Law of Excluded Quantum Gambling Strategies

Universitá di Pavia, Italia

March 31, 2022

Abstract

We introduce and analyze a quantum analogue of the Law of Excluded Gambling Strategies of Classical Decision Theory by the definition of different kind of quantum casinos.

The necessity of keeping into account entanglement (by the way we give a straightforward generalization of Schmidt’s entanglement measure) forces us to adopt the general algebraic language of Quantum Probability Theory whose essential points are reviewed.

The Mathematica code of two packages simulating, respectively, classical and quantum gambling is included.

The deep link existing between the censorship of winning quantum gambling strategies and the central notion of Quantum Algorithmic Information Theory, namely quantum algorithmic randomness (by the way we introduce and discard the naive noncommutative generalization of the original Kolmogorov definition), is analyzed.
Contents

1 Von Mises’ Frequentistic Foundation of Probability 3
2 Classical Gambling in the framework of Classical Statistical Decision Theory 9
3 Mises-Wald-Church randomness versus Martin Lőf-Solovay-Chaitin randomness 27
4 Quantum Gambling in the framework of Quantum Statistical Decision Theory 32
5 The censorship of winning quantum gambling strategies 67
1 Von Mises’ Frequentistic Foundation of Probability

The mirable features of the Kolmogorovian measure-theoretic axiomatization of Classical Probability Theory [21] has lead to consider it as the last word about Foundations of Classical Probability Theory, leading to the general attitude of forgetting the other different axiomatizations and, in particular, von Mises’ Frequentistic one [27].

Richard Von Mises’ axiomatization of Classical Probability Theory lies on the mathematical formalization of the following two empirical laws:

1. **Law of Stability of Statistic Relative Frequencies**
   
   It is essential for the theory of probability that experience has shown that in the game of dice, as in all other mass phenomena which we have mentioned, the relative frequencies of certain attributes become more and more stable as the number of observations is increased (cfr. pag.12 of [27]).

2. **Law of Excluded Gambling Strategies**
   
   Everybody who has been to Monte Carlo, or who has read descriptions of a gambling bank, know how many ‘absolutely safe’ gambling systems, sometimes of an enormously complicated character, have been invented and tried out by gamblers; and new systems are still suggested every day. The authors of such systems have all, sooner or later, had the sad experience of finding out that no system is able to improve their chance of winning in the long run, i.e. to affect the relative frequencies with which different colours of numbers appear in a sequence selected from the total sequence of the game. This experience forms the experimental basis of our definition of probability. (cfr. pagg.25-26 of [27]).

According to Von Mises Probability Theory concerns properties of collectivities, i.e. of sequences of identical objects.

Considering each individual object as a letter of an alphabet \(\Sigma\), we can then say that Probability Theory concerns elements of the set \(\Sigma^\infty\) of the sequences of letters from \(\Sigma\) or, more properly, a certain subset \(\text{Collectives} \subset \Sigma^\infty\) whose elements are called **collectives**, where:

**Definition 1.1**

SET OF THE STRINGS ON \(\Sigma\) :

\[
\Sigma^* \equiv \cup_{k \in \mathbb{N}} \Sigma^k
\]  

(1.1)
Definition 1.2

SET OF THE SEQUENCES ON $\Sigma$:

$$\Sigma^{\infty} \equiv \{\lambda\} \bigcup \{\bar{x} : \mathbb{N} \to \Sigma\}$$

where $\lambda$ denotes the empty string.

Given $\bar{x} \in \Sigma^*$ let us denote by $\bar{x}^n \in \Sigma^*$ the string made of n repetitions of $\bar{x}$ and by $a^\infty \in \Sigma^{\infty}$ the sequence made of infinite repetitions of $\bar{x}$.

It is important to remark that [3]:

Theorem 1.1

ON THE CARDINALITIES OF STRINGS AND SEQUENCES OVER A FINITE ALPHABET

HP:

$$\text{cardinality}(\Sigma) \in \mathbb{N}$$

TH:

$$\text{cardinality}(\Sigma^*) = \aleph_0$$
$$\text{cardinality}(\Sigma^{\infty}) = \aleph_1$$

Let us then introduce the set $\text{Attributes}(\Sigma)$ of the attributes of $C$’s elements defined as the set of unary predicates about the generic $C \in \text{Collectives}$.

The mathematical formalization of the Law of Stability of Statistic Relative Frequencies results in the following:

AXIOM 1.1

AXIOM OF CONVERGENCE

HP:

$C \in \text{Collectives}$

$A \in \text{Attributes}(\Sigma)$

TH:

$$\exists \lim_{n \to \infty} \frac{N(A|\bar{C}(n))}{n}$$

where $N(A|\bar{C}(n))$ denotes the number of elements of the prefix $\bar{C}(n)$ of $C$ of length n for which the attribute A holds.

Given an attribute $A \in \text{Attributes}(\Sigma)$ of a collective $C \in \text{Collectives}$ the axiom[1] make consistent the following definition:
Definition 1.3

VON MISES’ FREQUENTISTIC PROBABILITY OF A IN C:

\[ P_{VM}(A|C) := \lim_{n \to \infty} \frac{N(A|\bar{C}(n))}{n} \] (1.3)

Let us then introduce the following basic definition:

Definition 1.4

GAMBLING STRATEGY:

\[ S : \Sigma^{*} \stackrel{\circ}{\to} \{0,1\} \]

where, following the notation of [28], \( f : A \stackrel{\circ}{\to} B \) denotes a partial function from A to B, i.e. a total function \( f : \text{HALTING}(f) \to B \), with \( \text{HALTING}(f) \subseteq A \) called the halting set of \( f \). If \( x \in A - \text{HALTING}(f) \) we will say that \( f \) doesn’t halt on the input \( x \) and will denote it by \( f(x) = \uparrow \).

Given a gambling strategy \( S \):

Definition 1.5

SUBSEQUENCE EXTRACTION FUNCTION INDUCED BY \( S \):

\[ \text{EXT}[S] : \Sigma^{\infty} \to \Sigma^{\infty} : \]

\[ \text{EXT}[S](x_1x_2\cdots) := \text{ordered concatenation}(\{x_n : S(x_1 \cdots x_{n-1}) = 1, n \in \mathbb{N}_+\}) \] (1.4)

The name in the definition 1.5 is justified by the fact that obviously:

\[ \text{EXT}[S](\bar{x}) \leq_s \bar{y} \forall \bar{x} \in \Sigma^{\infty} \] (1.5)

where \( \leq_s \) is the following:

Definition 1.6

SUBSEQUENCE ORDERING RELATION ON \( \Sigma^{\infty} \)

\[ \bar{x} \leq_s \bar{y} := \bar{x} \text{ is a subsequence of } \bar{y} \] (1.6)

Example 1.1

BET EACH TIME ON THE LAST RESULT

Considered the binary alphabet \( \Sigma := \{0,1\} \), let us analyze the following gambling strategy:

\[ S(x_1 \cdots x_n) := \begin{cases} \uparrow & \text{if } n = 0, \\ x_n & \text{otherwise} \end{cases} \quad x_1 \cdots x_n \in \Sigma^n, n \in \mathbb{N} \] (1.7)
and the *subsequence extraction function* $\text{EXT}[S]$ it gives rise to.

Clearly we have that:

| $\vec{x}$ | $S(\vec{x})$ |
|-----------|--------------|
| $\lambda$ | $\uparrow$   |
| 0         | 0            |
| 1         | 1            |
| 00        | 0            |
| 01        | 1            |
| 10        | 0            |
| 11        | 1            |
| 000       | 0            |
| 001       | 1            |
| 010       | 0            |
| 011       | 1            |
| 100       | 0            |
| 101       | 1            |
| 110       | 0            |
| 111       | 1            |
| 0000      | 0            |
| 0001      | 1            |
| 0010      | 0            |
| 0011      | 1            |
| 0100      | 0            |
| 0101      | 1            |
| 0110      | 0            |
| 0111      | 1            |
| 1000      | 0            |
| 1001      | 1            |
| 1010      | 0            |
| 1011      | 1            |
| 1100      | 0            |
| 1101      | 1            |
| 1110      | 0            |
| 1111      | 1            |

Furthermore we have, clearly, that:

- $\text{EXT}[S](0^\infty) = \lambda$
- $\text{EXT}[S](0^\infty) = 1^\infty$
- $\text{EXT}[S](01^\infty \cdots) = 0^\infty$
- $\text{EXT}[S](10^\infty) = 0^\infty$
- $\text{EXT}[S](\bar{x}_{\text{Champernowne}}) = 0101 \cdots$

where $\bar{x}_{\text{Champernowne}}$ is the Champernowne sequence defined as the lexicographic ordered concatenation of the binary strings:

$\bar{x}_{\text{Champernowne}} = 010001101100000101001100101110111 \cdots$
Example 1.2

BET ON THE LESS FREQUENT LETTER

Considered again the binary alphabet $\Sigma := \{0, 1\}$, let us analyze the following gambling strategy:

$$S(\vec{x}) = \begin{cases} 
\uparrow & \text{if } \vec{x} = \lambda \text{ or } N_0(\vec{x}) = N_1(\vec{x}), \\
1 & \text{if } N_0(\vec{x}) > N_1(\vec{x}), \\
0 & \text{otherwise.}
\end{cases}$$

(1.8)

where $N_0(\vec{x}), N_1(\vec{x})$ denote the number of, respectively, zeros and ones in the string $\vec{x}$.

We have that:

| $\vec{x}$ | $S(\vec{x})$ |
| --- | --- |
| $\lambda$ | $\uparrow$ |
| 0 | 1 |
| 1 | 0 |
| 00 | 1 |
| 01 | $\uparrow$ |
| 10 | $\uparrow$ |
| 11 | 0 |
| 000 | 1 |
| 001 | 1 |
| 010 | 1 |
| 011 | 0 |
| 100 | 1 |
| 101 | 0 |
| 110 | 0 |
| 111 | 0 |
| 0000 | 1 |
| 0001 | 0 |
| 0010 | 1 |
| 0011 | $\uparrow$ |
| 0100 | 1 |
| 0101 | $\uparrow$ |
| 0110 | $\uparrow$ |
| 0111 | 1 |
| 1000 | 1 |
| 1001 | $\uparrow$ |
| 1010 | $\uparrow$ |
| 1011 | 0 |
| 1100 | $\uparrow$ |
| 1101 | 0 |
| 1110 | 0 |
| 1111 | 0 |
As to the extraction function of $S$:

\[ EXT[S](0^\infty) = 0^\infty \]
\[ EXT[S](1^\infty) = \lambda \]
\[ EXT[S](01^\infty) = 1^\infty \]
\[ EXT[S](10^\infty) = \lambda \]
\[ EXT[S](\tilde{x}_{\text{Champernowne}}) = 10011011 \ldots \]

Denoted by $Strategies(\text{Collectives})$ the set of gambling strategies concerning $\text{Collectives}$, we can formalize the Law of Excluded Gambling Strategies by the following:

AXIOM 1.2

AXIOM OF RANDOMNESS

HP:

\[ S \in Strategies_{\text{admissible}}(\text{Collectives}) \]
\[ C \in \text{Collectives} \]
\[ A \in Attributes(\Sigma) \]

TH:

\[ P_{VM}(A | EXT[S](C)) = P_{VM}(A | C) \]

where $Strategies_{\text{admissible}}(\text{Collectives}) \subseteq Strategies(\text{Collectives})$ is the set of admissible gambling strategies whose mathematical characterization will lead us, in the next sections, to the heart of Classical Algorithmic Information Theory.
2 Classical Gambling in the framework of Classical Statistical Decision Theory

Classical Statistical Decision Theory concerns the following situation:

- a decision maker have to make a single action \( a \in \text{Actions} \) from a space \( \text{Actions} \) of possible actions.

Features that are unknown about the external world are modelled by an unknown state of nature \( s \in \text{States} \) in a set \( \text{States} \) of possible states of nature.

The consequence \( c(a, s) \in \text{Consequences} \) of his choice depends both on the action chosen and on the unknown state of nature.

Before making his decision the decision maker may observe an outcome \( X = x \) of an experiment, which depends on the unknown state \( s \). Specifically the observation \( X \) is drawn from a distribution \( P_X(\cdot|s) \).

His objectives are encoded in a real valued utility function \( u(a, s) \).

Let us assume that the decision maker knows the action space \( \text{Actions} \), state space \( \text{States} \) and consequence space \( \text{Consequences} \), along with the probability distribution and the utility function.

His problem is:

- observe \( X = x \) and then choose an action \( d(x) \in \text{Actions} \), using the information that \( X = x \), to maximize, in some sense, \( u(d(x), s) \).

Every decision process may obviously be seen as a gambling situation: the action space \( \text{Actions} \) may be seen as the set of possible bets of the decision maker, that we will call from here and beyond the gambler, while the utility function gives the payoff.

Let us consider, in particular, the following gambling situation:

- in the city’s Casino at each turn \( n \in \mathbb{N} \) the croupier tosses a fair coin.

Before the \( n^{th} \) toss the gambler can choose among one of the possible choices:

- to bet one fiche on head
- to bet one fiche on tail
- not to play at that turn

Leaving all the philosophy behind its original foundational purpose we can, now, from inside the standard Kolomogorovian measure-theoretic formalization of Classical Probability Theory, appreciate the very intuitive meaning lying behind Von Mises’ axioms.

Let us indicate by \( X_n \) the random variable on the binary alphabet \( \Sigma := \{0, 1\} \) (where we will assume from here and beyond, that head = 1 and tail = 0) corresponding to the \( n^{th} \) coin toss and by \( x_n \in \Sigma \) the result of the \( n^{th} \) coin toss.

Let us, furthermore, denote by \( \bar{x} := (x_1, x_2, \cdots) \in \Sigma^\infty \) the sequence of all the results of the coin tosses and by \( \bar{x}(n) \in \Sigma^n \) its \( n^{th} \) prefix.

By hypothesis \( \{X_n\}_{n \in \mathbb{N}} \) is a Bernoulli(\( \frac{1}{2} \)) discrete-time stochastic process over \( \Sigma \).

A gambling strategy \( S : \Sigma^* \rightarrow \{0, 1\} \) determines the gambler’s decision at the \( n^{th} \) turn in the following way:
• if $S(\vec{x}(n-1)) = 1$ he bets on head
• if $S(\vec{x}(n-1)) = 0$ he bets on tail
• if $S(\vec{x}(n-1)) = \uparrow$ he doesn’t bet at that turn

The situation may be simulated by the following Mathematica code:
January 11, 2001

\textbf{Classical - Gambling Package}

\texttt{In[1]} := \texttt{Off[General::spell1];}
\texttt{In[2]} := \texttt{<< LinearAlgebra\'MatrixManipulation\';}
\texttt{In[3]} := \texttt{<< Calculus\'DiracDelta\';}
\texttt{In[4]} := \texttt{<< DiscreteMath\'Combinatorica\';}
\texttt{In[5]} := \texttt{<< Statistics\'DataManipulation\';}
\texttt{In[6]} := \texttt{<< Graphics\'Graphics\';}
\texttt{In[7]} := (** "NH" is the non-halting -
    symbol for partial functions ***)
\texttt{In[8]} := \texttt{Clear[NH];}
\texttt{In[9]} := (** "numberofdigits" specifies the number of
    decimal digits taken into account in calculating
    the algorithmic information ***)
\texttt{In[10]} := \texttt{numberofdigits = 1000;}
\texttt{In[11]} := \texttt{\$MaxPrecision = \infty;}
\texttt{In[12]} := \texttt{\$MaxExtraPrecision = numberofdigits;}
\texttt{In[13]} := \texttt{absoluteminimumvalue[f_, x_] := If[Length[x] == 0,
    \texttt{N[Min[\texttt{f[x]/.\texttt{Solve}[\partial_x f[x] == 0 , x]},
    numberofdigits],
    \texttt{N[Min[\texttt{f[x]/.\texttt{Table}[\partial_{x\{i\}} f[x] == 0 ,
    \texttt{\{i, 1, Length[x]\}],
    \texttt{\texttt{Table}[x[x\{i\}], \{i, 1, Length[x]\}]][],
    numberofdigits]]]}}]}}
```mathematica
In[14]:= absoluteminimum[f, x_] :=
     If[Length[x] == 0, N[Part[x/. Solve[Derivative[1][f][x] == 0, x], numberofdigits],
        absoluteminimumvalue[f, x]]],
     numberofdigits],
     N[Part[x/. Solve[Table[Derivative[1][f][x] == 0, {i, 1, Length[x]}],
        Table[x[[i]], {i, 1, Length[x]}]], Flatten[Position[
        N[f[x]/. Solve[Table[Derivative[1][f][x] == 0, {i, 1, Length[x]}],
        Table[x[[i]], {i, 1, Length[x]}]]], numberofdigits],
        absoluteminimumvalue[f, x]]]]},
     numberofdigits]]

In[15]:= (* * * powerQ[x, y] * * *)
In[16]:= powerQ[x_, y_] := IntegerQ[y/. x]

In[17]:= (* * * generalizedselect[l, predicate, y] * * *)
   picks out all elements e_i of the list l
for which the binary predicate[e_i, y] is True * * *)
   (* * N.B.: since the instruction is implemented throwing
away elements of pattern String the list l must
not to contain strings !!! * **)
In[18]:= generalizedselect[l_, predicate_, y_] := Block[{NH = 2},
   DeleteCases[Table[If[predicate[Part[l, i], y],
       Part[l, i], "throwaway"], {i, 1, Length[l]}],
   x_String]]

In[19]:= (* * * counterimages[y, f, n] is the sets of all
the descriptions of the word y via the description
method f that are not longer than n * * *)
In[20]:= auxiliarpredicate1Q[f_, x_, y_] := Equal[f[x], y]

In[21]:= auxiliarpredicate2Q[x_, listoffandy_] := auxiliarpredicate1Q[
   Part[listoffandy, 1], x, Part[listoffandy, 2]]

In[22]:= counterimages[y_, f_, n_] :=
   generalizedselect[wordsupto[n], auxiliarpredicate2Q, {f, y}]

In[23]:= (* * * lengthcounterimages[y, f, n] is the sets of the
length of all the descriptions of the word y via
the description method f that are not longer than n * * *)
```

