical limitations are inevitable from the moment that one wants to implement a form, a non-transmitted continuous informational state in real devices. So {$\mathcal{F}$,*,.,δ,**1**} algebra is "digitalized", $\mathcal{F}$ becomes Digital Fourier Transform (noted F), whose calculation is performed by FFT "butterfly" algorithm. In this case "distributions" multiplication becomes approximable.

This kind of limitations is well known in the case of boolean computers : it is impossible to comput infinite quantities, so the only arithmetic wich allows exact calculations is modulo 2 arithmetic. All the more so it's the same for calculations on real quantities or forms : hardware carrying out this operations is also necessarily approximate.

Normalization and NOT operator

FT et IFT (Inverse Fourier Transform) are two symmetrical operations. As F[F[x(t)]] = x(-t), we have, for an even distribution like δ, a relation similar to logic relation ¬¬ a = a.

Unfortunately, this is not true in the digital case, because of 1/N factor that takes place in the IFT :

$$X(j)=\sum_{k=0}^{N-1} x(k)\, e^{-i2\pi\frac{jk}{N}} \;;\; x(k)=\frac{1}{N}\sum_{j=0}^{N-1} X(j)\, e^{+i2\pi\frac{jk}{N}} \qquad (6.14)$$

So, if we want that ¬¬ **1**(t) ≡ **1**(t) et ¬¬ δ(t) ≡ δ(t), we must translate non-transmitted operators by applying an independance scale principle. So the main operator which is Fourier Transform becomes :

$$X_1(j) = F_1[x(k)] = \frac{F[x(k)]}{\sup\left|F[x(k)]\right|} \tag{6.15}$$

With this routine, all resultant amplitudes are kept equal to one. But this principle of calculation does not modify the shape of forms. So normalized FFT F_1 is seen as a NOT operator : a "dirac" that is normalized to one is the inverse of **1**(*t*) function, and conversely. This static scaling must not be confused with dynamic scaling used in FFT computation to avoid arithmetic overflows. Here, the maximum of FFT result magnitude is calculated, and all samples are divided by this quantity, such the max result is equal to one.

This normalization inserts a non-linearity in harmonic analysis, which loses some of its features so. This process is similar to a NOT logic gate, that is in fact a linear sign-changing amplifier whose saturation levels are held to 0 V and 5 V. So this NOT device loses some of its features as linear amplifier but keeps this basic one that is reversing.

Scaling and AND operator

To perform a NOT operator by FFT, and a AND operator by product, we need the following rules (in two's complement) :

NOT : F_1(positive full scale) —> positive full scale

AND : positive full scale X positive full scale —> positive full scale

In fixed-point format, the dynamic range is sufficient for our purpose. A 16-bit number can vary between –32,768 and +32,767. Q15 format is a programmer's convention : the location of the binary point affects neither the arithmetic unit, nor the multiplier in the DSP. It affects only the location from which the result will be read and has no relation to the hardware. So, if any sample keeps an amplitude equal to 0 or 1, the product result (AND) equals always 0 or 1 (as 1 x 1 = 1). For instance, it would be possible to consider the point between bits 14 and 13 (Q14 format), so the most positive number, which is equal to $1–2^{–15}$ in Q15 format, would be equal to $2–2^{–14}$. In this hypothesis, we have to limit positive numbers (i.e. scaling) to the value of one exactly. With such a convention, we lost 3 dB in dynamic range, but we do not need floating-point calculation.

OR operator

As for OR operator, we can : either directly perform convolution (as circular convolution) ; either perform it applying convolution theorem, as a logic OR may be calculated with logic AND and NOT according to De Morgan theorem. To be exact, it would be necessary to extend vectors to 2*N* points, filling with zero segments between *N*+1 and 2*N*, to take windowing and aliasing into account. But if we limit us to vectors as $\delta_1(t)$, **1**(*t*) and normalized combs, we do not care of this here.

"Superbits" : Dirac's Combs

Sampling signals make periodic their spectrum, and conversely. So the above calculations are performed in fact with digital Dirac's combs, whose periods are equal to one for **1**(t) and N for $\delta_1(t)$. As Fourier Transform is an homothetic operator :

$$\mathcal{F}[x(at)] = \frac{1}{|a|} X\left(\frac{\nu}{a}\right) \tag{6.16}$$

we may consider combs whose periods equal a and $1/a$. So a large enough value of N allows us full scope to choose superboolean Ø and 1 combs. It depends on the value of a only, which must equal a power of two between 1 and $N/2$. Clearly, if superboolean Ø and 1 are N-point vectors, this variables will get new algebraic properties, that have not ordinary boolean variables.

