DEMO 2015/

ΠΡΩΤΟΓΕΝΕΣ ΥΠΟΛΟΓΙΣΤΙΚΟ ΙΣΟΔΥΝΑΜΟ ΓΙΑ ΤΗΝ ΑΥΤΟ-ΑΦΙΝΙΚΗ ΙΣΟΔΥΝΑΜΙΑ ΠΟΛΥΩΝΥΜΙΚΩΝ ΑΝΑΠΤΥΓΜΑΤΩΝ ΜΗ ΜΕΤΑΘΕΤΙΚΩΝ ΜΕΤΑΒΛΗΤΩΝ ΚΑΙ ΔΥΝΑΜΙΚΗΣ Liouville – Von Neumann.

Θ. Ε. ΡΑΠΤΗΣ

A COMPUTATIONAL PRIMITIVE FOR THE SELF-AFFINE EQUIVALENCE OF MULTINOMIAL EXPANSIONS OF NON-COMMUTING VARIABLES AND Liouville-Von Neumann DYNAMICS.

T. E. RAPTIS

ΕΚΕΦΕ «ΔΗΜΟΚΡΙΤΟΣ»
Τ. Θ. 60228, 153 10 ΑΓΙΑ ΠΑΡΑΣΚΕΥΗ (Αθήνα) ΕΛΛΑΣ

"DEMOKRITOS"
National Centre of Scientific Research
P.O. BOX. 60228, 153 10 AGIA PARASKEVI (Athens) GREECE

ΠΡΩΤΟΓΕΝΕΣ ΥΠΟΛΟΓΙΣΤΙΚΟ ΙΣΟΔΥΝΑΜΟ ΓΙΑ ΤΗΝ ΑΥΤΟ-ΑΦΙΝΙΚΗ ΙΣΟΔΥΝΑΜΙΑ ΠΟΛΥΩΝΥΜΙΚΩΝ ΑΝΑΠΤΥΓΜΑΤΩΝ ΜΗ ΜΕΤΑΘΕΤΙΚΩΝ ΜΕΤΑΒΛΗΤΩΝ ΚΑΙ ΔΥΝΑΜΙΚΗΣ Liouville – Von Neumann

Θ. Ε. Ράπτης

ΕΚΕΦΕ «ΔΗΜΟΚΡΙΤΟΣ»
Διεύθυνση Τεχνολογικών Εφαρμογών
(rtheo@dat.demokritos.gr)

A COMPUTATIONAL PRIMITIVE FOR THE SELF-AFFINE EQUIVALENCE OF MULTINOMIAL EXPANSIONS OF NON-COMMUTING VARIABLES AND Liouville-Von Neumann DYNAMICS.

T. E. Raptis

NCSR "DEMOKRITOS"
Division of Applied Technologies
(rtheo@dat.demokritos.gr)

Αθήνα, Δεκέμβριος 2015
Athens, December 2015

DEMO 2015/

A Computational Primitive for the Self-Affine Equivalence of Multinomial Expansions of Non-Commuting Variables and Liouville-Von Neumann Dynamics

Theophanes E. Raptis
Division of Applied Technologies
National Center for Science and Research "Demokritos"
Patriarchou Grigoriou & Neapoleos, Athens, Greece, 60228
Email: rtheo@dat.demokritos.gr

Abstract: We present a preliminary investigation of certain generic maps between computational primitives and natural processes with an inherent arithmetic fractal structure that allow finding powerful recursion schemes to be used in the analysis of open quantum systems and their dynamics.

1. Introduction

In a previous report [1] we provided computational evidence for the significance of certain number theoretic properties restricting the combinatoric powerset of all possible measurements in experiments for the detection of Bell inequalities violations. While their appearance is solely associated with the inherent structure of large binary words as a production by a set of binary counters, it is natural to ask whether such a structure has any influence at a deeper level of physical reality.

One of the most important characteristic of such binary sets when seen as matrices also known as factorial designs, is their inherent primitive self-similarity which can be used as a prototype for other processes. In a previous work [2] we gave some more evidence extending the study of this property in any n-ary alphabet and tried to provide a toolbox for the study of its inherent recursive structure with a set of polynomials mappings.