In[24]:= lengthcounterimages[y_, f_, n_] :=
    Table[Length[Part[counterimages[y, f, n], i]],
        {i, 1, Length[counterimages[y, f, n]]}]

In[25]:= (* * * algorithmicinformationC[y, f, n] is the n — level — approximation of the plain algorithmic information of the word y w.r.t. the descriptive algorithm f * * *)

In[26]:= algorithmicinformationC[y_, f_, n_] :=
    If[Length[lengthcounterimages[y, f, n]] == 0, 
       
       ,  
       
       ]

In[27]:= (* * * lessprolixQ[f, g, n] is the n — level — approximation of the predicate stating that the descriptive method f is less prolix than the descriptive method g * * *)

In[28]:= lessprolixQ[f_, g_, n_] := Less[
    Max[Flatten[Table[algorithmicinformationC[word[k, m], g, n] —
        algorithmicinformationC[word[k, m], f, n],
        {m, 1, n}, {k, 1, Length[words[m]]}], 1]],
    
    ]

In[29]:= (* * * equallyprolixQ[f, g, n] is the n — level — approximation of the predicate stating that the descriptive method f has the same prolixity of the descriptive method g * * *)

In[30]:= equallyprolixQ[f_, g_, n_] :=
    And[lessprolixQ[f, g, n], lessprolixQ[g, f, n]]

In[31]:= (* * * optimalityQ[f, class, n] is the n — level — approximation of the predicate stating that the descriptive method f is optimal w.r.t. the class of descriptive methods class * * *)

In[32]:= optimalityQ[f_, class_, n_] :=
    Equal[Table[
        lessprolixQ[f, Part[class, i], n], {i, 1, Length[class]}],
        Table[True, {Length[class]}]]

In[33]:= (* * * objectivityQ[class, n] is the n — level — approximation of the predicate stating that Kolmogorov complexity by the class of description methods class is objective * * *)

In[34]:= objectivityQ[class_, n_] :=
    Unequal[Table[optimalityQ[
        Part[class, i], class, n], {i, 1, Length[class]}],
        Table[False, {Length[class]}]]
(* * * algorithmicincomprimibilityQ[y, f, n, \(\epsilon\)] is the n-level approximation of the predicate stating that the word y passes the test of randomness associated to the description method f with associated tolerance \(\epsilon\) * *)

algorithmicincomprimibilityQ[y_, f_, n_, \(\epsilon\)] := GreaterEqual[algorithmicinformationC[y, f, n], Length[x] - \(\epsilon\)]

(* * * belongQ[x, y] is the predicate stating that x belongs to the list y * *)

belongQ[x_, y_] := MemberQ[y, x]

(* * * incluseequalQ[x, y] is the predicate stating that the list x is incluse or equal to the list y * *)

incluseequalQ[x_, y_] := Equal[Intersection[x, y], x]

(* * * incluseQ[x, y] is the predicate stating that the list x is strictly incluse in the list y * *)

incluseQ[x_, y_] := And[incluseequalQ[x, y], Length[x] < Length[y]]

(* * * martinloftestQ[v, n, m] is the predicate stating that the sequence of families of words \(\{v[m]\}_{m \in N}\) is a Martin–Lof test w.r.t. the couple of integer numbers \(n \geq m \geq 1\) * *)

martinloftestQ[v_, n_, m_] := And[incluseQ[v[m + 1], v[m]], 
Length[generalizedselect[words[n], belongQ, v[m]]] < 
2^(-n - m)]

(* * * martinlof[v, n] is the n-level approximation of the predicate stating that the sequence of families of words \(\{v[m]\}_{m \in N}\) is a Martin–Lof test * *)

martinlof[v_, n_] := Equal[Table[martinloftestQ[v, n, m], \{m, 1, n\}], Table[True, \{m, 1, n\}]]
In[47]:= (* * * martinlofrandomnessQ[x, v, n] is the predicate
testing that the word x is random w.r.t. the Martin–
Lof test v at level m ***)

In[48]:= martinlofrandomnessQ[x_, v_, m_] :=
And[belongQ[x, v[m]], Length[x] > m ]

In[49]:= (* * * criticallevel[x, v, n] is the n – level approximation
of the critical level associated to the Martin–
Lof test v ***)

In[50]:= auxiliarpredicate3Q[m_, v_, x_] := belongQ[x, v[m]]

In[51]:= auxiliarpredicate4Q[m_, y_] :=
auxiliarpredicate3Q[m, Part[y, 1], Part[y, 2]]

In[52]:= criticallevel[x_, v_, n_] :=
If[belongQ[x, v[1]] , 
Max[generalizedselect[
Table[m, {i, 1, n}], auxiliarpredicate2Q, {v, x}]],
0]

In[53]:= (* * * numberof[i, l] gives the numbers of "i"’s
occurring in the lists "l" ***)

In[54]:= beequaltoQ[x_, y_] := Equal[x, y]

In[55]:= numberof[i_, l_] := Length[generalizedselect[l, beequaltoQ, i]]

In[56]:= (* * * "words[n]" is the list of all the words of length
n in lexicographic ordering ***)

In[57]:= words[n_] := Strings[{0, 1}, n]

In[58]:= (* * * wordsupto[n] is the list of all the words of
length less or equal to n in lexicographic ordering ***)

In[59]:= wordsupto[n_] :=
If[n == 1, words[1], Join[wordsupto[n - 1], words[n]]]

In[60]:= (* * * subwords[x] gives the list of all the sub–
words of the word x in lexicographic ordering ***)

In[61]:= subwords[x_] := LexicographicSubsets[x]

In[62]:= (* * * word[n] gives the
n’th word in global lexicographic ordering ***)

(* * * N.B. : up to now it is defined only for n < 16 ***)

In[63]:= word[n_] := Part[wordsupto[15], n]
In[64]:= (** * "locallexicographicnumber[x]" gives 
the local lexicographic number of the word x,
i.e. the lexicographic number of x w.r.t. to the set
of the words having the same length of x ***)

In[65]:= locallexicographicnumber[x_] :=
Part[Flatten[Position[words[Length[x]], x]], 1]

In[66]:= (** * "globallexicographicnumber[x]" gives the global 
lexicographic number of the word x, i.e. the lexicographic
number w.r.t. to the set of the words of any length***)

In[67]:= globallexicographicnumber[x_] :=
If[Length[x] == 1, locallexicographicnumber[x],
Length[wordsupto[Length[x] - 1]] +
locallexicographicnumber[x]]

In[68]:= (** * "quasilexicographicnumber[x]" 
gives the quasilexicographic number of the word x,
i.e. the lexicographic number w.r.t. to the set of the
words of any length considering also the null string***)

In[69]:= quasilexicographicnumber[x_] :=
If[x == {0}, 0, globallexicographicnumber[x] + 1]

In[70]:= (** * complementation, conjunction and disjunction are
the common boolean operators implemented on the set
of finite words of our binary alphabet {0, 1}***)

In[71]:= complementation[x_] :=
Table[If[Part[x, i] == 0, 1, 0], {i, 1, Length[x]}]

In[72]:= disjunction[x_, y_] :=
Table[If[Or[Part[x, i] == 1, Part[y, i] == 1], 1, 0],
{i, 1, min[Length[x], Length[y]]}]

In[73]:= conjunction[x_, y_] :=
Table[If[And[Part[x, i] == 1, Part[y, i] == 1], 1, 0],
{i, 1, min[Length[x], Length[y]]}]

In[74]:= (** * "clone[x, n]" is the word made of "n" repetitions
of the word "x" ***)

In[75]:= clone[x_, n_] := If[n == 1, Join[x, x], Join[clone[x, n - 1], x]]
In[76]:= (** * "palindromeQ[x]" is the predicate stating that the word "x" is palindrome ***)

In[77]:= palindromeQ[x_] := Equal[x, Reverse[x]]

In[78]:= (** * "universalbinarybooleanfunction[x, y]" gives the value on the string y of the boolean function having truth—table specified by x ***)

In[79]:= universalbinarybooleanfunction[x_, y_] :=
   If[IntegerQ[2^Length[x]],
     If[Length[y] == 2^Length[x],
       Part[x, locallexicographicnumber[y]], NH],
     NH]

In[80]:= (** * "universalunarybooleanfunction[x]" gives the value on the string formed by the first half of x of the boolean function having truth—table specified the bystring formed by the second half of x ***)

In[81]:= haltinglengths = Table[n + 2/n, {n, 1, 100}];

In[82]:= universalunarybooleanfunction[x_] :=
   If[MemberQ[haltinglengths, Length[x]],
     universalbinarybooleanfunction[Take[x, Length[x] -
       Part[Flatten[Position[haltinglengths, Length[x]]], 1]],
     Take[x,
       Part[Flatten[Position[haltinglengths, Length[x]]], 1]]],
     NH]

In[83]:= (** * "godelization[index, x]", where "index" is a list to the form "inputvariables, outputvariables, thruthtable" gives the value assumed on the string "x" by the algorithm with thruthtable "thruthtable", having a number of boolean input variables equal to the global lexicographic number of the word "inputvariables" and having a number of boolean output variables equal to the global lexicographic number of the word "outputvariables" ***)
\textbf{In[84]} := \textbf{godelization}[\text{index, } x_] := \\
\text{If}[\text{Length}[\text{Part}[\text{index, 3}]] == \\
\text{globallexicographicnumber}[\text{Part}[\text{index, 2}]]* \\
2\text{globallexicographicnumber}[\text{Part}[\text{index, 1}]], \\
\text{If}[\text{Length}[x] == \text{globallexicographicnumber}[\text{Part}[\text{index, 1}]], \\
\text{Take}[\text{Part}[\text{index, 3}], \\
\{\text{locallexicographicnumber}[x], \text{locallexicographicnumber}[x]+ \\
\text{globallexicographicnumber}[\text{Part}[\text{index, 2}]] - 1\}], \text{NH}], \\
\text{NH}]

\textbf{In[85]} := (* * * \textbf{godelnumber}[\text{index}] * * *) \\
\text{godelnumber}[\text{index}] := \\
\text{Length}[\text{index}] \\
\prod_{i=1}^\text{Length}[\text{index}] \text{Prime}[i]^{\text{globallexicographicnumber}[\text{Part}[\text{index, 1}]]}

\textbf{In[86]} := (* * * \textbf{godelword}[\text{index}] * * *) \\
\text{godelword}[\text{index}] := \text{word}[\text{godelnumber}[\text{index}]]

\textbf{In[87]} := (* * * \textbf{haltQ}[\text{index, x}] * * *) \\
\text{haltQ}[\text{index, x}] := \\
\text{Block}[\{\text{NH} = 2\}], \text{Unequal}[\text{godelization}[\text{index, x}], \text{NH}]]

\textbf{In[88]} := (* * * \textbf{diagonal}[\text{index}] * * *) \\
\text{diagonal}[\text{index}] := \\
\text{If}[\text{haltQ}[\text{index, godelword}[\text{index}]], \text{NH}, \text{index}]

\textbf{In[89]} := (* * * \textbf{prefix}[\text{x, n}] * * *) \\
\text{prefix}[\text{x, n}] := \\
\text{Take}[\text{x}, \text{n} - 1]
In[94]:= prefix[x_, n_] := Part[x, Table[i, {i, 1, n}]]

In[95]:= (** * prefixes[x] is the list of the prefixes of the word x * **)

In[96]:= prefixes[x_] := Table[prefix[x, n], {n, 1, Length[x] - 1}]

In[97]:= (** * "admittedsubwords[x, selectionrule]" gives the list of all the subwords of x admitted by the admissible—place—selection rule "selectionrule" in the sense of Von Mises’ theory of collectives * **)

In[98]:= admittedsubwords[x_, selectionrule_] := Select[subwords[x], selectionrule]

In[99]:= (** * "admittedsubwords[x, selectionrule, y]" gives the list of all the subwords of x admitted by the admissible—place—selection rule "selectionrule[y]" depending on the additional variable y in the sense of Von Mises’ theory of collectives * **)

In[100]:= admittedsubwords[x_, selectionrule_, y_] := generalizedselect[subwords[x], selectionrule, y]

In[101]:= (** * "subwordfrequencies[x, selectionrule]" gives the list of the frequencies of the subwords of x admitted by the admissible—place—selection rule "selectionrule" in the sense of Von Mises’ theory of collectives * **)

In[102]:= subwordfrequencies[x_, selectionrule_] := Table[Frequencies[admittedsubwords[x, selectionrule]], {i, 1, Length[admittedsubwords[x, selectionrule]]}]

In[103]:= (** * "subwordfrequencies[x, selectionrule, y]" gives the list of the frequencies of the subwords of x admitted by the admissible—place—selection rule "selectionrule[y]" depending on the additional variable y in the sense of Von Mises’ theory of collectives * **)

In[104]:= subwordfrequencies[x_, selectionrule_, y_] := Table[Frequencies[admittedsubwords[x, selectionrule, y]], {i, 1, Length[admittedsubwords[x, selectionrule, y]]}]
subwordtest[x, selectionrule] gives the histogram of the frequencies of the subwords of x admitted by the admissible—place—selection rule "selectionrule" in the sense of Von Mises' theory of collectives.

subwordtest[x, selectionrule, y] gives the histogram of the frequencies of the subwords of x admitted by the admissible—place—selection rule "selectionrule[y]" depending on the additional variable y in the sense of Von Mises' theory of collectives.

divisibilityQ[x, y, n] is the admissible—place—selection rule stating that the word "x" may be obtained dividing the word "y" in n segments of equal length.

Consequently, such a predicate corresponds to the test of Borel normality.

divisibilityQ[x, y, n] := auxiliarypredicate5Q[x, y, 0, n] := If[Length[x] == Length[y]/n, MemberQ[Partition[y, n], x], False]

divisibilityQ[x, y, n] := auxiliarypredicate5Q[x, y[1], y[2]]

INITIAL ASSIGNMENTS

n = 10000;

algorithm[past] is the algorithm computing how to bet at the next coin toss in function of the list "past" of the previous results.
In[116]:= algorithm [past_] := If[past == {}, NH, Last[past]]

In[117]:= (** END OF INITIAL ASSIGNATIONS ***)

In[118]:= (** "game" is the list of the coin tosses ***)

In[119]:= game = Table[Random[Integer], {n}];

In[120]:= (** "relativefrequency[x,i]" gives the relative frequency of "x"'s occurring in the first "i" coin tosses ***)

In[121]:= relativefrequency[x_, i_] := numberof[x, prefix[game, i]] / i

In[122]:= (** "discard[i]" gives the difference between the number of 0's and the number of 1's occurred in the first "i" coin tosses ***)

In[123]:= discard[i_] := numberof[0, prefix[game, i]] - numberof[1, prefix[game, i]]

In[124]:= (** "recurrenceQ[i]" is the predicate stating that in the first "i" coin tosses the relative frequency of 0's and of 1's are equal ***)

In[125]:= recurrenceQ[i_] := Equal[relativefrequency[1, i], 1/2]

In[126]:= (** "strategy[i]" fixes how to bet at the i"th coin toss ***)

In[127]:= strategy[ i_] := algorithm [Take[game, i - 1]]

In[128]:= (** "payoff" gives the final payoff obtained ***)

In[129]:= payoff = Sum[ If[ReplaceAll[strategy[ i] == NH, NH -> 2], 0, If[strategy[ i] == Part[game, i], 1, -1]], {i, n}];
where the initial assignations may be arbitrarily variated from their default: a number of 10000 coin tosses and the adoption of the gambling strategy discussed in the following example

**Example 2.1**

**APPLYING TO THE CASINO THE GAMBLING STRATEGY OF EXAMPLE 1.1**

Let us suppose that the first 10 coin tosses give the following string of results:

\[ \vec{x}(n) = 1101001001 \]

Our evening to Casino may be told by the following table:

| TOSS | RESULT OF THE TOSS | BET MADE ABOUT THAT TOSS | PAYOFF |
|------|--------------------|--------------------------|--------|
| 1    | 1                  | no bet                   | 0      |
| 2    | 1                  | 1                        | +1     |
| 3    | 0                  | 1                        | 0      |
| 4    | 1                  | 0                        | -1     |
| 5    | 0                  | 1                        | -2     |
| 6    | 0                  | 0                        | -1     |
| 7    | 1                  | 0                        | -2     |
| 8    | 0                  | 1                        | -3     |
| 9    | 0                  | 0                        | -2     |
| 10   | 1                  | 0                        | -3     |

As we see \( PAYOFF(10) = -3 \).