"Superbytes" : M-dimension signals

Fourier Transform of a separable M-dimension function is separable. So a M-dimension FFT is calculated as (here M = 2) :

$$X(j_1, j_2) = \sum_{k_2=0}^{N_2-1} \sum_{k_1=0}^{N_1-1} x(k_1,k_2)\, e^{-i2\pi\frac{k_1 j_1}{N_1}}\, e^{-i2\pi\frac{k_2 j_2}{N_2}} \tag{6.17}$$

Let $N_1 = N_2 = \ldots = N$ (square matrix). Let signals : $\delta_1(k_1).\delta_1(k_2)$; $\delta_1(k_1).\mathbf{1}(k_2)$; $\mathbf{1}(k_1).\delta_1(k_2)$; $\mathbf{1}(k_1).\mathbf{1}(k_2)$. This signals are equivalent to bits series : ØØ, Ø1, 1Ø, 11. We have (concatenation marked ":") :

$$\neg\{a : b\} = \neg a : \neg b \quad \Leftrightarrow \quad F_1[\, x(k_1).y(k_2)\,] = F_1[x(k_1)].F_1[y(k_2)] \tag{6.18}$$

This relations extend to any number of "bits", and calculations on M-bit words are possible, such as classic logic does. Bit-by-bit OR, AND, XOR,... operations ensue. Furthermore, a half-adder would be designed like this. Let :

$z = x + y$ with $x, y \in \{\, \delta_1(k_{1,2}), \mathbf{1}(k_{1,2}) \,\}$

$z \in \{\, \delta_1(k_1).\delta_1(k_2)\, ;\ \delta_1(k_1).\mathbf{1}(k_2)\, ;\ \mathbf{1}(k_1).\delta_1(k_2)\, ;\ \mathbf{1}(k_1).\mathbf{1}(k_2) \,\}$

Then :

$$s(k_1) = x(k_1) \oplus y(k_1) \qquad s \in \{\, \delta_1(k_1), \mathbf{1}(k_1) \,\}$$

$$c(k_2) = x(k_1) \wedge y(k_1) \qquad c \in \{\, \delta_1(k_2), \mathbf{1}(k_2) \,\} \tag{6.19}$$

This algorithm allows enumeration of integers, a N-value integer being in fact designed by a volume in M-dimension space, with $M \geq \lceil \log_2 N \rceil$

Timing

If we are just calculating with $\delta_1(t)$ and $\mathbf{1}(t)$ variables as noted above, clearly achieved results and properties are quite identical to classical logic. But we have now to consider translation in time of such signals like combs (whose sampling period is between 1 and *N*). This means to perform calculations with complex values. Algorithm could be executed in two steps : first, a proper calculus on complex amplitudes ; second, a "measure" process which gives the final data as a modulus. This process is equivalent to calculate light intensity from electromagnetic amplitude, or a quantum measure process. It is also evoking neural process, where nerve impulses are pulse groups whose phase is a kind of brain information coding [EGGERMONT, 1990].

6.4. HARDWARE APPLICATION

Accuracy

First, from *N* = 128 to 1024 are usual sizes for discrete signals. Physical considerations such indetermination principle, which is a consequence of Fourier Transform properties :

$$\Delta t . \Delta \upsilon \geq \frac{1}{N} \qquad (6.20)$$

mean enhancement of practical and theoretical result interest if *N* increases : sharp forms require great values of *N* to avoid aliasing. But indetermination principle means infinite accuracy that is impossible on a super-Boole computer, as *N*-bit lenght words of boolean computer are also limited.

Second, with a 2-point FFT (*N*=2), calculus operate on 0-phase and π-phase signals. Phase shift accuracy depends on size *N* of samples set : with an order of magnitude $N \approx 10^2$ (i.e. *N* = 128, 256,...), many cases may be considered.