The last decade, evidence for the significance of such constructs came to the light sporadically. In Baris *et al.* [3] we find algorithmic primitives for use in self-assembly while in Hamel [4] work on other group constructs known as difference sets present the interesting property of a dilation operation that can be traced back to the original property of counters. Similar constructs based on binary counters have recently been used to solve the Wang tiles problem [5].

In the present report, we attempt a generalization by showing that the polynomial representations correspond to a type of universal prototype for organized processes that stand for finite computations including non-commuting algebraic operations as a special class. To do so, we first introduce in section ***2*** a construct which serves as an abstract universal dynamical system acting as a global bifurcator and mimicking the way an arbitrary automaton performs computations. Trajectories of such systems always admit a combinatoric powerset structure that is isomorphic with that of n-ary alphabets. In section ***3*** we restrict attention to the particular case where the computation taking place is a non-commuting multiplication as is the case for the matrix representations used in quantum mechanical algebraic manipulations. In section ***4*** we establish the recursive structures associated with the inherent self-similarity of the above and we use the previously introduced block polynomials for the detailed analysis of this structure beyond mere Shannon entropic measures. In section ***5*** we establish the same equivalence in the case of the Kronecker tensor product and we examine the role of the internal recursive structure for the case of density matrix evolutionary dynamics. We conclude with possible applications including the case of simplified Feynman analysis of certain physical processes and the possible underlying algorithmic nature.

2. Polynomial representations of the integers as computational primitives

"*God made the integers; all else is the work of man*."
David Kronecker

The fundamental primitive recursion is given by the successor function $s(n) = n+1, n \in N$. If we also consider the polynomial representation of the integers we may construct a generalized successor function for the symbolic strings or words in an arbitrary alphabet in base b

$$n \to w_n^b : [\sigma_1, ..., \sigma_k]_b : \qquad n = < w_n^b \mid b^n >_k$$

$$\downarrow s(n) \qquad \downarrow gs(n,b)$$

$$n+1 \to w_{n+1}^b : [\sigma_1, ..., \sigma_{k'}]_b : \quad n+1 = < w_n^b \mid b^n >_{k'}$$

In the above, the $gs(n,b)$ can be defined independently of the polynomial representation using a variety of algorithms or equivalent "machine definitions." To do so we have to introduce an additional construct.

For ease of reference, we may denote any combinatoric powerset of the order b^N with a single N x b^N matrix termed the "Lexicon" of order N in base b or just $L(N,b)$. We note in passing that such a set can also be interpreted as the "unfolding" of the set of all paths over a b-nodes tree traversing it from the root node sequentially as in certain search methods used in A. I. and computer science.

We now notice that there are two complementary ways of building an arbitrary Lexicon matrix. If we denote each separate word associated with every integer in the $[0,\ldots,b^N]$ interval as a row in $L(N,b)$ then each such row can be easily reproduced by the "*Rotor*" map which is simply the expression of a permutation of symbols in the form of a dynamical system that rotates the word leftwise or rightwise. Taking the lowest base power as the beginning of the number such a map is given for a leftwise rotation by the equation

$$(1)\ v_{i+1} = \left[\frac{v_i}{b} \right] + (v_i \bmod b) b^{j_{Max}}, \sigma_{i+1} = v_{i+1} \bmod b$$

Repeating the above for a maximal number of symbols will give the corresponding word for any alphabet adding also trailing zeros at the end if it exceeds the maximal base power of a given integer. Additionally, we can provide a second map able to directly reproduce all of the columns of any $L(N,b)$ matrix which can be written as

$$(2)\ L_{ij} = \left[\left(\frac{i}{b^j} \right) \bmod b \right]$$

The above definitions (1) and (2) are not computationally efficient given that they are defined in terms of division and multiplication operations. They serve mostly as comprehensive examples of the inherent structure. In case one needs to examine very large words there are other ways to define sequential abstract automata for mimicking counters or additive machines. Some of them are examined in App. A.