**Example 2.2**

**APPLYING TO THE CASINO THE GAMBLING STRATEGY OF EXAMPLE 1.2**

Also this gambling situation may of course be simulated with the previously introduced Mathematica code by, simply, changing the initial assignation of the function \( \text{algorithm[past]} \) in the following way:

\[
\text{algorithm[past]} := \begin{cases} 
\text{If[\text{numberof}[1, \text{past}] == \text{numberof}[0, \text{past}], NH,} \\
\text{If[\text{numberof}[1, \text{past}] > \text{numberof}[0, \text{past}], 0, 1]} \end{cases} \quad (2.1)
\]

Let us suppose again that the first 10 coin tosses give the following string of results: \( \vec{x}(n) = 1101001001 \).

Our evening to Casino may be told by the following table:
As we see $PAYOFF(10) = +3$.

The probability distribution of the string $\vec{x}_n$ is the uniform distribution on $\Sigma^n$:

$$\text{Prob}[\vec{x}(n) = \vec{y}] = \frac{1}{2^n} \quad \forall \vec{y} \in \Sigma^n, \forall n \in \mathbb{N}$$  \hspace{1cm} (2.2)

When $n \to \infty$ such a distribution tends to the following:

**Definition 2.1**

**UNBIASED PROBABILITY MEASURE ON $\Sigma^\infty$:**

$P_{\text{unbiased}} : 2^\Sigma^\infty \xrightarrow{\text{d}} [0,1]$:

$$\text{HALTING}(P_{\text{unbiased}}) = \mathcal{F}_{cylinder}$$ \hspace{1cm} (2.3)

$$P_{\text{unbiased}}(\Gamma_{\vec{x}}) = \frac{1}{2^{||\vec{x}||}} \quad \forall \vec{x} \in \Sigma^*$$ \hspace{1cm} (2.4)

with $||\vec{x}||$ denoting the length of the string $\vec{x}$ and where:

**Definition 2.2**

**CYLINDER SET W.R.T. $\vec{x} = (x_1,...,x_n) \in \Sigma^*$:**

$$\Gamma_{\vec{x}} \equiv \{\vec{y} = (y_1,y_2,...) \in \Sigma^\infty : y_1 = x_1,...,y_n = x_n\}$$ \hspace{1cm} (2.5)

**Definition 2.3**

**CYLINDER - $\sigma$ - ALGEBRA ON $\Sigma^\infty$:**

$$\mathcal{F}_{cylinder} \equiv \sigma - \text{algebra generated by}\{\Gamma_{\vec{x}} : \vec{x} \in \Sigma^*\}$$ \hspace{1cm} (2.6)

Clearly the possible attributes of a letter on the binary alphabet are:

- $a_1 := << \text{to be 1} >>$
\[ a_0 := \langle \text{to be } 0 \rangle \]

so that:

\[ \text{Attributes}(\Sigma) = \{ a_1, a_0 \} \quad (2.7) \]

Whichever \textit{Collectives} \( \subseteq \Sigma^\infty \) is the axiom\(^1\) is, from inside the standard kolmogorovian measure-theoretic foundation, an immediate corollary of the Law of Large Numbers.

As far as axiom\(^1.2\) is concerned, anyway, the situation is extraordinarily subtler.

Every \textit{intrinsic regularity} of \( \vec{x}(n) \) could have been encoded by the gambler in a proper \textit{winning strategy up to the} \( n^\text{th} \) \textit{turn}.

The same definition of what a \textit{winning strategy} is requires some caution: we can, indeed, give two possible definitions of such a concept:

\textbf{Definition 2.4 (AVERAGE-WINNING STRATEGY UP TO THE} \( n^\text{th} \) \textit{TOSS})

\( a \) \textit{strategy so that the expectation value of the payoff after the first} \( n \) \textit{tosses} \( \text{payoff}(n) \) \textit{is greater than zero}.

The fact the a strategy is average-winning doesn’t imply that the payoff after the \( n^\text{th} \) toss will be strictly positive with certainty: it happens if we are lucky.

Let us now introduce a weaker notion of a winning strategy:

\textbf{Definition 2.5 (LUCKY-WINNING STRATEGY UP TO THE} \( n^\text{th} \) \textit{TOSS})

\( a \) \textit{strategy so that the the probability that the payoff after the first} \( n \) \textit{tosses} \( \text{payoff}(n) \) \textit{is greater than zero is itself greater than zero}.

For finite \( n \) every strategy is obviously lucky-winning.

Let us now consider the limit \( n \to \infty \).

By purely measure-theoretic considerations we may easily prove the following:

\textbf{Theorem 2.1 WEAK LAW OF EXCLUDED GAMBLING STRATEGIES}

\( \text{For } n \to \infty \text{ the set of the average-winning strategies tends to the null set} \)

\textbf{PROOF:}

Given a gambling strategy \( S : \Sigma^* \to \{ 0, 1 \} \) we have clearly that:

\[ E[\text{payoff}(n)|\text{payoff}(n - 1)] = \text{payoff}(n - 1) + \]

\[ If[S(\vec{x}_{n-1}) = \uparrow, 0, \frac{1}{2}] \]

\[ If[S(\vec{x}_{n-1}) = 1, 1, -1] + \frac{1}{2} \]

\[ If[S(\vec{x}_{n-1}) = 0, 1, -1] = \text{payoff}(n - 1) \quad \forall n \in \mathbb{N} \quad (2.8) \]

(\text{where I have adopted Mc Carthy’s LISP conditional notation \( \uparrow \) popularized by Wolfram’s \textit{Mathematica} [50]).

24
Furthermore:

\[ E[\text{payoff}(n)] = \sum_{k=-n+1}^{n-1} P[\text{payoff}(n-1) = k] E[\text{payoff}(n)|\text{payoff}(n-1)] \quad \forall n \in \mathbb{N} \] \hspace{1cm} (2.9)

We will prove that \( \lim_{n \to \infty} E[\text{payoff}(n)] = 0 \) by proving by induction on \( n \) that \( E[\text{payoff}(n)] = 0 \) \( \forall n \in \mathbb{N} \).

That \( E[\text{payoff}(1)] = 0 \) follows immediately by the fact that \( S(\lambda) = \uparrow \forall S \).

We have, consequentially, simply to prove that \( E[\text{payoff}(n-1)] = 0 \Rightarrow E[\text{payoff}(n)] = 0 \) \( \forall S \).

This is, anyway, an obvious consequence of the equations eq.2.8 and eq.2.9.

\[ \text{Theorem 2.1 is not, anyway, a great assurance for Casino’s owner:} \]
\[ \text{in fact it doesn’t exclude that the gambler, if enough lucky, may happen to get a positive payoff for } n \rightarrow \infty. \]
\[ \text{What will definitely assure him is the following:} \]

**Theorem 2.2** *STRONG LAW OF EXCLUDED GAMBLING STRATEGIES*

*For \( n \to \infty \) the set of the lucky-winning strategies tends to the null set*

And here comes the astonishing fact: Theorem 2.2 can’t be proved with purely measure-theoretic concepts.

Our approach will consist in taking von Mises’ axiom 1.2 as a definition of the set of subsequences to which such an axiom applies.

Let us then define the set of collectives \( \text{Collectives} \subset \Sigma^\infty \) as the set of sequences having not enough intrinsic regularity to allow, if they occur, a lucky-winning strategy. Clearly such a definition depends on the class \( \text{Strategies}_{\text{admissible}}(\text{Collectives}) \) of admissible gambling strategies.

It would appear natural, at first, to admit every gambling strategy.

But such a choice would lead immediately to conclude that \( \text{Collectives} = \emptyset \) since given two gambling strategies \( S_0 \) and \( S_1 \) so that:

\[ \text{EXT}[S_i](\bar{x}) \text{ is made only of } i = 0, 1 \forall \bar{x} \in \Sigma^\infty \] \hspace{1cm} (2.10)

we would have clearly that

\[ P_{VM}(a_i | \text{EXT}[S_1](\bar{x})) \neq P_{VM}(a_i | \text{EXT}[S_2](\bar{x})) \forall \bar{x} \in \Sigma^\infty \] \hspace{1cm} (2.11)

The history of the attempts of characterizing in a proper way the class of the admissible gambling strategies is very long and curious [13], [14] and involved many people: Church, Copeland, Dörge, Feller, Kamke, Popper, Reichenbach, Tornier, Waismann and Wald; I will report here only the conceptually more important contributions:

in the thirties Abraham Wald showed that:

\[ (\text{cardinality}(\text{Strategies}_{\text{admissible}}(\text{Collectives})) = \aleph_0) \Rightarrow (\text{Collectives} \neq \emptyset) \] \hspace{1cm} (2.12)
In the fourties, basing on the observation that gambling strategies must be effectively followed, Alonzo Church proposed, according to his Church Thesis \[ \text{Church Thesis} \] to consider admissible a gambling strategy if and only if it is a \textbf{partial recursive function}.

With such an assumption:

\[ \text{Strategies}_{\text{admissible}}(\text{Collectives}) := \Delta^0_0(\Sigma^*) \] (2.13)

that I will adopt from here and beyond, it can be proved that:

\[ P_{\text{Unbiased}}((\text{Collectives})) = 1 \] (2.14)

immediately implying Theorem 2.2.
3 Mises-Wald-Church randomness versus Martin Löf-Solovay-Chaitin randomness

The intuitive idea underlying behind the notion of von Mises - Wald - Church collectives is very similar to the idea underlying the most fascinating concept of Classical Algorithmic Information Theory, namely classical algorithmic randomness. For this reason we will refer, from here and beyond, to von Mises - Wald - Church collectives also as the von Mises - Wald - Church random sequences and will denote them by $\text{RANDOM}_{\text{MWC}}(\Sigma^\infty)$.

The universally accepted notion of classical algorithmic randomness, i.e. Martin Löf-Solovay-Chaitin randomness, is, anyway, stronger [5],[47]:

$$\text{RANDOM}_{\text{MWC}}(\Sigma^\infty) \subset \text{RANDOM}_{\text{MLSC}}(\Sigma^\infty)$$  (3.1)

Surprisingly it has been proved [41] even that:

$$\text{cardinality}(\text{RANDOM}_{\text{MLSC}}(\Sigma^\infty) - \text{RANDOM}_{\text{MWC}}(\Sigma^\infty)) = \aleph_1$$  (3.2)

We will restrict here to analyze Martin Löf algorithmic-measure-way of defining $\text{RANDOM}_{\text{MLSC}}(\Sigma^\infty)$.

Given a classical probability space $\text{CPS} \equiv (M, \mu)$:

Definition 3.1

$S \subset M$ IS A NULL SET OF CPS:

$$\forall \epsilon > 0 \exists \mu_\epsilon \in \text{HALTING}(\mu) : S \subset F_\epsilon \text{ and } \mu(F_\epsilon) < \epsilon$$  (3.3)

Let us introduce the following notions:

Definition 3.2

UNARY PREDICATES ON $M$:

$$\mathcal{P}(M) \equiv \{ p(x) : \text{predicate about } x \in M \}$$  (3.4)

Definition 3.3

TYPICAL PROPERTIES OF CPS:

$$\mathcal{P}(\text{CPS})_{\text{TYPICAL}} \equiv \{ p(x) \in \mathcal{P}(M) : \{ x \in M : p(x) \text{ doesn’t hold } \} \text{ is a null set} \}$$  (3.5)

Example 3.1
TYPICAL PROPERTIES OF A DISCRETE CLASSICAL PROBABILITY SPACE

If CPS is discrete-finite \((M = \{a_1, \ldots, a_n\})\) or discrete-infinite \((M = \{a_n\}_{n \in \mathbb{N}})\) it is natural to assume that \(\mu(\{a_i\}) > 0 \ \forall i\) since an element whose singleton has zero probability can be simply thrown away from the beginning.

It follows, than, that CPS has no null sets and, consequentially, typical properties are simply the holding properties.

Example 3.2

SOME TYPICAL PROPERTY OF THE UNBIASED UNITARY REAL SEGMENT:

The unbiased real unitary segment is the classical probability space \((\mathbb{R}, \mu_{\text{Lebesgue}})\).

All the following predicates are clearly typical:

- \(p_1(x) \equiv \ll x \notin \mathbb{Z} \gg \)
- \(p_2(x) \equiv \ll x \text{ is transcendental} \gg \)
- \(p_3(x) \equiv \ll x \notin \mathbb{Q} \gg \)

Example 3.3

SOME TYPICAL PROPERTY OF THE UNBIASED SPACE OF CBITS’ SEQUENCES:

The unbiased real unitary segment \((\mathbb{R}, \mu_{\text{Lebesgue}})\) and the unbiased space of cbits’ sequences \(UCS := (\Sigma^\infty, \mathcal{P}_{\text{unbiased}})\) are isomorphic as can be immediately proved considering the dyadic expansion of any \(x \in [0,1)\) \[3\].

Clearly such an isomorphism maps the typical predicates \(p^1, p^2, p^3\) of \((\mathbb{R}, \mu_{\text{Lebesgue}})\) in typical properties \(\tilde{p}^1, \tilde{p}^2, \tilde{p}^3\) of \((\Sigma^\infty, \mathcal{P}_{\text{unbiased}})\).

Kolmogorov’s original idea about the characterization of the intrinsic randomness of an individual object was to consider it as more random as more it is conformistic, in the sense of conforming itself to the collectivity belonging to all the overwhelming majorities, i.e. posessing all the typical properties \[47, 4\].

Such an attitude results in the following:

Definition 3.4

SET OF THE KOLMOGOROV-RANDOM ELEMENT OF UCS:

\[
KOLMOGOROV - RANDOM(UCS) \equiv \{ x \in M : p(x) \text{ holds } \forall p \in \mathcal{P}(UCS)_{\text{TYPICAL}} \} \quad (3.6)
\]

But here here comes the following astonishig fact:

Theorem 3.1
NOT EXISTENCE OF KOLMOGOROV RANDOM SEQUENCES OF CBITS

\[ KOLMOGOROV - RANDOM(UCS) = \emptyset \quad (3.7) \]

PROOF:

Let us introduce the following family of unary predicates over \( \Sigma^\infty \) depending on the parameter \( y \in \Sigma^\infty \):

\[ p_y(x) \equiv \langle\langle x \neq y \rangle\rangle \quad (3.8) \]

Clearly:

\[ p_y(x) \in \mathcal{P}(UCS)_{TYPICAL} \quad \forall y \in \Sigma^\infty \quad (3.9) \]

and:

\[ p_x(x) \text{ doesn’t hold} \quad \forall x \in \Sigma^\infty \quad (3.10) \]

So \( p(x) \) is a typical property that is not satisfied by any element of \( \Sigma^\infty \), immediately implying the thesis \( \blacksquare \)

The theorem 3.1 shows that we have to relax the condition that a random sequence of cbits possesses all the typical properties requiring only that it satisfies a proper subclass of typical properties.

The right subclass was proposed by P. Martin Löf who observed that all the Classical Laws of Randomness, i.e. all the properties of Classical Probability Theory that are known to hold with probability one (such as the Law of Large Numbers, the Law of Iterated Logarithm and so on) are effectively-falsifiable in the sense that we can effectively test whether they are violated (though we cannot effectively certify that they are satisfied).

This leads, assuming Church’s Thesis \([28]\) and endowed \( \Sigma^\infty \) with the product topology induced by the discrete topology of \( \Sigma \), to introduce the following notions:

**Definition 3.5**

\( S \subset \Sigma^\infty \) IS ALGORITHMICALLY-OPEN:

\[ (S \text{ is open}) \text{ and } (S = X \Sigma^\infty \text{ X recursively - enumerable}) \quad (3.11) \]

**Definition 3.6**

ALGORITHMIC SEQUENCE OF ALGORITHMICALLY-OPEN SETS:

A sequence \( \{S_n\}_{n \geq 1} \) of algorithmically open sets \( S_n = X_n \Sigma^\infty : \exists X \subset \Sigma^* \times \mathbb{N} \) recursively enumerable with:

\[ X_n = \{ \vec{x} \in \Sigma^* : (\vec{x}, n) \in X \} \quad \forall n \in \mathbb{N}_+ \]
Definition 3.7

$S \subset \Sigma^\infty$ IS AN ALGORITHMICALLY-NULL SET:

$\exists \{G_n\}_{n \geq 1}$ algorithmic sequence of algorithmically-open sets :

$S \subset \bigcap_{n \geq 1} G_n$

and:

$$\text{alg} \lim_{n \to \infty} P_{\text{unbiased}}(G_n) = 0$$

i.e. there exist and increasing, unbounded, recursive function $f : \mathbb{N} \to \mathbb{N}$ so that $P_{\text{unbiased}}(G_n) < \frac{1}{2^k}$ whenever $n \geq f(k)$

Definition 3.8

LAWS OF RANDOMNESS

$L_{\text{randomness}} \equiv \{ p(\bar{x}) \in \mathcal{P}(\Sigma^\infty) : \{ \bar{x} \in \Sigma^\infty : p(\bar{x}) \text{ doesn’t hold} \} \text{ is an algorithmically null set} \}$

(3.12)

Definition 3.9

MARTIN - LÖF - SOLOVAY CHAITIN RANDOM SEQUENCES OF CBITS:

$$\text{RANDOM}_{MLSC}(\Sigma^\infty) \equiv \{ \bar{x} \in \Sigma^\infty : p(\bar{x}) \text{ holds } \forall p \in L_{\text{randomness}} \}$$

(3.13)

The name in definition 3.9 is justified by the fact the Martin - Löf characterization of classical algorithmic randomness resulted to be equivalent both to Solovay’s algorithmic measure-theoretic one and to Chaitin’s definition as algorithmic incompressibility lying at the heart of Classical Algorithmic Information Theory [9], [5], [47].