Memory considerations

-Each super-Boole operator needs a buffer size of 2*N* words (complex data of *N* samples) for just one superbit. On an other hand, an on-chip memory is more usefull, facing the complexity of the whole super-Boole computer. For instance, I.C. Texas 'C20 has 544 words of on-chip data RAM, organized in two 256-word blocks : this memory allows to implement a NOT operator with 256-sample signal and in-place FFT computation. Or it allows a 2-input OR, AND,... with 128-sample signal using 2-block RAM configuration of 'C20.

-Each super-Boole operator needs always the same program, to perform for instance a 2-input NAND. A macro implements a radix-2 DIT N-point FFT, with static scaling to ensure normalization (ranging in 0-1 magnitude) of superbits. With looped code, ROM matrix of TMS320 such 'C25 (4K words on masked ROM) holds easely FFT and normalization implementations.

Calculation time

A radix-2 128-point looped FFT needs 21,879 clock cycles and 4.375 ms execution time on 'C20 (with 5 Mhz clock) [Papamichalis, 1989]. This features must be compared with ordinary boolean circuits, which perform boolean operations in some nanoseconds ! So there is an order of 10^6 between time calculation of Boole and super-Boole computers. This is inherent to working principles of the latter. In return, it could present new properties that can only appear at this level of complexity, as we will see later.

Timing

One of this new features is to be able to comput with lead or lag superbits. So there is necessary to ensure a good synchronization inside the whole device to keep some phase constant throughout a calculation. For example, simulating an holographic operation, where the result is (hologram = NOR super-Boole op.) :

$$F[x*y] \quad \Leftrightarrow \quad \neg\,(a \vee b) \qquad (6.21)$$

needs a sharp control of phase between x and y signals. This synchronicity does not appear on an ordinary boolean device, because a and b boolean values are supposed to be holded on input wires of NOR operator during the whole operation. So a timer (like a program counter) must control the calculation development on each supergate using HOLD inputs of TMS. A general RESET (for 'C50) or SYNC operation (for 'C25) before each set of superboolean calculations must be perform to control time origin.

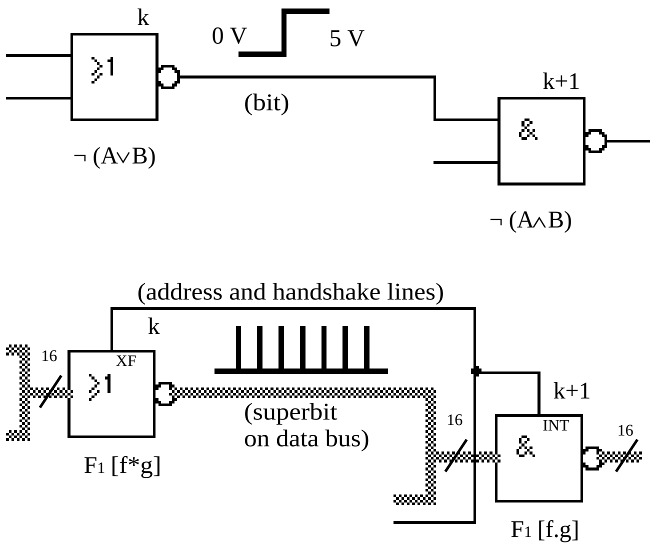
This facilities which are encountered on such silicon devices are the main advantage in relation to quantum computers : it is extremely easy to keep coherence between components, even this ones are separated by a lot of intermediary steps. This is not the case on quantum devices — it is far from being so.

Multiprocessing

Performing an in-place FFT requires to know the whole set of samples of superbits, like an "offline" process (this is not the case for convolution, that is possible to perform "online"). So we have to transmit the samples from the supergate k to the next one k+1 as a "burst" mode as soon as computation by k-gate is ended.

Gates k and k+1 can be interfaced using several solutions. Interrupt (INT) and branch on I/O (BIO) are inputs that are provided from asynchronous sources within a system. The BIO input provides a convenient approach to implementing polled I/O. Two processors can also be synchronized by using BR and READY signals. An other approach for TMS multiprocessing is a multiple master-slave configuration using direct access memory (Fig. 15) [see Texas doc.]. The master (k+1) requests data from the slaves (gates k, "providers"). After each slave's buses control and hold acknowledge signal (from HOLDA slave to BIO master), the slave's XF signal is wired to an INT input of the master : XF is used to indicate to the master when next data are avaible. Such the supergate k+1 may use up to 4 inputs contoled by INT1,2,3,4 with 'C50.