Regarding the inherent self-similarity of (2) one can easily see the self-affine nature of the operation creating every next column from a dilation of the previous one over successive exponential intervals. This fundamental property is inherited by many symbol-wise functions that can be defined on the set of all words and it is of interest in further analyzing the relevant algebraic structures as we will see in section 4. In contrast, (1) causes the appearance of a type of randomization due to the quasiperiodicity inherent to a system of n-ary counters. Thus any such matrix serves also as a prototype of *complementarity*. For ease of reference and clarification in the discussion that will follow at the end, we will term any column-wise implementation of a Lexicon as given by the constructor defined in (2) a *phase-wise* implementation while any row-wise constructor as in (1) will be termed a *path-wise* implementation. To explain the principle we show the two paths via a schematic below

Path-wise

↑ A A A A A A…
A A B B A A…
A B A B A B…

→

Phase-wise

In the last section we discuss a possible association of this fundamental effect inherent in the symbolic structure of the integers with quantum mechanics as well as with the so called *Zuse-Fredkin-Wolfram* (*ZFW*) *Conjecture.*

We now introduce a prototype of an abstract dynamical system for a universal computing language of which the trajectories can be put in a 1-1 correspondence with the Lexicon construct. We first consider a simple *IF…THEN….ELSE* structure of which the algebraic expression can always be given in the two alternative forms

(3) $D_{n+1} = F_1(D_n)^{\chi(D_n)} F_2(D_n)^{1-\chi(D_n)} = \chi(D_n)F_1(D_n) + \overline{\chi}(D_n)F_2(D_n), \chi + \overline{\chi} = 1$

In the above we use D_{n} to denote a data structure which can be represented by a sufficiently large integer and two complementary Boolean variables satisfying $\chi + \overline{\chi} = 1$ that can be associated with appropriate conditions defined as characteristic functions over the same data structure. For Boolean functions χ we can also write a trivial identity like $F^{\chi} = \chi F + \overline{\chi}$. One might add an additional formal axiom for the precedence of left operators so as to avoid double evaluation. In the same way, the exponential form would require a precedence of exponents' axiom which is the opposite from ordinary computing languages.

The above can be generalized for an arbitrary number of computational primitives like those associated with general string rewrite automata like Post Tag systems [6] or Church's λ-Calculus [7]. Let then k primitive functions $\{F_i\}_{i=1}^{k}$ that are sufficient for universal computation acting on an unbounded data structure D conceived as a large integer under an appropriate reading protocol and a program composed of a succession of symbols derived from a set of k indicator functions that at each step satisfy the condition of only one being 1 thus forming a binary shift. Any such program can be written as a k x n array. We may then write a generic dynamical system as a generalization of the well known *SWITCH…CASE* construct of ordinary languages

(4) $D_{n+1} = \prod_{i=1}^{k} F_i(D_n)^{\chi_i^n(D_n)} = \sum_{i=1}^{k} \chi_i^n(D_n) F_i(D_n)$

An additional difficulty stems from the fact that the right choice of the indicator functions may require them to have a *Non-Markovian* character thus depending also in previous steps. At the moment we may omit the details as we are only interested in a formal similarity with the Lexicon construct.

Indeed, if we use a special "*parsing*" map $\hat{p} : w \in N \to \{F_i\}_{i=1}^{k}$ we can identify each separate word of an $L(n,k)$ matrix with a possible computation via the correspondence

$$w_n^b : [\sigma_1, ..., \sigma_n] \xrightarrow{\hat{p}} F_{\sigma_1}(...(F_{\sigma_n})...)(D_0), \sigma_i \in [0, k-1]$$

Thus a Lexicon matrix when read phase-wise, represents the set of all histories of possible computations or programs even those that would not be necessarily meaningful or even valid for a particular purpose or task. Then any specific program represents a specific choice of an ever increasing word among all the possible histories contained in a succession of *L* matrices when these are read path-wise.