To appreciate the difference between Mises-Wald-Church randomness and Martin Löf-Solovay-Chaitin randomness let us introduce the following:

Definition 3.10

PROPERTY OF INFINITE RECURRANCE:

$$p_{IR}(\bar{x}) := << \text{cardinality}\{n \in \mathbb{N} : \frac{N(a_1|\bar{x}(n))}{n} = \frac{1}{2} \} = \aleph_0 >>$$

(3.14)

$p_{IR}(\cdot)$ may be easily shown to be a law of randomness.

Anyway in 1939 J. Ville proved the following [47]:

Theorem 3.2
VILLE’S THEOREM

\( \{ \bar{x} \in \text{RANDOM}_{MWC} : p_{IR}(\bar{x}) \text{ holds} \} \neq \emptyset \quad (3.15) \)

Thus there exist Mises-Wald-Church random sequences that if could occur at Casino when we are playing as in the example \[1.1\] would give rise to the following curious situation:

\[
PAYOFF(n) \geq 0 \forall n \in \mathbb{N} \text{ and } \lim_{n \to \infty} PAYOFF(n) = 0 \quad (3.16)
\]
i.e., though satisfying the Law of Excluded Classical Gambling Systems, would make us not losers for any finite time.

Obviously, not being Martin L"of-Solovay-Chaitin random, they won’t occur with certainty.
4 Quantum Gambling in the framework of Quantum Statistical Decision Theory

Quantum Decision Theory was invented by P.A. Benioff [2] and extensively developed by C.W. Helstrom [18] [1]. A renewed interest in such field has recently grown up in the framework of Quantum Game Theory [22], [4].

As in the classical case we can always interpret a quantum decision problem as a quantum gambling situation, with the utility function playing the role of the payoff.

Let us consider a gambler going to a Quantum Casino in which the croupier, at each turn $n$, throws a quantum coin.

Such a situation may be interpreted in different ways giving rise to different types of Quantum Casinos.

Definition 4.1

FIRST KIND QUANTUM CASINO:
a quantum casino specified by the following rules:

1. At each turn $n$ the croupier extracts with unbiased probability a pure state $|\psi > (n) \in \mathbb{H}_2$, where $\mathbb{H}_2$ is the one qubit Hilbert space.

2. Before each quantum coin toss the gambler can decide, according to a direct gambling strategy, among the following possibilities:

   • to bet one fiche on a vector $|\alpha > \in \mathbb{H}_2$
   • not to bet at the turn

3. If he decides for the first option it will happens that:

   • he wins a fiche if the distance among $|\psi > (n)$ and $|\alpha >$ is less or equal to fixed quantity $\epsilon_{\text{Casino}}$.
   • he loses the betted fiche if the distance among $|\psi > (n)$ and $|\alpha >$ is greater than $\epsilon_{\text{Casino}}$

But it is also possible to see the result of a quantum coin toss as a mixed state, resulting in the following:

Definition 4.2

SECOND KIND QUANTUM CASINO:
a quantum casino specified by the following rules:

1. At each turn $n$ the croupier extracts with unbiased (quantum) probability a density matrix $\rho_n$ on the one qubit alphabet $\mathbb{H}_2$.

I will denote from here and beyond with $\mathcal{D}(\mathbb{H})$ the set of density matrices on the Hilbert space $\mathbb{H}$.

32
2. Before each quantum coin toss the gambler can decide, according to a **direct gambling strategy**, among the following possibilities:

- to bet one fiche on a density matrix $\sigma \in \mathbb{H}_2$
- not to bet at the turn

3. If he decides for the first option it will happen that:

- he wins a fiche if the distance among $\sigma$ and $\rho_n$ is less or equal to fixed quantity $\epsilon_{\text{Casino}}$
- he loses the betted fiche if the distance among $\sigma$ and $\rho_n$ is greater than $\epsilon_{\text{Casino}}$

To complete the definition of first and second kind quantum casinos (definition 4.1 and definition 4.2) we have to clarify:

1. what we mean by the distance of pure and mixed states on an Hilbert space $\mathbb{H}$.
2. what we mean by a **direct gambling strategy**

The more physical notions of distance between quantum states are the following:

1. **Definition 4.3**

**QUANTUM TRACE DISTANCE ON $D(\mathbb{H})$:**

$$D(\rho_1, \rho_2) := \frac{1}{2}\|\rho_1 - \rho_2\|_1 = \frac{1}{2}Tr(|\rho_1 - \rho_2|)$$ (4.1)

with:

$$|a| := \sqrt{a^\dagger a} \quad a \in B(\mathbb{H})$$ (4.2)

and where given $n \in \{1, 2, \cdots, \infty\}$ and denoted by $B(\mathbb{H})$ the Von Neumann algebra of the bounded linear operators on $\mathbb{H}$:

**Definition 4.4**

$n^{th}$ **OPERATORIAL NORM ON $B(\mathbb{H})$**

$$\|a\|_n := (Tr(a^\dagger a)^{\frac{n}{2}})^{\frac{1}{n}}$$ (4.3)

The $n = \infty$ norm is called the **operator norm** and will be considered from here and beyond as the default norm on $B(\mathbb{H})$:

$$\|a\| := \|a\|_\infty \quad a \in B(\mathbb{H})$$ (4.4)
2. Definition 4.5

QUANTUM ANGLE DISTANCE ON $\mathcal{D}(\mathbb{H})$:

$$A(\rho_1, \rho_2) := \arccos F(\rho_1, \rho_2)$$  \hspace{1cm} (4.5)

where:

Definition 4.6

QUANTUM FIDELITY ON $\mathcal{D}(\mathbb{H})$:

$$F(\rho_1, \rho_2) := \text{Tr} \sqrt{\sqrt{\rho_1} \rho_2 \sqrt{\rho_1}}$$  \hspace{1cm} (4.6)

As will become clearer in the general mathematical framework of Quantum Probability Theory \cite{14, 16, 18, 19, 21, 22, 24} we will introduce later, the definition 4.3 is the natural quantum corrispective of the following distance on the set $\mathcal{D}(\Sigma)$ of the probability distributions on a discrete set $\Sigma$:

Definition 4.7

CLASSICAL TRACE DISTANCE ON $\mathcal{D}(\Sigma)$:

$$D(p_1, p_2) := \frac{1}{2} \sum_{x \in \Sigma} |p_1(x) - p_2(x)|$$  \hspace{1cm} (4.7)

The intuitive meaning of the definition 4.7 is clarified by the following \cite{10}:

Theorem 4.1

CLASSICAL TRACE DISTANCE AS DISTANCE OF THE CLASSICAL PROBABILITY OF ANTIPODAL EVENTS:

$$D(p_1, p_2) = \max_{e \in 2^\Sigma} |p_1(e) - p_2(e)|$$  \hspace{1cm} (4.8)

It is remarkable that an analogous interpretation is admissible also in the quantum case \cite{10}:

Theorem 4.2

QUANTUM TRACE DISTANCE AS DISTANCE OF THE QUANTUM PROBABILITY OF ANTIPODAL EVENTS:

$$D(\rho_1, \rho_2) = \max_{P \in B(\mathbb{H})^+} \text{Tr} P (\rho_1 - \rho_2)$$  \hspace{1cm} (4.9)

where $B(\mathbb{H})^+$ denotes the positive cone of the Von Neumann algebra $B(\mathbb{H})$ of the bounded linear operators on $\mathbb{H}$, i.e. the set of positive bounded linear operators on $\mathbb{H}$.

Theorem 4.2 has an immediate interpretation in terms of quantum measurements:
Definition 4.8

QUANTUM MEASUREMENT ON A SYSTEM S:

a dynamical evolution of the system S, to be necessarily (owed to the endophysics - incompleteness of Quantum Theory [33], [41], [30], [37] not allowing a description from inside of its own means of verification) described as an open system, specified by a collection \( \{M_r\}_{r \in R} \) of trace preserving quantum operations on S’s Hilbert space, where R is the set of the possible measurement outcomes, such that:

\[
\sum_{r \in R} M_r^\dagger M_r = I
\]  

(4.10)

If the state of S before the measurement is \( |\psi> \):

- the probability that result \( r \in R \) occurs is given by:

\[
p(r) = <\psi|M_r^\dagger M_r|\psi>
\]

(4.11)

- If the result \( r \in R \) occurs, then the state of S after the measurement is:

\[
\frac{M_r|\psi>}{\sqrt{<\psi|M_r^\dagger M_r|\psi>}}
\]

(4.12)

The collection of trace preserving quantum operations \( \{M_r\}_{r \in R} \) specifying a quantum measurement induces the following positive operator valued measure (POVM) \( \{P_r\}_{r \in R} \):

\[
P_r := M_r^\dagger M_r \quad \forall r \in R
\]

(4.13)

Furthermore it may be shown that the POVM \( \{P_r\}_{r \in R} \) identifies univoquely the set of trace preserving quantum operations \( \{M_r\}_{r \in R} \) and can, consequentially, be seen as a different characterization of the same quantum measurement.

Theorem 4.2 states that \( D(\rho_1, \rho_2) \) is the maximal distance of the classical probabilities of a measurement outcome between the case in which the state before the measurement is \( \rho_1 \) and the case in which the state before the measurement is \( \rho_2 \).

Also the quantum angle distance is the natural quantum corrispective of a distance on the set \( D(\Sigma) \) of the probability distributions on a discrete set \( \Sigma \), namely:

Definition 4.9

CLASSICAL ANGLE DISTANCE ON \( D(\Sigma) \):

\[
D(p_1, p_2) := \arccos F(p_1, p_2)
\]

(4.14)

where:
Definition 4.10

CLASSICAL FIDELITY ON $\mathcal{D}(\Sigma)$:

$$F(p_1, p_2) := \sum_{x \in \Sigma} \sqrt{p_1(x)p_2(x)} \quad (4.15)$$

The name of both the classical and the quantum angle distances is owed to their interpretation as the angle between two points of the unit sphere: in the classical case such interpretation is self-evident as it appears considering the two distributions as versors

$$\begin{pmatrix} p_1(x_1) \\ p_1(x_2) \\ \vdots \end{pmatrix}, \begin{pmatrix} p_2(x_1) \\ p_2(x_2) \\ \vdots \end{pmatrix}$$

where $x_1, x_2, \ldots \in \Sigma$.

In the quantum case such an interpretation arises from the following:

Theorem 4.3

UHLMANN’S THEOREM:

HP:

$$\rho_1, \rho_2 \in \mathcal{D}(\mathbb{H})$$

TH:

$$F(\rho_1, \rho_2) = \max_{|\psi_1 > \in PUR(\rho_1, \mathbb{H}) : |\psi_2 > \in PUR(\rho_2, \mathbb{H})} | < \psi_1 | \psi_2 > |$$

where, given two generic Hilbert spaces $\mathbb{H}_A$ and $\mathbb{H}_B$ and a density matrix $\rho \in \mathcal{D}(\mathbb{H}_A)$:

Definition 4.11

PURIFICATIONS OF $\rho$ WITH RESPECT TO $\mathbb{H}_B$:

$$PUR(\rho, \mathbb{H}_B) := \{ |\psi > \in \mathbb{H}_A \bigotimes \mathbb{H}_B : Tr_{\mathbb{H}_B} |\psi > = \rho \} \quad (4.16)$$

So the cosin of the angle distance between two density matrices is equal to the maximum inner product between purifications of such density matrices.

Now here comes the rub:

which among the above two distances have we to choice in the definition of a second kind Quantum Casino?

Fortunately it can be proved that the trace distance and the angle distance are qualitatively equivalent. Namely:

Theorem 4.4
QUALITATIVE EQUIVALENCE OF TRACE AND ANGLE DISTANCES ON STATES:

\[ 1 - F(\rho_1, \rho_2) \leq D(\rho_1, \rho_2) \leq \sqrt{1 - F(\rho_1, \rho_2)} \] (4.17)

and, conseuqentially, it doesn’t matter which of them we use in order to define a second kind Quantum Casino.

For the case we are interested to in which the underlying Hilbert space is the one qubit Hilbert space \( \mathbb{H}_2 \), furthermore, the adoption of the trace distance may be preferred since it satisfies the following:

**Theorem 4.5**

**QUANTUM TRACE DISTANCE IN TERMS OF THE BLOCH SPHERE:**

\[ D(\text{Bloch}(\vec{r}_1), \text{Bloch}(\vec{r}_2)) = \frac{\|\vec{r}_1 - \vec{r}_2\|}{2} \quad \forall \vec{r}_1, \vec{r}_2 \in \text{Ball}(2) \] (4.18)

with:

**Definition 4.12**

**BLOCH SPHERE BIJECTION:**

\[ \text{Bloch} : \text{Ball}(2) \to \mathcal{D}(\mathbb{H}_2) : \]

\[ \text{Bloch}(\vec{r}) := \frac{\mathbb{I} + \vec{r} \cdot \hat{\sigma}}{2} \] (4.19)

where \( \text{Ball}(2) := \{ \vec{r} \in \mathbb{R}^3 : \|\vec{r}\| \leq 1 \} \) is the unit-radius 2-ball while \( \hat{\sigma} := \begin{pmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \end{pmatrix} \) is the vector of the Pauli matrices.

Let us observe that the extraction with unbiased probability of an element of \( \mathcal{D}(\mathbb{H}_2) \) involved in the definition 4.2 may be reconducted, through the definition 4.12, to the extraction of a value of uniform-distributed random point on the unit radius 2-ball \( \text{Ball}(2) \).

Let us, now, clarify what we mean by a direct gambling strategy. To make his decision at the \( n \)th turn, the gambler can take in consideration the result of all the previous \( n-1 \) quantum coin tosses.

He can do this in two different ways:

- he can think on the direct products of the previous outcomes; we will call such a strategy a **direct gambling strategy**

- he can think on the tensor products of the previous outcomes; we will call such a strategy a **tensor gambling strategy**
In a first kind and second kind Quantum Casino the gambler has to play according to a direct gambling strategy.

We will introduce, later a third kind of Quantum Casino, in which the gambler has to play according to a tensor gambling strategy.

The direct gambling strategies according to which the gambler plays in a first kind and second kind Quantum Casino will be called, respectively, first kind and second kind quantum gambling strategies and defined in the following way:

**Definition 4.13**

FIRST KIND QUANTUM GAMBLING STRATEGY:

\[ S : \mathbb{H}_2 \to \mathbb{H}_2 \]

**Definition 4.14**

SECOND KIND QUANTUM GAMBLING STRATEGY:

\[ S : \mathcal{D}(\mathbb{H}_2) \to \mathcal{D}(\mathbb{H}_2) \]

Let us now consider the sets \( \mathbb{H}_2^\infty \) and \( \mathcal{D}(\mathbb{H}_2)^\infty \) of sequences of, respectively, one qubit vectors and one qubit density matrices.

Our objective is to characterize two subsets \( \mathcal{Q}\text{Colle}ctives^1 \subset \mathbb{H}_2^\infty \) and \( \mathcal{Q}\text{Colle}ctives^2 \subset \mathcal{D}(\mathbb{H}_2)^\infty \), that we will call, respectively, first kind quantum collectives and second kind quantum collectives, defined by the condition of satisfying Von Mises's axiom\(^1,2\) when the class of the first kind quantum admissible gambling strategies \( \mathcal{Q}\text{Strategies}_{\text{admissible}}(\mathcal{Q}\text{Colle}ctives^1) \) and the class of the second kind quantum admissible gambling strategies \( \mathcal{Q}\text{Strategies}_{\text{admissible}}(\mathcal{Q}\text{Colle}ctives^2) \) are chosen according to a proper algorithmic-effectiveness characterization specular to the classical one of eq.\(^2,13\).