Such a device is not only able to perform boolean calculus from input variables to output results, but also able to perform backward calculation from outputs to inputs : since data buses are bidirectionnal, coefficients (like adaptative filters) can be calculated from results to tune previous operations.



• *Fig. 15* : Master-Slave TMS Multiprocessing

XOR

The basic process making a programmed device with logic gates is XOR operator. The basic function is a YES/NOT programmed operator, wich is performed as $c = a \oplus b$: if a = Ø then $c = b$ else $c = \neg\, b$. The boolean operation XOR a⊕b = (a∧b) ∨ (a∧b) here means a new DSP operation, where signal and spectrum are multiplied :

$$x \oplus y = x.\mathrm{F}_1(y) * \mathrm{F}_1(x).y = x.Y_1 * X_1.y \qquad (6.22)$$

For example, y is a programming input, while x is the variable. If signals are basic superbits (normalized combs), the condition if... then... else... is performed exactly like a boolean calculation. But if the shape of y at logical state "1" is not exactly like a sampled **1**(u), with some samples changing about this value for instance, new conditions add to trigg the execution of the if...then...else condition.

Super-Boole ALU

At last, a whole logic device as 4-bit ALU SN74xx181 can be translated in super-Boole logic. Therefore, the internal diagram of such a device, with about thirty TMS320 interconnected, is to be reconsidered. If we examine the detailed logic diagram of a "simple" SN74xx283 4-bit adder , a basic solution is : each TMS320 takes place of each gate (a maximum of 4 inputs is necessary). But it would be possible to simplify this diagram, because TMS's are underused if they have to perform just a multiplication (AND operators). The internal bus structure of the 283 allows, with multiplexers and further addressing features, connecting some less processors.

RS flip-flop

The flip-flop RS operation means here a recursive calculation :

$$Q_{n+1} = S \vee (Q_n \wedge R) \;\; => \;\; z_{n+1} = x * (z_n . F_1[y]) \qquad (6.23)$$

This flowchart implies a polling algorithm so that RS is able to change state when a new data occurs on inputs R or S. This means a TMS320 would be performing this calculation at any time, a further handshaking being necessary between the flip-flop and the previous devices to take count of new data.

If the shape of f and g at logical states "Ø" or "1" are not exactly like sampled $\delta_1(u)$ and $\mathbf{1}(u)$, with some samples changing about the right shape, new conditions add to trigg the RS :

-first, limit of convolution of some signal (except δ_1, **1**, ...) with itself is a gaussian (central limit theorem). In turn, because of aliasing, limit of sampled gaussian is $\mathbf{1}(u)$:

$$x \vee \left[x \vee \left[x \vee \left[x \vee \left[\cdots\right]\right]\right]\right] \;\to\; e^{-\frac{t_k^2}{2\sigma^2}} \;\to\; 1(k) \qquad (6.24)$$

-second, limit of multiplication of a sample with itself is : either 1 if sample amplitude is equal to 1 (1 X 1 = 1) ; either 0 if sample amplitude is contained between 0 and 1 (0.x X 0.x X ... → 0). So, for some signals, we have :

$$x \wedge \left[x \wedge \left[x \wedge \left[x \wedge \left[\cdots\right]\right]\right]\right] \;\to\; \delta_1(k) \qquad (6.25)$$

This means super-RS is acting like a filter. This is a stability property that, from noise signals, returns normalized combs and basic shapes $\delta_1(u)$ and $\mathbf{1}(u)$.

6.5. NEW FEATURES

Going beyond Boole algebra with DSP is possible because superbits are vectors on which it is possible to perform more powerful basic operations like Fourier Transform and convolution algebra. Moreover, DSP processors could compute several superimposed signals like combs of different frequencies and phases, so it would be possible to simulate the superposed states of a quantum computer operation. In this section, we shall examine briefly some simple features which could enhance the power of electronic calculus.

Symmetry

Discrete Fourier Transform of a scalar (DFT on a single point) is this scalar itself :

$$F[k] = k \tag{6.26}$$

It means superbits are necessary vectors to be distinguished. But vectors imply to define the origine and the direction with wich they are defined. We will see the problem of origine latter and consider now the direction.