We also notice that (4) can be rewritten as a simple inner product if we introduce a binary vector $\mathbf{p}_n = [\chi_1^n,...,\chi_k^n]$ and another one representing different possible actions over the same data structure as $\mathbf{F}(D_n)$ in which case a program is equivalent to successive unit vectors spanning a k-dimensional hypercube omitting a normalization factor. The history of any computation can then be interpreted as a succession of projections from the same fundamental primitive vector ***F*** on a discrete k-dimensional lattice. One can also conceive of a cloud of such primitive vectors from the action of ***F*** on all integers that could represent a possible data structure independently of their time ordering. In such a lattice all computations preexist and the sole role of the program is a reordering in order to select a particular history. In such a space computational time itself is the primary production.

It should be stressed that Non-Markovianity is a characteristic more general than non-commutativity for sequences of functional compositions even in the absence of any specific program. By this we mean that when the same primitive will be found to correspond to a block of same symbols it cannot be "reused" independently as it is affected by different sets of previous histories. This stronger restriction can be relaxed in the case of matrix representations of non-commuting algebras that we study in the next section.

3. Generic expansion formulas for non-commuting variables

The study of functions of non-commuting operators is a vast field that goes back at the foundations of quantum mechanics. In a classic paper by Kumari [8] generic functions of two operators of the form $f(\hat{A}+\hat{B})$ are studied. Simpler cases of binomial expansions are also been studied in Potter[9] and Schutzenberger [10] in the restricted case of *q-commuting* variables. Characteristic functions for infinite sequences of non-commutign operators were researched by Popescu [11] In standard quantum mechanics textbooks the well-known Zassenhaus formula [12] is also widely used.

Generalizations in cases of multinomial expansions are tedious due to the generic operator ordering issue as is the case with the time-order operator in Feynman integrals [13]. We will now use the symbolic tools of section **2** to unify the study of similar expansions and also show possible advantages of our approach.

We first associate each term in an expansion with a "*parsing*" map $\hat{p}: N \to S$ from the integers to the set of all algebraic expressions which we denoted with *S*. This map associates each symbol of an n-ary string in its combinatorics powerset with an algebraic expression as

$$w_n^b : [\sigma_1,...,\sigma_k] \xrightarrow{\hat{p}} \hat{X}_1 \circ ... \circ \hat{X}_k$$

Doing so makes the set of all such expressions recursively enumerable. Moreover, when we are dealing with simple operations like multiplication even in the non-commuting case, the inherent self-similarity of any Lexicon matrix eases the analysis of the contents and may lead to certain simplifications that will be discussed in the next section.

We may take as an instantiation of the above the simple binomial expansion

$$(5)\ \left(\hat{A}_0 + \hat{A}_1\right)^n = \sum_{i=0}^{2^n-1} \prod_{j=1}^{n} \hat{A}_{\sigma(i,j)}, \sigma(i,j) \in L(n,2)$$

In the same spirit as in (4) we could have as well written (5) in the somewhat obfuscated equivalent form

$$(6)\ \left(\hat{A}_0 + \hat{A}_1\right)^n = \sum_{i=0}^{2^n-1} \prod_{j=1}^{n} \hat{A}_0^{\sigma(i,j)} \hat{A}_1^{1-\sigma(i,j)}$$

Given the previous analysis we may now interpret the operators or matrices $A_{0,1}$ as the computational primitives and the particular sequence of symbols $\sigma(i,j)$ as the "program". It is notable that (6) brings about a particular symmetry of any binary Lexicon matrix. In fact, for any length *n*, there is always a subdivision in two intervals $S_1 = [0,...,2^{n-1}-1]$, $S_2 = [2^{n-1},...,2^n-1]$ such that for every word $w \in S_1$ there is a bitwise-NOT complement $\overline{w} \in S_2$. The same can be written analytically as $\overline{n} = 2^n - n - 1, \forall n \in S_1, \overline{n} \in S_2$. If we consider the products as the trajectories defined by an abstract dynamical system we see that there are always two complementary sets of them hence the study of half of them sufficiently characterizes the whole set. Other symmetries like palindromic or self-reflective words of same length *n* can also be isolated inside each step of the hierarchy of Lexicon matrices but we will not go into details here.