We arrive, consequentially, to the following definitions:

**Definition 4.15**

FIRST KIND QUANTUM COLLECTIVES:

\[ \mathcal{Q}\text{Colle}ctives^1 \subset \mathbb{H}_2^\infty \] induced by the axiom\(^1,2\) and the assumptions that the first kind quantum admissible gambling strategies are nothing but the quantum algorithms on \( \mathbb{H}_2^\infty \):

\[ \mathcal{Q}\text{Strategies}_{\text{admissible}}(\mathcal{Q}\text{Colle}ctives^1) := \mathcal{Q} - \text{Algorithms}(\mathbb{H}_2^\infty) \quad (4.20) \]

**Definition 4.16**

SECOND KIND QUANTUM COLLECTIVES:

\[ \mathcal{Q}\text{Colle}ctives^2 \subset \mathcal{D}(\mathbb{H}_2)^\infty \] induced by the axiom\(^1,2\) and the assumption that the second kind quantum admissible gambling strategies are nothing but the quantum algorithms on \( \mathcal{D}(\mathbb{H}_2)^\infty \):

\[ \mathcal{Q}\text{Strategies}_{\text{admissible}}(\mathcal{Q}\text{Colle}ctives^2) := \mathcal{Q} - \text{Algorithms}(\mathcal{D}(\mathbb{H}_2)^\infty) \quad (4.21) \]
But we are then faced to the following dramatic question: who is the class of quantum algorithms? The answer, which ever it is, touches the extremely subtle and controversial debate about the relation existing between Church’s Thesis and Quantum Mechanics for which we demand to our paper [38]. Using the terminology therein introduced, let us recall here briefly the key points:

• Church’s Thesis doesn’t imply the equality of Physically-classical computability ($C_{\Phi}$-computability) and Physically-quantistical computability ($Q_{\Phi}$-computability) of Mathematically-nonclassical objects ($NC_{M}$-objects)

• as far as Computability of Mathematically-nonclassical objects ($NC_{M}$-objects) is concerned the proposal of analogous theses, such as the Pour-El Thesis [36], [32], playing for special classes of Mathematically-nonclassical objects the role played for Mathematically-classical objects ($C_{M}$-objects) by Church’s Thesis should be considered with great attention

Let us now pass to analyze third kind Quantum Casinos. As we already announced in such a Quantum Casino the gambler has to play according to a tensor gambling strategy. This means that he will consider as the quantum corrispective of a string of n qubits $\vec{x} = x_1 \cdots x_n$ not the string of vectors $|x_1 \rangle \cdots |x_n \rangle \in \mathbb{H}_n^2$ but the vector $|x_1 \cdots x_n \rangle \in \mathbb{H}_2^\otimes n$, where:

Definition 4.17

$n^{th}$ TENSOR POWER OF THE HILBERT SPACE $\mathbb{H}$:

$$\mathbb{H}^\otimes n := \bigotimes_{k=1}^{n} \mathbb{H}$$

(4.22)

In particular:

Definition 4.18

SPACE OF THE QUANTUM STRINGS OF n QUBITS: $\mathbb{H}_2^\otimes n$

As to the definition of the quantum analogue of the set of strings of an arbitrary (but finite) number of qubits $\{0,1\}$ the substituition of direct products by tensor products automatically leads to the substitution of the union operator of the definition [1.1] by the direct sum operator. So given an Hilbert space $\mathbb{H}$, we are led to the following definition:

Definition 4.19
QUANTUM STRINGS OVER $\mathbb{H}$:

\[
\mathbb{H}^\otimes * := \bigoplus_{n \in \mathbb{N}} \mathbb{H}^\otimes n
\]  

and in particular:

**Definition 4.20**

SPACE OF THE QUANTUM STRINGS OF QUBITS: $\mathbb{H}_2^\otimes *$

Clearly $\mathbb{H}^\otimes *$ is nothing but the **Fock space** associated to $\mathbb{H}$.

We can, consequentially, look for a more autentically quantistic definition of a **quantum gambling strategy**, in which **entanglement** is taken into account and used by the gambler in order to maximize his payoff.

To quantify the **entanglement properties** of quantum strings of qubits it is useful, at this point, to introduce the concept of **Schmidt number**.

This requires, first of all, to introduce the following basic theorem [10]:

**Theorem 4.6**

**SCHMIDT DECOMPOSITION**

HP:

$\mathbb{H}_A, \mathbb{H}_B$ finite dimensional Hilbert spaces

$|\psi > \in \mathbb{H}_A \bigotimes \mathbb{H}_B$

**TH:**

There exist $\{|i_A >\}$ orthonormal basis of $\mathbb{H}_A$ and $\{|i_B >\}$ orthonormal basis of $\mathbb{H}_B$ (called the **Schmidt bases** for, respectively, $\mathbb{H}_A$ and $\mathbb{H}_B$) so that:

$|\psi > = \sum_i \lambda_i |i_A > |i_B >$

where the **Schmidt coefficients** $\{ \lambda_i \in \mathbb{R} \cup \{0\} \}$ satisfy the following condition:

$$\sum_i \lambda_i^2 = 1$$

Let us observe that given two finite dimensional Hilbert spaces $\mathbb{H}_A$ and $\mathbb{H}_B$ and a vector $|\psi > \in \mathbb{H}_A \bigotimes \mathbb{H}_B$ the theorem [10] states the existence but not the uniqueness of the Schmidt decomposition of $|\psi >$

Anyway it may be proved that the following quantity is well-defined:
Definition 4.21

SCHMIDT NUMBER OF $|\psi>$:

$$n_{Schmidt}(|\psi>) := \text{cardinality}\{\lambda_i > 0\}$$

We will say that:

Definition 4.22

$|\psi>$ IS ENTANGLED:

$$n_{Schmidt}(|\psi>) > 1$$

Returning to our space of qubit strings let us then introduce the following concept:

Definition 4.23

DEGREE OF ENTANGLEMENT OF THE $n$-QUBIT STRING $|\psi>$$\in \mathbb{H}_2^\otimes n$:

$$d_{\text{entanglement}}(|\psi>) := \max_{k=1,\ldots,n-1} n_{Schmidt}(|\psi>,k) - 1 \quad (4.24)$$

where $n_{Schmidt}(|\psi>,k)$ denotes the Schmidt number of $|\psi>$ with respect to the splitting of $\mathbb{H}_2^\otimes n$ as $\mathbb{H}_2^\otimes k \otimes \mathbb{H}_2^\otimes n-k$.

And what about quantum sequences of qubits?

They will be, clearly, the protagonists of the limit $n \to \infty$ under which a Law of Excluded Quantum Gambling Strategies for autentically quantum strategies, i.e. for third kind Quantum Casinos, may be conjectured to hold.

One could think that the space of quantum sequences over $\mathbb{H}_2$ may be easily defined in terms of the computational bases.

Definition 4.24

COMPUTATIONAL BASIS OF $\mathbb{H}_2$:

$$E := \{ |0>, |1> \} : <0|0> = <1|1> = 1 \quad <0|1> = 0 \quad (4.25)$$

Given any positive integer number $n \geq 3$:

Definition 4.25

COMPUTATIONAL BASIS OF $\mathbb{H}_2^\otimes n$:

$$E_n := \{ |\vec{x}>, \vec{x} \in \{0,1\}^n \} \quad (4.26)$$

Definition 4.26
COMPUTATIONAL BASIS OF $\mathbb{H}_2^\otimes$: 

$$E_\star := \{|\vec{x}>, \vec{x} \in \{0,1\}^*\} \quad (4.27)$$

The generic vector of $\mathbb{H}_2^\otimes^*$ is then given by a linear combination of the form $\sum_{\vec{x} \in \Sigma^*} c_{\vec{x}} |\vec{x}>$. 

As an example let us recall the famous Einstein-Podolsky-Rosen (EPR) states:

$$|00> + |11> \sqrt{2}$$

$$|00> - |11> \sqrt{2}$$

$$|01> + |10> \sqrt{2}$$

$$|01> - |10> \sqrt{2}$$

or the equally famous Greenberger-Horne-Zeilinger (GHZ) state:

$$|1100> - |0011> \sqrt{2}$$

So one could think to consider the set:

$$E_\infty := \{|\vec{x}>, \vec{x} \in \{0,1\}^\infty\} \quad (4.28)$$

and to define the Hilbert space of qubits’ sequences by the imposition that $E_\infty$ is a basis of it, i.e. to introduce the following notion:

$$\mathbb{H}_2^\otimes^\infty := \{ \int_{\{0,1\}^\infty} dP_{\text{unbiased}}(\vec{x})c(\vec{x}) |\vec{x}>, c(\vec{x}) \in \mathbb{C} \forall \vec{x} \in \{0,1\}^\infty\} \quad (4.29)$$

As we saw in the example the classical probability spaces $(\Sigma^\infty, P_{\text{unbiased}})$ and $(\{0,1\}, \mu_{\text{Lebesgue}})$ are isomorphic.

So $\mathbb{H}_2^\otimes^\infty$ could seem very similar to the space of Dirac’s kets $\mathcal{H}_{ket}$ generated by position autokets of a quantum nonrelativistic particle living on the unitary segment $[0,1)$:

$$\hat{x}|x> = x|x> \quad x \in [0,1) \quad (4.30)$$

$$<x_1|x_2> = \delta(x_1 - x_2) \quad (4.31)$$
\[ \int_{(0,1)} d\mu_{\text{Lebesgue}} |x><x| = \hat{1} \] (4.32)

One could even be tempted to introduce by analogy a sequence operator \( \hat{x} \) on \( H_{\otimes\infty} \) having \( E_{\infty} \) as Dirac’s autokets:

\[ \hat{x}|\bar{x}> = \bar{x}|\bar{x}> \quad \bar{x} \in \Sigma_{\infty} \] (4.33)

\[ <\bar{x}_1|\bar{x}_2> = \delta(\bar{x}_1 - \bar{x}_2) \] (4.34)

\[ \int_{(0,1)} dP_{\text{unbiased}} |\bar{x}><\bar{x}| = \hat{1} \] (4.35)

Now it is well known that Dirac’s original bra and ket formalism \[1\] is mathematically nonrigorous \[39\], \[40\], \[20\]: the space of kets \( H_{\text{ket}} \) is not an Hilbert space; what constitutes a well-defined Hilbert space is the set \( L^2((0,1), \mu_{\text{Lebesgue}}) \) of the wave functions \( \psi|_{\alpha} (x) := <x|\alpha> \).

Now, if the problem in the introduction of \( H_{\otimes\infty} \) was only this one, it would be justified to see it as a false problem, a mathematical pignolery with no physical counterpart behind.

Indeed, by a restyling operation involving the formal sophistication of looking at \( H_{\text{ket}} \) as a rigged Hilbert space, Dirac’s bra and ket formalism may be recasted a completely rigorous way.

The problem we are facing, anyway, is terribly more serious and is not only a matter of form but of substance:

it ultimately concerns the inadequacy of the naive-Hilbert space formalism of the thermodynamical limit of Quantum Statistical Mechanics, whose solution requires the introduction of some notion of Quantum Probability Theory \[4\], \[13\], \[23\], \[24\], \[31\], \[29\], \[26\].

**Definition 4.27**

ALGEBRAIC PROBABILITY SPACE: \( (A, \omega) \) where:

- \( A \) is a Von Neumann algebra
- \( \omega \) is a state on \( A \)

The notion of algebraic probability space is a noncommutative generalization of the notion of classical probability space as is implied by the following considerations:

\(^1\)Great caution must be taken in handling the locutions Quantum Probability Theory and quantum probability space in that they are used by different schools with different meanings. By adhering here to Luigi Accardi - school’s terminology we invite the (eventual) lectors not to make confusion between the definition \[4.28\] and Stanley P. Gudder’s definition of a quantum probability space as a sample space endowed with a probability amplitude \[14\] or the lattice-theoretic definitions such as the Enrico Beltrametti - Gianni Cassinelli’s one consisting in substituting as halting set of a probability measure the classical Boolean lattice of a \( \sigma \) - algebra of subsets of the sample space with a generic orthomodular lattice \[8\] or Pavel Pták - Sylvia Pulmannova’s definition of a generalized probability space as a generic couple made up by a sum-logic endowed with a state \[34\]
1. A generic classical probability space \((X, \mu)\) may be equivalently seen as the abelian algebraic probability space \((L^\infty(X, \mu), \omega_\mu)\), where:

\[
\omega_\mu(A) \in S(A) \\
\omega_\mu(a) := \int_X a(x)d\mu(x)
\]

(4.36)

with \(S(A)\) denoting the set of states over \(A\).

2. Given a generic abelian algebraic probability space \((A, \omega)\) there exists a classical probability space \((X, \mu)\) and a \(*\)-isomorphism \(I_{GELFAND} : A \to L^\infty(X, \mu)\) called the Gelfand isomorphism under which the state \(\omega \in S(A)\) corresponds to the state \(\omega_\mu \in L^\infty(X, \mu)\).

**Definition 4.28**

QUANTUM PROBABILITY SPACE: a non-abelian algebraic probability space

Given an algebraic random variable \(a \in A\) on the algebraic probability space \((A, \omega)\) and a number \(n \in \mathbb{N}_+\):

**Definition 4.29**

\(n^{th}\) MOMENT OF \(a\):

\[
M_n(a) \equiv \omega(a^n)
\]

(4.37)

The first moment \(M_1(a)\) of a noncommutative random variable \(a \in A\) on \((A, \omega)\) is usually called its expectation value and denoted by \(E(a)\).

An other important quantity to mention is the following:

**Definition 4.30**

VARIANCE OF \(a\):

\[
Var(a) \equiv E(a^2) - (E(a))^2
\]

(4.38)

playing a rule in the following fundamental:

**Theorem 4.7**

THEOREM OF INDETERMINATION:

\[
|E(\frac{[a,b]}{2i})| \leq \sqrt{Var(a)}\sqrt{Var(b)} \quad \forall a, b \in A
\]

(4.39)

where \([a,b] \equiv ab - ba\) is the commutator between \(a\) and \(b\).

Given a classical set \(M\), let us introduce the following terminology:

**Definition 4.31**
M IS OF TYPE $I_n$:

\[
\text{cardinality}(M) = n
\]  
\hspace{1cm} (4.40)

\textbf{Definition 4.32}

M IS OF TYPE $I_\infty$:

\[
\text{cardinality}(M) = \aleph_0
\]  
\hspace{1cm} (4.41)

\textbf{Definition 4.33}

M IS OF TYPE $II_1$:

\[
\text{cardinality}(M) = \aleph_1 \text{ and } M \text{ admits an unbiased probability measure } P_{\text{unbiased}}
\]  
\hspace{1cm} (4.42)

\textbf{Definition 4.34}

M IS OF TYPE $II_\infty$:

\[
\text{cardinality}(M) = \aleph_1 \text{ and } M \text{ doesn’t admit an unbiased probability measure } P_{\text{unbiased}}
\]  
\hspace{1cm} (4.43)

\textbf{Example 4.1}

THE REAL AXIS AND ITS UNITARY SEGMENT

Both the whole real axis and the its unitary segment have the continuum power:

\[
\text{cardinality}(\mathbb{R}) = \text{cardinality}([0, 1)) = \aleph_1
\]  
\hspace{1cm} (4.44)

The Lebesgue measure is an unbiased probability measure for the unitary segment while it is not a probability measure on the whole real axis since it is not normalizable. Hence:

\[
\text{Type}([0, 1)) = II_1
\]  
\hspace{1cm} (4.45)

\[
\text{Type}(\mathbb{R}) = II_\infty
\]  
\hspace{1cm} (4.46)
An analogous situation exists in the quantum case, where the rule of a quantum set is played by a non-abelian Von Neumann algebra $A$ or, better, the building blocks it is made of, i.e. the factors contributing to its factor decomposition.

Introduced the following terminology:

**Definition 4.35**

QUANTUM UNBIASED PROBABILITY MEASURE ON $A$: a finite, faithful, normal trace on $A$

and assumed for simplicity that $A$ is itself a factor we have a classification very similar to the classical one:

**Definition 4.36**

A IS OF TYPE $I_n$:

\[
\text{cardinality}(\text{range}(d)) = n
\]  
(4.47)

**Definition 4.37**

A IS OF TYPE $I_\infty$:

\[
\text{cardinality}(\text{range}(d)) = \aleph_0
\]  
(4.48)

**Definition 4.38**

A IS OF TYPE $II_1$:

\[
\text{cardinality}(\text{range}(d)) = \aleph_1 \text{ and } A \text{ admits an unbiased quantum probability measure } P_{\text{unbiased}}
\]  
(4.49)

**Definition 4.39**

Great caution must be taken in handling the locution quantum set since it is used by various authors with completely different meanings. Our approach consists in considering the words quantum and noncommutative as synonomous and adhering to the phylosophy underlying Noncommutative (Quantum) Geometry according to which one starts from the Gelfand isomorphism to introduce noncommutative (quantum) spaces and then define on them the whole hierarchy of more and more refined structures: measure-theoretic (the only one playing a role in this paper), topological, differential-geometric, and (pseudo)riemannian-geometric. Such an acception of the locution quantum set is completely different both from the exoteric quantum sets of Takeuty [16] and from the even more exoteric quantum sets of Finkelstein [12]. Finally it must be remarked that the same expression quantum set may be someway misleading because someone could be tempted to look at it erroneously as a departure from Classical Set Theory for a new set theory: this is absolutely not the case since a Von Neumann algebra is, in particular, obviously a classical set!!!
A IS OF TYPE \( II_\infty \):

\[
\text{cardinality}(\text{range}(d))) = \aleph_1 \text{ and } \text{A doesn’t admit an unbiased probability measure} \ P_{unbiased}
\]

where \( d \) is an arbitrary dimension function on the complete, orthomodular lattice \( \text{Pr}(A) \) of the projections of \( A \) \[34\], \[35\], \[15\], \[16\].