First, as we noted, $F[F[x(t)]] = x(-t)$: double negation is not equivalent to a simple NOT operator. We do not care of this fact if function $x(t)$ is even. But in the general case, we have to consider hermitian symmetry, which transforms convolution in correlation :

$$x(t) \longrightarrow x^*(-t) \qquad \rightarrow \qquad x(t) \otimes y(t) = x(t) * y^*(-t) \tag{6.27}$$

$$x(t) \otimes y^*(-t) = x(t) * y(t)$$

Convolution and correlation has two distinct algebraic structures [BORSELLINO, POGGIO, 1973], as convolution is a commutative and associative operation, while correlation is not. To study the two algebras noted $A^{\bullet}$ (avec $\bullet = *$ ou $\otimes$) let the terms :

$$[x,y]^{\bullet} = x \bullet y - y \bullet x \qquad \text{(commutator)} \tag{6.28}$$

$$[f,g,h]^{\bullet} = f \bullet (g \bullet h) - (f \bullet g) \bullet h \qquad \text{(associator)} \tag{6.29}$$

It is easy to show that :

$$[f,g]^{*} = 0 \qquad [f,g,h]^{*} = 0 \tag{6.30}$$

$$[f,g]^{\otimes} \neq 0 \qquad [f,g,h]^{\otimes} \neq 0 \tag{6.31}$$

This relations characterize A^{*} as an abelian algebra (as Boole algebra), while one shows that :

$$[f,f]^{\otimes} = 0 \tag{6.32}$$

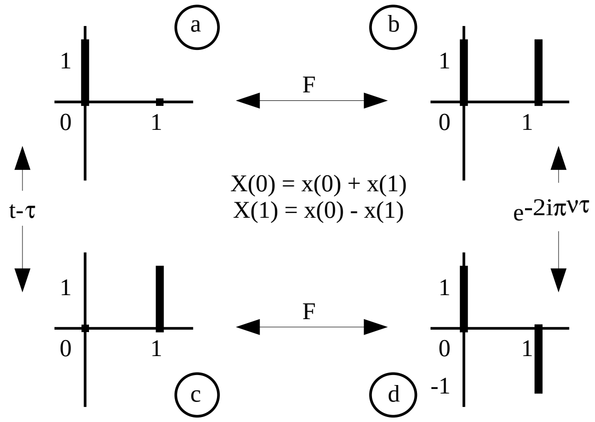
$$[[f,g]^{\otimes},h]^{\otimes} + [[g,h]^{\otimes},f]^{\otimes} + [[h,f]^{\otimes},g]^{\otimes} = 0 \tag{6.33}$$

This relations characterize $A^{\otimes}$ as a Lie algebra, which is a basic formalism of quantum mechanics.

On a limited integration domain (the case of physical signals and filters), *start* of convolution is calculated summing product of *start* of x form by *start* of y form. This calculation can be performed in place between input data and system pulse response, as instantaneous coding. While *start* of correlation is calculated somming product of *start* of x form by *end* of y form or, symmetrical but not commutative calculation, somming product of *start* of y form by *end* of x form. So, it is not any more equivalent calculating $a \vee b$ and $b \vee a$. Information direction becomes significant.

Phase

An other feature of super-Boole calculus concerns the definition of origin from which samples are repaired. Consider for exemple a 2-point FFT whose signals are $a = \delta_1(k)$ and $b = \mathbf{1}(k)$. In this conditions, FFT is :



• *Fig. 16* : *x —> y*

$$X(0) = x(0) + x(1)$$
$$X(1) = x(0) - x(1) \qquad (6.34)$$

Consider *c* which is the translated of *a*. With 2-point signals, this corresponds to a phase difference of π (Fig. 16).

If we calculate basic superboolean operators from a and b, we obtain the following results :

x y	x AND y	x OR y	x —> y
a a	a	a	b
a b	a	b	b
b a	a	b	a
b b	b	b	b
c c	c	a	-d
c b	c	b	0
b c	c	b	c
b b	b	b	b

• *Table 3* : 2-point FFT with lead/lag signals

For AND and OR 2-point supergates, results differ not much from ordinary logic calculations.

As we noted former, we can calculate the amplitude of the result (as a measure process), so AND and OR are strictly the same. But there is not the case for logical implication. Consider the arithmetic zero (marked 0) means a null condition, something which is not significant — a "NAM" (Not-A-Meaning) like it exists "NAN" (Not-A-Number) in high level programming languages. Then, something which is false cannot imply a true consequence, contrary to formal logic. Just a non-sens.