Given this structure, if one considers the overall action of the total object in (5) or (6) to a vector space then one can also assume a superposition principle in which case one may isolate 2^n separate actions of the resulting matrices and interpret the result as an average over a self-similar set. The degree of self-similarity induced by the parsing map will depend on the particulars of the matrix elements but the important factor is the recursive structure of all interference terms in the trajectory space. This suggests isolating two *Principal Trajectories* that span the set of all interference terms as

$$(7)\ \begin{cases} \mathbf{t} \leftarrow \left\{ \hat{A}_0, \hat{A}_0^2, \ldots \right\} \\ \bar{\mathbf{t}} \leftarrow \left\{ \hat{A}_1, \hat{A}_1^2, \ldots \right\} \end{cases}$$

In contrast with the case of abstract functional compositions in section ***2*** all the intermediate components of the two principal trajectories are now reusable. That means that any other interference term can be constructed by the components of these two thus allowing also to turn indices to exponents and reducing the length of the resulting intermediate trajectories. This leads directly to the analysis of the next section dealing with the exact properties and the recursive structure of all interference terms.

We can generalize the above for arbitrary multinomial expansions using the extended formula

$$(8)\ \left(\sum_{k=1}^{K} \hat{A}_k \right)^n = \sum_{i=0}^{K^n-1} \prod_{j=1}^{n} \hat{A}_{\sigma(i,j)}, \sigma(i,j) \in L(n,K)$$

Let us also note that one can also interpret such formulas as projections of a single entity which can be given via a *K* x *K* diagonal matrix representation as

$$\mathbf{K} = \begin{bmatrix} \hat{A}_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \hat{A}_K \end{bmatrix}$$

Then we can rewrite (8) in the symbolic form

$$(9) \quad \left(\sum_{k=1}^{K} \hat{A}_k \right)^n = \sum_{i=0}^{K^n - 1} \prod_{j=1}^{n} \left(\mathbf{e}(\sigma_{ij}) \cdot \mathbf{K} \right), \sigma_{ij} \in L(n, K)$$

In (9), $\boldsymbol{e}(\sigma)$ is another binary matrix with only a single nonzero entry block $\boldsymbol{I}$ at the position corresponding to the particular symbol of a K-ary alphabet present in the particular word that corresponds to the i index. Thus we can exchange the previous parsing map with a new one that sends every word to a set of projectors defined by the set of K binary shifts across the diagonal. The set of all interference terms inside the summand is again spanned by the K principal trajectories defined as

$$(10) \quad \begin{array}{l} \mathbf{t}_i \leftarrow \left(\{ \Sigma(i)\mathbf{K}, (\Sigma(i)\mathbf{K}), \ldots \} \right)(\mathbf{V}) \cong \left\{ \left(\{ \hat{A}_i, \hat{A}_i^2, \ldots \} \right)(\mathbf{v}_0) \right\}_{j=1}^{K} \\ i \in [1, \ldots, n], \mathbf{v}_0 \in V \subset \mathbf{C}^n, \mathbf{V} \in V^K \end{array}$$

In (10) we also expand any original N-dimensional vector space V in another KN-dimensional one with K copies of all possible actions. Hence, (9) actually says that every trajectory represents a simple reordering of a preexisting entity that is spanned by the trajectories in (10) and which can be directly precomputed. This simplifies (9) in the form

$$(11) \quad \left(\sum_{k=1}^{K} \hat{A}_k \right)^n \cdot \mathbf{v}_0 = \sum_{i=0}^{K^n - 1} \left(\prod_{j=1}^{n} \mathbf{e}(\sigma_{ij}) \cdot \mathbf{t}_j \right)$$

In (10) we interpret the action of the subsequent $\boldsymbol{e}$ matrices as operators from the left. Having isolated the essential building blocks it remains to find out how does the recursive structure inherent in the σ set gets inherited by the particular type of mapping of (11).

4. Self-similarity, Block-analysis and recursive relations

To ease the reader we recapitulate our strategy till now.

a. All entities from a combinatorial powerset represented by $L(n,b)$ in its natural lexicographic ordering together with an appropriate parsing map over some algebra of compositions form an effectively single dimensional representation or an *unfolding* which can be interpreted as a discrete *fibration* over the product space $L(n,b) \times F$ where F is a set of some functors with an associated algebra.
b. For a set of b elements and a composition law where Markovianity holds we may isolate b distinct fibers as the principal trajectories each corresponding to a unique symbolic dynamical system for every separate operation. The rest of the resulting trajectories from an unfolding over $L(n,b)$ shall be called interference terms and they can be put into a direct correspondence with the loci of common points from a set of helical periodic orbits on n-dimensional hypertorii with exponentially increasing periods in base b.