It must be said, for completeness, that in the quantum case there exist a suppletive one-parameter family of cases \( Type(A) = III_\lambda \lambda \in [0,1] \) having no corrispective in the classical case that (fortunately) won’t have no rule in this paper.

These considerations justify the introduction of the following terminology:

**Definition 4.40**

ONE QUBIT ALGEBRAIC ALPHABET:

\[
\Sigma_{\text{alg}} := M_2(\mathbb{C})
\]

**Definition 4.41**

ALGEBRAIC SPACE OF QUANTUM STRING OF \( n \) QUBITS:

\[
\Sigma_{\text{alg}}^{\otimes n} := M_2(\mathbb{C})^{\otimes n}
\]

**Definition 4.42**

ALGEBRAIC SPACE OF QUANTUM STRINGS OF QUBITS:

\[
\Sigma_{\text{alg}}^{\star} := \bigoplus_N \Sigma_{\text{alg}}^{\otimes n}
\]

Obviously \( \Sigma_{\text{alg}} \) is a \( I_2 \)-factor; furthermore \( \Sigma_{\text{alg}}^{\otimes n} \), being \( \star \)-isomorphic to \( M_n(\mathbb{C}) \), is a \( I_n \)-factor. Hence it is also \( \star \)-isomorphic to \( B(\mathbb{H}_n) \) (with \( \mathbb{H}_n \) denoting an \( n \)-dimensional Hilbert space) and admits the unbiased quantum probability measure \( \tau_n(\cdot) := \frac{1}{n}Tr(\cdot) \).

Furthermore every state \( \omega \in S(\Sigma_{\text{alg}}^{\otimes n}) \) on it is normal and hence there exists a density matrix \( \rho_\omega \in D(\mathbb{H}_2^{\otimes n}) \) so that:

\[
\omega(a) = Tr(\rho_\omega a) \quad \forall a \in \Sigma_{\text{alg}}^{\otimes n}
\]

Eq.4.54 shows that the algebraic characterization of quantum strings is absolutely equivalent to the usual Hilbert space one based on the definitions definition4.18 and definition4.20.
The whole operator algebraic machinery would then seem (as is, indeed, often considered by not enough acculturated physicists) an arbitrary mathematical sophistication to recast in a hieratic mathematical language simple physical statements.

As far as our issues is concerned, if for \( n \to \infty \) \( \Sigma^{\otimes n}_{alg} \) tended to an \( I_\infty \)-factor \( \Sigma^{\otimes \infty}_{alg} \) this would be indeed true since, in this case, there would exist an infinite-dimensional Hilbert space \( \mathbb{H} \) such that:

1. \( \Sigma^{\otimes \infty}_{alg} \) would be \( \ast \)-isomorphic to \( B(\mathbb{H}) \)

2. every state on \( \Sigma^{\otimes \infty}_{alg} \) \( \omega \in S(\Sigma^{\otimes \infty}_{alg}) \) would be normal and hence there would exist a density matrix \( \rho_\omega \in D(\mathbb{H}^{\otimes \infty}_2) \) so that:

\[
\omega(a) = \text{Tr}(\rho_\omega a) \quad \forall a \in \Sigma^{\otimes \infty}_{alg}
\]

admitting to recast again the analysis in the usual Hilbert space formulation (at the price of some quantum-logical subtility owed to the fact that the lattice \( \text{Pr}(\Sigma^{\otimes \infty}_{alg}) \) in this case wouldn’t be modular)

But the factor \( \Sigma^{\otimes \infty}_{alg} \) to which \( \Sigma^{\otimes n}_{alg} \) tends for \( n \to \infty \) is of type \( II_1 \).

This can be shown in the following way:

the restriction of the unbiased quantum probability measure \( \tau_n \) to \( \text{Pr}(\Sigma^{\otimes n}_{alg}) \) is a dimension function \( d_n \) so that:

\[
\text{Range}(d_n) = \left\{ \frac{k}{2^n} : k = 0, 1, 2, \ldots, 2^n \right\}
\]

Since:

\[
\lim_{n \to \infty} \text{cardinality}\left( \left\{ \frac{k}{2^n} : k = 0, 1, 2, \ldots, 2^n \right\} \right) = \aleph_1
\]

it follows that the infinite tensor product of \( M_2(\mathbb{C}) \) can’t be of type \( I_\infty \) and, consequentially:

1. it is not \( \ast \)-isomorphic to a \( B(\mathbb{H}) \)

2. a state on it is not, in general, normal and, hence, can’t be represented by a density matrix

Making more rigorous the previous informal arguments let us introduce the following:

**Definition 4.43**

**Algebraic Space of Quantum Sequences of Qubits:**

\[
\Sigma^{\otimes \infty}_{alg} := \text{completion}_{\| \cdot \|} \bigotimes_N M_2(\mathbb{C})
\]
It can be proved that $\Sigma^\otimes\infty_{alg}$ admits an unbiased quantum probability measure and is then of type $II_1$ (implying that the lattice $Pr(\Sigma^\otimes\infty_{alg})$ is modular).

Let us finally introduce the following notions:

**Definition 4.44**

ALGEBRAIC QUANTUM COIN: a quantum random variable on the quantum probability space $(M_2(\mathbb{C}), \tau_2)$

**Definition 4.45**

THIRD KIND QUANTUM CASINO:

a quantum casino specified by the following rules:

1. At each turn $n$ the croupier throws an algebraic quantum coin $A_n$ obtaining a value $a_n \in \Sigma_{alg}$

2. Before each algebraic quantum coin toss the gambler can decide, by adopting a quantum gambling strategy, among the following possibilities:
   - to bet one fiche on an a letter $b \in \Sigma_{alg}$
   - not to bet at the turn

3. If he decides for the first option it will happens that:
   - he wins a fiche if the distance among $a_n$ and $b$ $d(a_n, b) := \|a_n - b\|$ is less or equal to a fixed quantity $\epsilon_{Casino}$.
   - he loses the betted fiche if the distance among $a_n$ and $b$ $d(a_n, b) := \|a_n - b\|$ is greater than $\epsilon_{Casino}$

where the adoption of a tensor gambling strategy is formalized in terms of the following notion:

**Definition 4.46**

THIRD KIND QUANTUM GAMBLING STRATEGY:

$S : \Sigma^\otimes\infty_{alg} \rightarrow M_2(\mathbb{C})$

The concrete way in which the gambler applies, in every kind of Quantum Casino, the chosen strategy $S$ is always the same:

- if $S$ doesn’t halt on the previous game history he doesn’t bet at the next turn
- if $S$ halts on on the past game history he bets $S$(previous game history)

Let us denote by $\bar{a} \in \Sigma^\otimes\infty_{alg}$ the occurred quantum sequence of qubits and with $\bar{a}(n) := a_1 \otimes \cdots \otimes a_n \in \Sigma^\otimes n_{alg}$ its quantum prefix of length $n$, i.e. the quantum string of the results of the first $n$ quantum coin tosses.

Quantum Casinos could seem, at this point, an abstruse mathematical concept; they are, anyway, as concrete as classical casinos and may be concretely simulated by the following Mathematica code:
Quantum-Gambling Package

January 9, 2001

\textbf{In[1]}:= \texttt{Off[General::spell1];}
\textbf{In[2]}:= \texttt{\textless \textless \text{LinearAlgebra\textbackslash MatrixManipulation};}
\textbf{In[3]}:= \texttt{\textless \textless \text{Calculus\textbackslash DiracDelta};}
\textbf{In[4]}:= \texttt{\textless \textless \text{DiscreteMath\textbackslash Combinatorica};}
\textbf{In[5]}:= \texttt{\textless \textless \text{Statistics\textbackslash DataManipulation};}
\textbf{In[6]}:= \texttt{\textless \textless \text{Graphics\textbackslash Graphics};}
\textbf{In[7]}:= (* * * for Mathematica an array is a list of depth 2 whose i^{th} element is the list of the elements of the i^{th} row. So a column vector \binom{x_1}{x_2} is a list of the form \{x_1, x_2\} while a row vector \{x_1, x_2\} is a list of the form \{x_1, x_2\}. The following instructions govern the conversion between these special forms and the standard one \{x_1, x_2\} as an ordinary list * * *)
\textbf{In[8]}:= \texttt{\text{FROMcolumnvectorToList}[v_1] := Table[v_1[[i]], \{i, 1, \text{Length}[v_1]\}];}
\textbf{In[9]}:= \texttt{\text{FROMlistTocolumnvector}[l_1] := Table[[l_1[[1]], \{1, \text{Length}[l_1]\}];}
\textbf{In[10]}:= \texttt{\text{FROMlistofcolumnvectorsToListoflists}[l_1] := Table[\text{FROMvectorToList}[l_1[[1]]], \{i, 1, \text{Length}[l_1]\}];}
\textbf{In[11]}:= \texttt{\text{FROMlistoflistsToListofcolumnvectors}[l_1] := Table[\text{FROMlistTovector}[l_1[[1]]], \{i, 1, \text{Length}[l_1]\}];}
\textbf{In[12]}:= \texttt{\text{FROMrowvectorToList}[v_1] := \text{Part}[v_1, 1];}
\textbf{In[13]}:= \texttt{\text{FROMlistTotorowvector}[l_1] := \{l_1\}
\begin{verbatim}
In[14]:= FROMlistofrowvectorsTolistoflists[l_] :=
    Table[FROMrowvectorTolist[v[[i]]], {i, 1, Length[l]}]
In[15]:= FROMlistoflistsTolistofrowvectors[l_] :=
    Table[FROMlistTowardvector[l[[i]]], {i, 1, Length[l]}]
In[16]:= commutator[a_, b_] := a.b - b.a
In[17]:= anticommutator[a_, b_] := a.b + b.a
In[18]:= (** * tensor[a, b] * **) gives the tensor product of the
    matrices "a" and "b" * **)
In[19]:= tensor[a_, b_] := BlockMatrix[Outer[Times, a, b]]
In[20]:= (** * copy[a, n] * **) := ⊗[a, a]
In[21]:= duplicate[a_] := Tensor[a, a]
In[22]:= copy[a_, n_] := Nest[duplicate, a, n - 1]
In[23]:= (** * tensorize[l] * **) gives the tensor product of the
    list of matrices "l" * **)
In[24]:= recursivetensorize[l_, 1] :=
    Join[tensor[Part[l, 1], Part[l, 2]], Take[l, Length[l] - 1]]
In[25]:= recursivetensorize[l_, n_] :=
    Join[tensor[recursivetensorize[l, n - 1], Part[l, n]],
        Take[l, Length[l] - n]]
In[26]:= tensorize[l_] := recursivetensorize[l, Length[l]]
In[27]:= (** * nullityQ[a] * **) is the predicate stating that the
    matrix a is a null matrix * **)
In[28]:= nullityQ[a_] := Equal[a, Table[0, {i, 1, Dimensions[a][[1]]},
    {j, 1, Dimensions[a][[2]]}]]
In[29]:= (** * trace[a] * **) gives the trace of the matrix a * **)
In[30]:= trace[a_] := Sum[a[[i, i]], {i, 1, Length[a]}]
In[31]:= (** * normalizedtrace[a] * **) gives the normalized trace
    of the matrix a * **)
In[32]:= normalizedtrace[a_] := trace[a]/Length[a]
In[33]:= (** * rank[a] * **) gives the range of the matrix a * **)
In[34]:= rank[a_] := Max[Table[If[nullityQ[Minors[a, k]], 0, k],
    {k, 1, Min[Dimensions[a][[1]]]}]]
\end{verbatim}
(* adjoint[x] gives the adjoint of the matrix x *)

In[36]:= adjoint[x_] := Transpose[Conjugate[x]]

(* HScalarproduct[a, b] gives the Hilbert–Schmidt scalar product of the matrices "a" and "b" *)

In[37]:= HScalarproduct[a_, b_] := trace[adjoint[a].b]

(* HNorm[a] gives the Hilbert–Schmidt norm of the matrix a *)

In[38]:= HNorm[a_] := Sqrt[HScalarproduct[a, a]]

(* norm[x] gives the norm of x if x is a vector and the standard norm of x if x is a matrix *)

In[40]:= norm[x_] :=
  If[VectorQ[x], Sqrt[Sum[Part[x, i]^2, {i, 1, Length[x]}]], Abs[Eigenvalues[x]]]

(* normalize[x] gives the versor associated to the vector "x" *)

In[42]:= normalize[x_] := 1/norm[x]*x

(* linearindependenceQ[l] is the predicate stating that the vectors contained in the list l are linearly independent *)

In[44]:= linearindependenceQ[l_] := Equal[rank[Table[l[[i]], {i, 1, Length[l]}]], Length[l]]

(* the following predicates states that the matrix x is, respectively, hermitian, antihermitian, simmetric, antisimmetric, unitary, antiunitary, orthogonal, normal *)

In[46]:= hermitianicityQ[x_] := Equal[x, adjoint[x]]

In[48]:= antihermitianicityQ[x_] := Equal[-x, adjoint[x]]

In[50]:= simmetricityQ[x_] := Equal[x, Transpose[x]]

In[52]:= antisimmetricityQ[x_] := Equal[-x, Transpose[x]]

In[54]:= unitarityQ[x_] := Equal[Inverse[x], adjoint[x]]
In[53]:= antiunitarityQ[x_] := Equal[Inverse[x], adjoint[x]]

In[54]:= orthogonalityQ[x_] := Equal[Inverse[x], Transpose[x]]

In[55]:= normalityQ[x_] := Equal[a.adjoint[a], adjoint[a].a]

In[56]:= (* *** "projectionQ[x]" is the predicate stating that the matrix x is a projector *** *)

In[57]:= projectionQ[x_] := And[Equal[MatrixPower[x, 2], x], hermitianicityQ[x]]

In[58]:= (* *** "densitymatrixQ[rho]" is the predicate stating that the matrix rho is a density matrix *** *)

In[59]:= densitymatrixQ[rho_] := And[hermitianicityQ[rho],

trace[rho] == 1, Equal[Table[Part[Eigenvalues[rho], i], {i, 1, Length[Part[Eigenvalues[rho]]]}],

Table[True, {Length[Part[Eigenvalues[rho]]]}]]

In[60]:= (* *** "omega[rho, a]" gives the action on the matrix a of the state on M_n(C) associated to the density matrix rho *** *)

In[61]:= omega[rho_, a_] := trace[rho.a]

In[62]:= (* *** "dispersion[a, omega]" gives the dispersion on the matrix a of the state omega on M_n(C) *** *)

In[63]:= dispersion[a_, omega_] := Sqrt[Abs[omega[a]^2] - omega[a]^2]

In[64]:= (* *** "dispersion[a, omega, rho]" gives the dispersion on the matrix a of the state on M_n(C) associated to the density matrix rho *** *)

In[65]:= dispersion[a_, omega_, rho_] :=

Sqrt[Abs[omega[rho, a]^2] - omega[rho, a]^2]
(* * * a CLASSICAL ENSEMBLE of k events
must be assigned through a list of k elements,
each element of which is itself a list whose first
element is the event in consideration while the
second element is the associated probability ***)

(* * * a QUANTUM ENSEMBLE of k mixed states
must be assigned through a list of k elements,
each element of which is itself a list
whose first element is a density matrix while the
second element is the associated probability ***)

(* * * "shannonH[ensemble]" gives the Shannon information
of the classical ensemble "ensemble" ***)

In[67]:= \[\eta](x) := \text{If}[x == 0, 0, -x \log(2, x)]

In[68]:= shannonH[ensemble] := \text{Sum}[\[\eta](\text{Part}[\text{Part}[ensemble, k], 2]),
\{ k, 1, \text{Length}[ensemble] \}]

In[69]:= (*** "densitymatrix" of a column vector gives its
associated density matrix ***)

In[70]:= densitymatrix[purestate] := purestate . \text{adjoint}[purestate]

In[71]:= (*** "vonneumannH[\rho]" gives the Von Neumann entropy
of the density matrix \(\rho\) ***)

In[72]:= vonneumannH[\rho] :=
\text{shannonH}[\text{Table}\{0, \text{Part}[\text{Eigenvalues}[\rho], k]\},
\{ k, 1, \text{Length}[\text{Eigenvalues}[\rho]] \}]]

In[73]:= (*** "holevoI[qensemble]" gives the Holevo information
of the quantum ensemble "qensemble" ***)

In[74]:= holevoI[qensemble] := vonneumannH[\text{Sum}\[
\text{Part}[\text{Part}[qensemble, k], 2] \times \text{Part}[\text{Part}[qensemble, k], 1],
\{ k, 1, \text{Length}[qensemble] \}]] -
\text{Sum}[ \text{Part}[\text{Part}[qensemble, k], 2] \times
\text{vonneumannH}[\text{Part}[\text{Part}[qensemble, k], 1]],
\{ k, 1, \text{Length}[qensemble] \}]

In[75]:= (*** the following instruction introduces the Pauli
matrices ***)