Modulation

After symmetry and phase, a third possibility of changing and improving the basic logic is to modulate or to translate the superbits. Then, the following diagram show that there is a common structure between Fourier calculus and modal logic :

$$F[x(t-\tau)] = e^{-2i\pi\nu\tau} . F[x(t)] \qquad \neg \Box p = \Diamond \neg p \tag{6.35}$$

$$F[e^{-2i\pi at} . x(t)] = X(\nu - a) \qquad \neg \Diamond p = \Box \neg p \tag{6.36}$$

$$x(t-\tau) = F[e^{-2i\pi\nu t} . X(\nu)] \qquad \Box p = \neg \Diamond \neg p \tag{6.37}$$

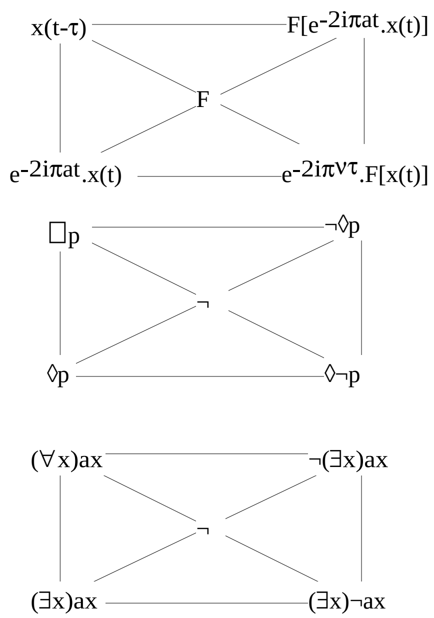
$$e^{-2i\pi at} . x(t) = F[X(\nu - a)] \qquad \Diamond p = \neg \Box \neg p \tag{6.38}$$

For instance, such a modal scheme can be interpreted by $\forall$ and $\exists$ quantificators (Fig. 17). With $x(t) = \delta_1(t)$, $\mathbf{1}(t)$ it is possible to show that :

$$\Box p \rightarrow p \;\Leftrightarrow\; |\, F_1[\, x(t-\tau).F_1[x(t)]\,]\,| = 1 \tag{6.39}$$

$$p \rightarrow \Diamond p \;\Leftrightarrow\; |\, F_1[\, x(t).F_1[e^{-2i\pi at} . x(t)]\,]\,| = 1 \tag{6.40}$$

are true.



• *Fig. 17* : modulation and modal logic

This means it would be possible to go beyond classical logic calculus, introducing a modal logic nearer thought features than boolean calculus. A computer designed with DSP processors performing intelligent calculations from the underlying hardware level, could be powerful to solve some challenges of automatic reasoning.