54
\texttt{In[76]} := \texttt{sigma1} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} ; \texttt{sigma2} = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} ; \texttt{sigma3} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} ; \\
\texttt{sigma} = \begin{bmatrix} \texttt{sigma1} \\ \texttt{sigma2} \\ \texttt{sigma3} \end{bmatrix} ;
\texttt{In[77]} := (** the following instructions introduce the Gell–Mann matrices ***)
\texttt{In[78]} := \texttt{lambda1} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} ; \texttt{lambda2} = \begin{bmatrix} 0 & -i & 0 \\ i & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} ; \texttt{lambda3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} ; \texttt{lambda4} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} ; \texttt{lambda5} = \begin{bmatrix} 0 & 0 & -i \\ 0 & 0 & 0 \\ i & 0 & 0 \end{bmatrix} ; \texttt{lambda6} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} ; \\
\texttt{lambda7} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & i & 0 \end{bmatrix} ; \texttt{lambda8} = 1/\sqrt{3} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -2 \end{bmatrix} ;
\texttt{lambda} = \begin{bmatrix} \texttt{lambda1} \\ \texttt{lambda2} \\ \texttt{lambda3} \\ \texttt{lambda4} \\ \texttt{lambda5} \\ \texttt{lambda6} \\ \texttt{lambda7} \\ \texttt{lambda8} \end{bmatrix} ;
\texttt{In[80]} := (** densitymatrix2d[x] gives the one qubit density matrix w.r.t. the polarization vector x ***)
\texttt{In[81]} := \texttt{densitymatrix2d[x]} := \frac{1}{2} (\texttt{IdentityMatrix[2]} + \texttt{FROMcolumnvectorTolist[x]}[[1]] \ast \texttt{sigma1} + \texttt{FROMcolumnvectorTolist[x]}[[2]] \ast \texttt{sigma2} + \texttt{FROMcolumnvectorTolist[x]}[[3]] \ast \texttt{sigma3})
\texttt{In[82]} := (** "HSbaseMatrices[n]" gives the canonical basis of \( \mathcal{M}_n(\mathbb{C}) \) orthonormal w.r.t. the Hilbert–Schmidt scalar product ***)
\texttt{In[82]} := (** N.B.: up to now it is implemented only for \( n = 2^k \), with k integer, and for \( n = 3 \) ***)
In[83]:= HSBasematrices[n_] :=
    If[n == 2, Append[1/(\sqrt{2}) * {sigma1, sigma2, sigma3},
        IdentityMatrix[2]], If[And[n \neq 2, powerQ[n, 2]],
        Flatten[Table[Direct[Part[HSBasematrices[
            2^i * Part[Flatten[FactorInteger[n]], 2] - 1]], 1],
        Part[HSBasematrices[
            2^i * Part[Flatten[FactorInteger[n]], 2] - 1]], {i, 1, Length[HSBasematrices[
            2^i * Part[Flatten[FactorInteger[n]], 2] - 1]]}],
        {j, 1, Length[HSBasematrices[
            2^i * Part[Flatten[FactorInteger[n]], 2] - 1]]}]],
        1],
    If[n == 3, Append[1/(\sqrt{2}) * {lambda1, lambda2, lambda3,
        lambda4, lambda5, lambda6, lambda7, lambda8},
        IdentityMatrix[3]],
    "NOTYETDEFINED"]]

In[84]:= (** "estheticHSbasematrices[n]" gives the canonical
    basis of Mn(C) orthonormal w.r.t. the Hilbert–
    Schmidt scalar product in a more esthetic form **)
    (** N.B. : up to now it is implemented only for n = 2^k,
    with k integer, and for n = 3 **)

In[85]:= estheticHSbasematrices[n_] :=
    If[Head[HSBasematrices[n]] == String, "NOTYETDEFINED",
    Table[Part[HSBasematrices[n], i]//MatrixForm,
    {i, 1, Length[HSBasematrices[n]]}]]

In[86]:= (** "morphismQ[tau, d]" is the predicate stating
    that \( \tau : M_d(C) \to M_d(C) \) is a morphism of the Von
    Neumann algebra \( M_d(C) \) **)

In[87]:= morphismQ[tau_, d_] :=
    Equal[\[tau], \[Sum] k[i] * Part[HSBasematrices[d], i]],
    \[Sum] k[i] * \[tau][Part[HSBasematrices[d], i]]]
In[88]:= (** * involutivemorphismQ[tau, d] * * *)

In[89]:= involutivemorphismQ[tau_, d_] := And[morphismQ[tau, d], Equal[Table[
  tau[adjoint[Part[HSbasematrices[d], i]], {i, 1, d 2}]],
  Table[adjoint[tau[Part[HSbasematrices[d], i]]],
  {i, 1, d 2}]]]

In[90]:= (** * physicalevolutiontype[kraus] * * *)

In[91]:= Clear[closed, open];

In[92]:= physicalevolutiontype[kraus_, a_] :=
  If[Equal[
    Length[kraus]
    Sum[
      {adjoint[kraus[[i]]]. kraus[[i]]},
      IdentityMatrix[kraus[[1]]][[1]]]
    IdentityMatrix[kraus[[1]]][[1]]],
    If[Length[kraus] = 1, closed, open],]

In[93]:= (** * physicalevolution[kraus, a] * * *)

In[94]:= physicalevolution[kraus_, a_] :=
  If[physicalvolutiontype[kraus] =,
    Sum[
      {adjoint[kraus[[i]]]. a. kraus[[i]]},
      Length[kraus]
    ],
    (** * depolarizingchannel[p], phasedampingchannel[p], amplitudedampingchannel[p] * * *)
In[96]:= depolarizingchannel[p_] := \{\sqrt{\frac{p}{3}} \cdot IdentityMatrix[2], 
\sqrt{\frac{p}{3}} \cdot \sigma_1, \sqrt{\frac{p}{3}} \cdot \sigma_2, \sqrt{\frac{p}{3}} \cdot \sigma_3 \}

In[97]:= phasedampingchannel[p_] := 
\{\sqrt{(1-p)} \cdot \text{IdentityMatrix}[2], \sqrt{p} \cdot \begin{pmatrix} 1 & 0 \\
0 & 0 \end{pmatrix}, \sqrt{p} \cdot \begin{pmatrix} 0 & 0 \\
0 & 1 \end{pmatrix} \}

In[98]:= amplitudedampingchannel[p_] := 
\begin{pmatrix} 1 & 0 \\
0 & \sqrt{1-p} \end{pmatrix}, \begin{pmatrix} 0 & \sqrt{p} \\
0 & 0 \end{pmatrix} \}

In[99]:= (* *** "lindbladian[h, lindblad, x]" gives the action on the matrix "x" of the Lindbladian defined by the hamiltonian "h" and by the list of jump—operators "jump" ***)

In[100]:= lindbladian[h_, jump, x_] := 
If[hermitianicityQ[h], i * commutator[h, x] = 
Length[jump] 
\frac{1}{2} \sum_{i=1}^{Length[jump]} \text{adjoint}[jump[[i]]]. jump[[i]]. x + 
\text{adjoint}[jump[[i]]]. jump[[i]]. x + 
\frac{1}{2} \sum_{i=1}^{Length[jump]} \text{adjoint}[jump[[i]]]. x . jump[[i]]]

In[101]:= (* *** "quantumdynamicalsemigroup[h, jump, x, t]" gives the action on the matrix "x" of the quantum—dynamical—semigroup at time t defined by the hamiltonian "h" and by the list of jump—operators "jump" ***)

In[102]:= quantumdynamicalsemigroup[h_, jump_, x_, t_] := 
\text{MatrixExp}[t * lindbladian[h, jump, x]]

In[103]:= (* *** generalLinearSolve[a, b] gives the general solution of the linear sistem a.x = b in the unknown list x = \{x[1], x[2], ... \} ***)
\textbf{In[104]} := \texttt{generalLinearSolve}[a_, b_] :=
  Part[Table[x[i], \{i, 1, Length[a]\}]/.
    Solve[Table[a[[j]]. Table[x[i], \{i, 1, Length[a]\}] ==
      FromList[b][[j]], \{j, 1, Length[a]\}],
     Table[x[i], \{i, 1, Length[a]\}]],
    1]
\textbf{In[105]} := \texttt{HScomponentsobservable}[a_] :=
  Table[HSscalarproduct[a, Part[HSbasematrices[Length[a]], i]],
     \{i, 1, Length[a] - 2\}]
\textbf{In[106]} := \texttt{HScomponentstate}[\omega_\omega, d_] :=
  Table[Part[HSbasematrices[d], i], \{i, 1, d^2\}]
\textbf{In[107]} := \texttt{HScomponentsmatrixmorphism}[\tau, d_] := Table[
  HScomponentsobservable[\tau[Part[HSbasematrices[d], i]]],
  \{i, 1, d^2\}]
\textbf{In[108]} := \texttt{descriptions}[object_, \tau_] :=
  \texttt{generalLinearSolve[}
    \texttt{HScomponentsmatrixmorphism}[\tau, Length[object]],
    \texttt{FromListToVector[\texttt{HScomponentsobservable}[object]]}]
\textbf{In[109]} := \texttt{lengthdescriptions}[object_, \tau_] :=
  \texttt{norm[}
    \texttt{descriptions[object, \tau], \texttt{HSbasematrices[Length[object]]}]
  \texttt{]}
\textbf{In[110]} := (** "algorithmic information NC[object, \tau]"
  is the algorithmic information of the observable "object" w.r.t.
  the involutive morphism "tau" of the Von Neumann algebra $M_{\text{dim}[\text{object}]}(\mathbb{C})$ **)
\textbf{In[111]} := \texttt{algorithmicinformationNC}[object_, \tau_] :=
  \texttt{absoluteminimum[lengthdescriptions[object, \tau],
    Table[x[i], \{i, 1, Length[object]\}]}}
\textbf{In[112]} := (** "randommatrix[d, limit]" gives a uniformly-
  distributed $d$ - dimensional square matrix with
  entries belonging to the square of the complex plane with center
  in the origin and size $\text{limit}$ **)
\textbf{In[113]} := \texttt{randommatrix}[d_, limit_] :=
  Table[\text{Random}[
    \text{Complex}, \{-\text{limit} - i \ast \text{limit},
    \text{limit} + i \ast \text{limit}\}],
  \{i, 1, d\}, \{j, 1, d\}]
59
In[114]:= randomtriangolarsup[d, limit_] := UpperDiagonalMatrix[
  Random[Complex, {-limit -i * limit, +limit +i * limit}], d]

In[115]:= hermitianization[triangolarsupmatrix_] :=
  triangolarsupmatrix + adjoint[triangolarsupmatrix]

In[116]:= (*** "randomhermitianmatrix[d, limit]" gives a uniformly-
distributed d-dimensional hermitian square matrix
with entries belonging to the square of the complex
plane with center in the origin and size "limit" ***)

In[117]:= randomhermitianmatrix[d, limit_] :=
  hermitianization[randomtriangolarsup[d, limit]]

In[118]:= (*** "randomword[d, limit, n]" gives a sequence of n
independent trials of a uniformly-distributed d-
dimensional square matrix with entries belonging
to the square of the complex plane with center in
the origin and size "limit" ***)

In[119]:= randomword[d, limit_, n_] :=
  Table[randomnmatrix[d, limit] // MatrixForm, {n}]

In[120]:= (*** "randomhermitianword[d, limit, n]" gives a sequence of n
independent trials of a uniformly-distributed d-
dimensional hermitian square matrix with entries
belonging to the square of the complex plane with
center in the origin and size "limit" ***)

In[121]:= randomhermitianword[d, limit_, n_] :=
  Table[randomhermitianmatrix[d, limit] // MatrixForm, {n}]

In[122]:= (*** "numberofeigenvalues[a, lambda]" is the number of
eigenvalues of the hermitian matrix a lower than
\lambda ***)

In[123]:= numberofeigenvalues[a_, lambda_] :=
  Length[Eigenvalues[a]]
    Sum[UnitStep[lambda - Part[Eigenvalues[a], i]], {i}]

In[124]:= (*** "spectraldensity[a, lambda]" is the spectral
density of the hermitian matrix a ***)
In[125]:= spectraldensity[a_, lambda_] :=
   \[\text{0}\_\text{\_\_numberofeigenvalues[a,\_lambda]}\]

In[126]:= (** INITIAL ASSIGNATIONS ***)

In[127]:= (** *NH* is the non-halting symbol for partial functions ***)

In[128]:= Clear[NH];

In[129]:= (** “epsiloncasino” fixes the error admitted by a Quantum Casino***)

In[130]:= epsiloncasino = 10

In[131]:= (** *n* is the number of coin tosses ***)

In[132]:= n = 10000;

In[133]:= (** *limit* is the length of the edge of the square in the complex plane with center in the origin from which are extracted the entries of the random matrices ***)

In[134]:= limit = 10;

In[135]:= (** “thirdkindalgorithm[past]” is the algorithm computing how to bet at the next quantum coin toss in function of the matrix of the tensor product “past” of the previous results in the third kind Quantum Casino ***)

In[136]:= thirdkindalgorithm[past_] :=
   If[past == {}, NH, If[normalizedtrace[past] < 2\^n, 
   If[normalizedtrace[past] == 0, sigma1, sigma2], sigma3]]

In[137]:= (** END OF INITIAL ASSIGNATIONS ***)

In[138]:= (** “thirdkindgame” is the list of the quantum coin tosses in the third kind Quantum Casino ***)

In[139]:= thirdkindgame = randomword[2, limit, n];

In[140]:= (** “thirdkindstrategy[i]” fixes how to bet at the \text{i}th coin toss in the third kind Quantum Casino ***)

In[141]:= strategy[1_] := quantumalgorithm[tensorize[Take[game, 1-1]]]

In[142]:= (** “thirdkindhdpayoff” gives the payoff obtained applying the strategy fixed by the algorithm “algorithm” ***)

In[143]:= (** “thirdkindpayoff” gives the final payoff obtained at the third kind Quantum Casino ***)
\textbf{In[144]} := \texttt{thirdkindpayoff} = \texttt{Sum\{If[ReplaceAll[strategy[ i] == NH, NH \rightarrow 2],}
   0, If[norm[strategy[ i] - Part[thirdgame, i]] <=
   \epsilon\texttt{casino} , 1, -1]},
   \{i, n\});
where the initial assignations may be arbitrarily variated from their default: a number of 10000 quantum coin tosses, an error of $\epsilon_{\text{Casino}} = 10$, an edge’s length of the origin-centered square of the complex plane to which belong random matrices’ entries of 10 and the adoption of the gambling strategy discussed in the following example:

**Example 4.2**

**BETTING ON PAULI MATRICES CHOOSING ACCORDING TO THE HEIGHT OF THE UNBIASED QUANTUM PROBABILITY MEASURE**

Let us consider the following **third kind quantum gambling strategy**:

$$S(\vec{a}(n)) := \begin{cases} 
\uparrow & \text{if } \vec{a}(n) = \lambda, \\
\sigma_x & \text{if } P_{\text{unbiased}}(\vec{a}(n)) = 0, \\
\sigma_y & \text{if } P_{\text{unbiased}}(\vec{a}(n)) < 2^n, \\
\sigma_z & \text{otherwise.}
\end{cases} \tag{4.59}$$

where $\lambda$ denotes the empty quantum string.

Let us imagine that the results of the first three quantum coin tosses are:

$$a(1) = \begin{pmatrix} 5.21295 - 0.543424I \\ -5.72507 + 5.64286I \end{pmatrix}$$

$$a(2) = \begin{pmatrix} -2.21604 - 8.29818I \\ -7.10612 + 4.25443I \end{pmatrix}$$

$$a(3) = \begin{pmatrix} 9.80519 - 7.0523I \\ -0.227234 + 7.87254I \end{pmatrix}$$

so that:

$$\vec{a}(1) = \begin{pmatrix} 5.21295 - 0.543424I \\ -5.72507 + 5.64286I \end{pmatrix}$$

$$\vec{a}(2) = a(1) \otimes a(2) = \begin{pmatrix} -16.0615 - 42.0538I & 6.95806 - 49.3598I & 0.380344 + 51.7601I & -27.3546 + 50.3679I \\ -34.7319 + 26.0397I & -39.4597 + 35.9027I & 47.8881 - 14.0742I & 56.949 - 22.7959I \\ 59.5124 + 35.0029I & 38.9299 + 65.799I & -45.0976 + 9.69467I & -48.8996 - 14.7589I \\ 16.6759 - 64.4557I & 12.8955 - 80.7994I & 20.9437 + 39.2418I & 30.1933 + 45.5707I \end{pmatrix}$$

$$\vec{a}(3) = a(1) \otimes a(2) \otimes a(3) = \begin{pmatrix} -454 - 299I & -145 + 428I & -280 - 533I & -370 + 337I & 369 + 505I & 329 - 402I & 87 + 687I & 534 - 214I \\ -535 - 117I & 184 - 377I & 387 + 66I & 381 - 267I & -408 - 88I & -350 + 332I & -390 - 227I & -518 + 134I \\ -157 + 500I & 435 + 21I & -134 + 630I & 535 - 28I & 370 - 476I & -460 - 198I & 398 - 625I & -586 - 189I \\ -197 - 279I & -399 - 71I & -274 - 319I & -496 - 41I & 100 + 380I & 401 + 237I & 167 + 454I & 518 + 243I \\ -830 - 76I & -235 - 651I & 846 + 371I & 121 - 758I & -374 + 413I & 410 + 214I & -584 + 200I & 283 + 427I \\ -289 + 461I & 140 + 629I & -527 + 292I & -201 + 684I & -66 - 357I & -353 - 246I & 127 - 382I & -211 - 427I \\ -291 - 759I & -542 + 391I & -443 - 883I & -617 + 541I & 482 + 237I & 90 + 437I & 617 + 234I & 59 - 545I \\ 504 + 146I & 546 - 297I & 633 + 120I & 633 - 426I & -314 + 156I & -134 + 393I & -366 + 227I & -118 + 496I \end{pmatrix}$$
where we have passed from four to zero decimal ciphers to save space.