****

1. ASPECT, A., DALIBARD, J., ROGER, G. : Experimental tests of Bell's inequalities using time-varying analysers, *Physical Review Letters,* vol.49, p 1804-1807, 1982.
2. BELL, J.S. : On the Einstein-Podolsky-Rosen paradox, *Physics 1,* p 195-200, 1964 ; *Rev. Mod. Phys.*, vol.38, p 447, 1996.
3. BORSELLINO A., POGGIO T. : Convolution and Correlation Algebras, *Kybernetik,* vol. 13, 1973, p 113-122.
4. CERF, N. J., ADAMI, C. : Entropic Bell Inequalities, *Phys. Rev. A*, vol.55 p 3371-3374, 1997 `eprint quant-ph/9608047`
5. CERF, N. J., ADAMI, C. : Information theory of quantum entanglement and measurement, *Physica D,* vol.120, p 62-81, 1998. (special issue for PhysComp '96) `eprint quant-ph/9605039.`
6. CERF, N. J., ADAMI, C. : Quantum extension of conditional probability, *Phys. Rev. A*. vol.60, p 893-897, and CERF, N. J., ADAMI, C., GINGRICH, R.M. : Reduction criterion for separability, *Phys. Rev. A*. vol.60, p 898-909, 1999.
7. CHAITIN, G. : On the Length of Programs for Computing Finite Binary Sequences : Statistical Considerations, *Journal of the ACM,* vol.16, n°1, p 145-159, 1969.
8. CHAITIN, G. : *Algorithmic Information Theory*, Cambridge Univ Press, Cambridge 1987, 1990.
9. CLAUSER, J.F., HOLT, R.A., HORNE, M.A., SHIMONY, A. : A proposed experiment to test local hidden-variable theories, *Physical Review Letters,* vol.23, p 880-884, 1969.
10. DUBACQ, J.-C. : *Introduction à la théorie algorithmique de l'information,* Research Report n°98-05, Ecole Normale Supérieure de Lyon, 1998.
11. DURAND, B., PORROT, S. : *Comparison between the complexity of a function and the complexity of its graph,* Research Report n°98-27, Ecole Normale Supérieure de Lyon, 1998. `eprint http://www.ens-lyon.fr/LIP/`
12. EGGERMONT, J. : *The correlative brain,* Springer-Verlag, Berlin,1990.
13. EINSTEIN, A., PODOLSKY, B., ROSEN, N. : Can quantum mechanical description of physics reality be considered complete ? *Physical Review,* vol.47, p 777-780, 1935.
14. GÀCS, P. : On the relation between descriptional complexity and algorithmic probability, *Theoretical Computer Science*, n° 22, p 71-93, 1983.
15. HOLEVO, A. S. : Bounds for the quality of information transmitted by quantum communication channel, *Problems of Information Transmission,* vol.9, p 177-183, 1973.
16. KOLMOGOROV, A.N. : Three Approaches to the Quantitative Definition of Information, *Problems Information Transmission,* vol.1, n°1, p 1-7, 1965.
17. LEVIN, L.A. : Various measures of complexity for finite objects (axiomatic description), *Soviet Math. Dokl.,* n°17, p 522-526, 1976 (Translated from the Russian version).
18. LI M., VITÀNYI P.: *An Introduction to Kolmogorov Complexity and Its Applications,* Springer-Verlag, Berlin, 1993, 1997.
19. LOVELAND, D.W. : On minimal-program complexity measures, *Conference Record of the ACM Symposium on theory of computing,* p 61-65, mai 1969a.
20. LOVELAND, D.W. : A variant of the Kolmogorov concept of complexity, *Information and Control,* vol.15, p 510-526, 1969b.
21. PAPAMICHALIS P. : Implementation of Fast Fourier Transform Algorithms with the TMS32020, *Digital Signal Processing Applications with the TMS320 Family : Theory, Algorithms and Implementations,* vol. 1, Texas Instrument, 1989, p 69-168.

22. PINSON G. : Cognitive Information Theory, *14th Intern. Congress on Cybernetics*, Namur, Belgium, 1995
23. PINSON, G. : Beyond Boole Algebra with TMS320 DSP Multiprocessing, *Proceedings of The First European DSP Education and Research Conference,* Texas Instrument, ESIEE, Paris, 25-26 sept. 1996.
24. PRESKILL, J. : *Advanced Mathematical Methods of Physics,* Physics 229, UCLA, Los Angeles, 1997-98 et 1998-99. `http://www.theory.caltech.edu/~preskill/ph229`.
25. SCHUMACHER, B.W. : Information and quantum nonseparability, *Physical Review A,* vol.44, n°11, p 7047-7052, 1991.
26. SHANNON, C.E. : A Mathematical Theory of Communication, *AT&T Bell Laboratories Technical Journal*, vol.27, p 379-423 et p 623-656, 1948.
27. *TMS320C5x User's Guide*, Texas Instrument, 1993.
28. USPENSKY V.A. : Complexity and Entropy : An Introduction to the Theory of Kolmogorov Complexity, *in* WATANABE O. (dir.), *Kolmogorov Complexity and Computational Complexity,* EATCS, Springer-Verlag, Berlin, 1992.
29. USPENSKY V.A., SHEN, A. : Relations Between Varieties of Kolmogorov Complexities, *Mathematical Systems Theory,* vol.29, n°3, p271-291, 1996.
30. VON NEUMANN, J. : Mathematical Foundations of Quantum mechanics, Princeton University Press, Princeton, NJ, 1955.
31. WEHRL, A. : General properties of entropy, *Review of Modern Physics*, vol.50, n°2, p 221-260, 1978.
32. ZVONKIN, A.K., LEVIN, L.A. : The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms, *Russian Math. Surveys*, vol.25, n°6, p 83-124, 1970.