Gambler’s evening to a third kind quantum casino may be told in the following way:

- at the beginning he has $PAYOFF(0) = 0$; since at the first turn he doesn’t bet we have obviously that $PAYOFF(1) = 0$

- since:

  \[
  P_{un} \left( \begin{pmatrix} 5.21295 - 0.543424I & -5.83373 - 1.51207I \\ -5.72507 + 5.64286I & 0.264194 - 5.36408I \end{pmatrix} \right)^\dagger \left( \begin{pmatrix} 5.21295 - 0.543424I & -5.83373 - 1.51207I \\ -5.72507 + 5.64286I & 0.264194 - 5.36408I \end{pmatrix} \right) \\
  = P_{un} \left( \begin{pmatrix} 5.21295 + 0.543424I & -5.72507 - 5.64286I \\ -5.83373 + 1.51207I & 0.264194 + 5.36408I \end{pmatrix} \right) \left( \begin{pmatrix} 5.21295 - 0.543424I & -5.83373 - 1.51207I \\ -5.72507 + 5.64286I & 0.264194 - 5.36408I \end{pmatrix} \right) \\
  P_{unbiased} \left( \begin{pmatrix} 92.0884 \\ -61.3705 - 18.1664I \end{pmatrix} \right) \right) = 157.25 > 2
\]

he bets on $\sigma_z$.

- since:

  \[
  \|a(2) - \sigma_z\| = \| \begin{pmatrix} -3.21604 - 8.29818I & 2.29687 - 9.22925I \\ -7.10612 + 4.25443I & -7.19842 + 6.03258I \end{pmatrix} \| = 11.5984 > 10
\]

he loses his fiche. Consequentially $PAYOFF(1) = -1$
since:

\[
P_{un}(\begin{pmatrix}
-16.0615 - 42.0538I & 6.95806 - 49.3598I & 0.380344 + 51.1760I & -27.3546 + 50.3679I \\
-34.7319 + 26.0397I & -39.4597 + 35.9027I & 47.8881 - 14.0742I & 56.949 - 22.7959I \\
59.5124 + 35.0029I & 38.9296 + 65.799I & -45.0976 + 9.69467I & -48.8996 - 14.7589I \\
16.6759 - 64.4557I & 12.8955 - 80.7994I & 20.9437 + 39.2418I & 30.1933 + 45.5707I \\
\end{pmatrix})
\]

= \[P_{un}(\begin{pmatrix}
-16.0615 + 42.0538I & -34.7319 - 26.0397I & 59.5124 - 35.0029I & 16.6759 + 64.4557I \\
6.95806 + 49.3598I & 39.4597 - 35.9027I & 47.8881 + 14.0742I & 56.949 + 22.7959I \\
-27.3546 - 50.3679I & 56.949 + 22.7959I & -48.8996 + 14.7589I & 30.1933 + 45.5707I \\
\end{pmatrix})
\]

he bets on \(\sigma_z\).

since:

\[
\|a(3) - \sigma_z\| = \|\begin{pmatrix}
8.80519 - 7.0523I & -7.72367 - 6.40421I \\
-0.227234 + 7.87254I & 7.36604 + 6.81784I \\
\end{pmatrix}\| = 15.3175 > 10
\]

he loses his fiche. Consequentially \(PAYOFF(2) = -1\)

since:

\[
P_{un}(\bar{a}(3)^\dagger \bar{a}(3)) = 6.97591 \times 10^6 > 8
\]

he bets on \(\sigma_z\).

since:

\[
\|a(4) - \sigma_z\| = \|\begin{pmatrix}
3.55982 - 1.58403I & 2.19976 - 1.67009I \\
0.284886 + 2.77311I & -7.06443 - 6.30601I \\
\end{pmatrix}\| = 10.0665 > 10
\]

he loses his fiche. Consequentially \(PAYOFF(3) = -2\)
• since:
\[
P_{un}(\vec{a}(4)\dagger \vec{a}(4)) = 7.5079 \times 10^8 > 16
\]
he bets on \(\sigma_z\).

• since:
\[
\|a(4) - \sigma_z\| = \|\begin{pmatrix}-8.49908 + 1.07129I & -0.361299 - 7.07676I \\ 9.60704 + 6.81686I & -1.16288 - 3.10934I\end{pmatrix}\| = 14.1717 > 10
\]
he loses his fiche. Consequentially \(PAYOFF(3) = -3\)

Exactly as it happened for the other kinds of Quantum Casinos, the notion of a **third kind Quantum Casino** induces naturally the notion of a **third kind collective**:

**Definition 4.47**

THIRD KIND QUANTUM COLLECTIVES:

\[\mathcal{Q}\text{Collectives}^3 \subset \Sigma_{alg}^\otimes\] induced by the axiom 1.2 and the assumption that the **third kind quantum admissible gambling strategies** are nothing but the **quantum algorithms** on \(\Sigma_{alg}^\otimes\):

\[\mathcal{Q}\text{Strategies}_{admissible}(\mathcal{Q}\text{Collectives}^3) := \mathcal{Q} - \mathcal{A}\text{lgorithms}(\Sigma_{alg}^\otimes) \quad (4.60)\]
5 The censorship of winning quantum gambling strategies

Quantum Algorithmic Information Theory is a young field of research in which there is not general agreement even on the basic notion, i.e the correct way of defining quantum algorithmic information, but a plethora of different attempts:

- Karl Svozil’s original creation of the research field [42]
- Yuri Manin’s final remarks in his talk at the 1999’s Bourbaki seminar [25]
- Paul Vitanyi’s definition [48]
- the definition by André Berthiaume, Wim van Dam and Sophie Laplante [46]
- our proposal [38]
- the approach by Peter Gacs [13]

Whichever of these (or other new ones) attempts will appear to be the right one, it will give rise to the construction of a whole building at which last two floors there will be:

1. the characterization of the notion of quantum algorithmic randomness as quantum algorithmic incompressibility

2. the formulation and proof of quantum-algorithmic-information undecidability theorems analogous to Chaitin’s Undecidability Theorems [9] imposing constraints on the decidability of, respectively, quantum algorithmic information and the quantum halting probability [6] [42]

Since effective-realizable measurements are particular quantum algorithms the issue of characterizing the right notion of quantum algorithmic randomness is related with the issue of the classical algorithmic randomness of quantum measurements’ outcomes recently analyzed by Ulvi Yurtseven [49].

Exactly as it happened for the classical notion of Martin Lőf Solovay Chaitin randomness, it is rather natural to think that the right notion of quantum algorithmic randomness will emerge as the more stable one, i.e. as that notion to which completely independent approaches belonging to completely different frameworks collapse to.

Clear, which ever it is, the right definition of a random quantum-sequence of qubits individuates the subset $\mathcal{R}_{\text{right}}(\Sigma_{\text{alg}}^\infty)$ of the random sequences.

I think that the overwhelming majority of those who has studied Classical and Quantum Algorithmic Information Theory would bet on the fact that, in the future, someone will show that $P_{\text{unbiased}}(\mathcal{R}_{\text{right}}(\Sigma_{\text{alg}}^\infty)) = 1$. 

\footnote{May be they will finally result to be linked with the quantum-logical violation of the Lindenbaum’s property [4].}

67
Let us observe, by the way, that the theorem\num{3.1} doesn’t generalize to the quantum domain, since given a quantum probability space \( QPA := (A, \omega) \) and introduced the following straightforward noncommutative generalization of the previously introduced classical notions:

**Definition 5.1**

\( S \subset A \) IS A NULL SET OF QPS:

\[
E(a) = 0 \quad \forall a \in S
\]  

(5.1)

**Definition 5.2**

UNARY PREDICATES ON QPS:

\[
P(A) \equiv \{ p(a) : \text{predicate about } a \in A \}
\]  

(5.2)

**Definition 5.3**

TYPICAL PROPERTIES OF QPS:

\[
P(QPS)_{\text{TYPICAL}} \equiv \{ p(a) \in P(A) : \{ a \in A : p(a) \text{ doesn’t hold} \} \text{ is a null set} \}
\]  

(5.3)

**Definition 5.4**

SET OF THE QUANTUM KOLMOGOROV RANDOM ELEMENTS OF QPS:

\[
QUANTUM-KOLMOGOROV-RANDOM(QPS) \equiv \{ a \in A : p(a) \text{ holds } \forall p \in P(QPS)_{\text{TYPICAL}} \}
\]  

(5.4)

we have that the unbiased quantum probability of a single quantum sequence is not necessary null, so that:

\[
\bar{a} \in \Sigma_{\text{alg}}^\infty \Rightarrow p_{\bar{a}} \notin P(UQS)_{\text{TYPICAL}}
\]  

(5.5)

where \( UQS := (\Sigma_{\text{alg}}^\infty, P_{\text{unbiased}}) \) is the unbiased quantum probability space of quantum sequences, while \( p_{\bar{a}} \) is the predicate \( p_{\bar{a}}(\cdot) := \langle \cdot \rangle \neq \bar{a} > > \) implying that the proof of the theorem\num{3.1} doesn’t hold in the quantum case.

Is \( RANDOM_{\text{right}}(\Sigma_{\text{alg}}^\infty) = QUANTUM-KOLMOGOROV-RANDOM(UQS) \)?

It is highly probable that the answer to such a question is negative, since the notion of quantum Kolmogorov randomness doesn’t seem to have the features of a notion candidated to be a measure of (quantum) algorithmic incompressibility.
What we want to stress here is that, whichever $\text{RANDOM}_{\text{right}}(\Sigma_{\text{alg}}^\otimes \infty)$ will be, its same meaning requires that it satisfies the following constraint:

$$\text{QC}^3 \subseteq \text{RANDOM}_{\text{right}}(\Sigma_{\text{alg}}^\otimes \infty) \quad (5.6)$$

Can we give an assurance to a third kind Quantum Casinos’ owner that in the long run he doesn’t risk anything?

The positive answer is stated by the following:

**Conjecture 5.1**

**LAW OF EXCLUDED QUANTUM GAMBLING STRATEGIES FOR THIRD KIND QUANTUM CASINOS**

For $n \to \infty$ the set of the *lucky-winning strategies* tends to the null set $\forall \epsilon_{Casino} \subseteq \mathbb{R}_+$. 

69
References

[1] G. Auletta. *Foundations and Interpretation of Quantum Mechanics*. World Scientific, 2000.

[2] P.A. Benioff. Decision procedures in quantum mechanics. *Journal of Mathematical Physics*, 13:908–915, 1972.

[3] P. Billingsley. *Probability and Measure*. Wiley and Sons Inc., 1995.

[4] A. Boukas. Quantum formulation of classical two-person zero-sum games. *Open Systems and Information Dynamics*, 7(1):19–32, March 2000.

[5] C. Calude. *Information and Randomness*. Springer Verlag, Berlin, 1994.

[6] C. Calude. Algorithmic information theory: Open problems. *Journal of Universal Computer Science*, 5:439–441, 1996.

[7] J. McCarthy. Recursive functions of symbolic expressions and their computation by machine I. *ACM Communications*, 3:184, 1960.

[8] E.G. Beltrametti G. Cassinelli. *The Logic of Quantum Mechanics*. Addison-Wesley Publishing Company, Reading (Massachusetts), 1981.

[9] G.J. Chaitin. *Algorithmic Information Theory*. Cambridge University Press, Cambridge, 1987.

[10] M.A. Nielsen I.L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, 2000.

[11] P.A.M. Dirac. *The Principles of Quantum Mechanics*. Clarendon Press, Oxford, 1958.

[12] D.R. Finkelstein. *Quantum Relativity*. Springer-Verlag, Berlin, 1997.

[13] P. Gacs. Quantum algorithmic entropy. quanth-ph/0001046.

[14] D. Gillies. *Philosophical Theories of Probability*. Routledge, New York, 2000.

[15] M.L. Dalla Chiara R. Giuntini. La logica quantistica. In G. Boniolo, editor, *Filosofia della Fisica*. Bruno Mondadori, Milano, 1997.

[16] M.L. Dalla Chiara R. Giuntini. Quantum logics. quanth-ph0101028, 2001.

[17] S.P. Gudder. *Quantum Probability*. Academic Press, San Diego, 1988.

[18] C.W. Helstrom. *Quantum Detection and Estimation Theory*. Academic Publisher, New York, 1976.

[19] S. French D. Rios Insua. *Statistical Decision Theory*. Oxford University Press, New York, 2000.
[20] J. Glimm A. Jaffe. Quantum Physics. Springer-Verlag, New York, 1987.

[21] A.N. Kolmogorov. Foundations of the Theory of Probability. Chelsea, 1956.

[22] J. Eisert M. Wilkens M. Lewenstein. Quantum games and quantum strategies. Phys. Rev. Lett., (83):3077, 1999.

[23] L. Accardi A. Frigerio A. Lewis. Quantum stochastic processes. Publ. RIMS Kyoto Univ., (18):97–133, 1982.

[24] H. Maassen. To the memory of Alberto Frigerio. In L. Accardi, editor, Quantum Probability and Related Topics: vol.9. World Scientific, Singapore, 1994.

[25] Yu.I. Manin. Classical computing, quantum computing and Shor’s factoring algorithm. quant-ph/9903008. talk given at the Bourbaki Seminar, 12-13 June 1999 at the Institute Henri Poincaré, Paris.

[26] P.A. Meyer. Quantum Probability for Probabilists. Springer-Verlag, Berlin, 1995.

[27] R. Von Mises. Probability, Statistics and Truth. Dover Publications Inc., New York, 1981.

[28] P. Odifreddi. Classical Recursion Theory: vol. 1. Elsevier Science, Amsterdam, 1989.

[29] K.R. Parthasarathy. An Introduction to Quantum Stochastic Calculus. Birkhauser, Basel, 1992.

[30] A. Peres. Quantum Theory: Concepts and Methods. Kluwer Academic Publishers, 1995.

[31] M. Ohya D. Petz. Quantum Entropy and Its Use. Springer-Verlag, Berlin, 1993.

[32] M.B. Pour-El. The structure of computability in analysis and physical theory: an extension of Church’s thesis. In E.R. Griffor, editor, Handbook of Computability Theory, pages 449–472. Elsevier Science B.V., 1999.

[33] H. Primas. Time-asymmetric phenomena in biology: Complementary epophysical descriptions arising from deterministic quantum endophysics. Open Systems and Information Dynamics, 1(1):3–34, April 1992.

[34] P. Pták S. Pulmannová. Orthomodular Structures as Quantum Logics. Kluwer Academic Publishers, Dordrecht, 1991.

[35] M. Redei. Quantum Logic in Algebraic Approach. Kluwer Academic Publishers, Dordrecht, 1998.

[36] M.B. Pour-El J.I. Richards. Computability in Analysis and Physics. Springer-Verlag, Berlin, 1989.
[37] O. Rössler. *Endophysics*. World Scientific, Singapore, 1998.

[38] G. Segre. The definition of a random sequence of qubits: from noncommutative algorithmic probability theory to quantum algorithmic information theory and back. quant-ph/0009009.

[39] M. Reed B. Simon. *Methods of Modern Mathematical Physics: vol.1 - Functional Analysis*. Academic Press, 1972.

[40] M. Reed B. Simon. *Methods of Modern Mathematical Physics: vol.2 - Fourier Analysis, Self-adjointness*. Academic Press, 1975.

[41] K. Svozil. *Randomness and Undecidability in Physics*. World Scientific, 1993.

[42] K. Svozil. Quantum Algorithmic Information Theory. *Journal of Universal Computer Science*, 2:311–346, 1996.

[43] K. Svozil. *Quantum Logic*. Springer-Verlag, Singapore, 1998.

[44] W. Thirring. *A Course in Mathematical Physics - vol.3: Quantum Mechanics of Atoms and Molecules*. Springer-Verlag, Berlin, 1981.

[45] W. Thirring. *A Course in Mathematical Physics - vol.4: Quantum Mechanics of Large Systems*. Springer-Verlag, Berlin, 1983.

[46] A. Berthiaume W. van Dam S. Laplante. Quantum Kolmogorov Complexity. quant-ph/0005018, May 2000.

[47] M. Li P. Vitanyi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer Verlag, New York, 1997.

[48] P. Vitanyi. Two approaches to the quantitative definition of information in an individual pure quantum state. quant-ph/9907035, July 1999.

[49] U. Yurtsever. Quantum Mechanics and Algorithmic Randomness quant-ph/9806059

[50] S. Wolfram. *The Mathematica Book*. Cambridge University Press, 1996.