To examine the particular symmetries from the basic self affine map that stands as a generic constructor of all possible $L(n,b)$ matrices as given in (2), we start from the set of symbol-wise morphisms. While one meets sporadically certain cases where such morphisms have been known to be endowed with the structure of arithmetic fractals, it is obvious that we are in need of a more systematic approach for classification purposes.

To give a rather well known example, all bitwise Boolean functions like *AND* and *XOR* are known to produce similar arithmetic fractals. While these are defined as two variable functions we can always perform a type of endomorphism on a higher Lexicon matrix $L(2n, 2)$ where the product $L(n,2)$ x $L(n,2)$ is projected as a one dimensional graph albeit somewhat distorted. Eventually, every such many dimensional morphism defined on an integer lattice can always be reduced on its associated one dimensional form via the correspondence

(12) $$f(\sigma_1,...,\sigma_n) \cong f(\sigma_1 + \sigma_2 b + ... + \sigma_n b^{n-1})$$

A study of similar morphisms associated with the paper folding problem and certain well known sequences can be found in Morton and Mourant [14]. Generalizations appear in Lapidus[15] and more recently in Rastegar [16]. An attempt to apply arithmetic operations into continuous fractals can be found in Aerts [17] based on previous work by Czahor [18]. We are here more interested in understanding the complete structure of a combinatoric set like $\{\mathbf{e}(\sigma_{ij})\cdot\mathbf{t}_j\}$of (11) when lexicographically ordered in the sense of (12).

Understanding the kind of mixing of the principal trajectories in (11) can be seen with a simple example from a binary expansion of 3rd order in tabulated form as shown in table 1 below.

1	2	3	2	2	3	2	1
A_0^3	$A_1A_0^2$	$A_0A_1A_0$	$A_1^2A_0$	$A_0^2A_1$	$A_1A_0A_1$	$A_0A_1^2$	A_1^3
000	100	010	110	001	101	011	111

Table 1

Indeed, we see that the lengths of the expressions in the top row appear to vary again in an ordered and possibly recursive way. We will name this quantity the *Blcok Dimension* and written as $d_B(n)$ hereafter for convenient reference. Analysis of this sequence structure follows after some simple paradigmatic cases that allow the necessary generalization required.

In order to elucidate the underlying order we have to introduce some auxiliary tools that correspond to solvable recursive relations. As a primitive example we may examine the structure of a simple number theoretic function known as the digit-sum of which a detailed study with references can be found in Borwein [19]. Utilising the self affine nature of the row constructor of every Lexicon in phase-wise order, it is trivial to verify that for all exponential intervals 2^n we have the following recursion reproducing all values of the binary digit-sum function over every such interval

(13) $$\begin{cases} S_D(i+1) \leftarrow \{S_D(i), S_D(i)+1\} \\ S_D(0) \leftarrow \{0\} \end{cases}$$

In (13) the algebraic addition of 1 is performed element-wise to every member of the previous set. The above recursion can be solved with the aid of a 2-period pulse function $h(n, p_1, p_2)$ defined as an indefinite repetition of the binary word $\{0\}^{p_1}\{1\}^{p_2}$ given by

(14) $$S_D(\nu) = \sum_{k=1}^{n} h(\nu, 2^k, 2^k), \nu \in [0,...,2^n]$$

With a bit more effort one can verify that for arbitrary $L(n,b)$ matrices the associated digit-sum functions can be found through a recursion of the form

$$(15)\quad \begin{cases} S_D(i+1) \leftarrow \{S_D(i)+w_1 b^i,\ldots,S_D(i)+w_b b^i\} \\ w_k \leftarrow [\sigma_1,\ldots,\sigma_{b^{n-1}}], \sigma_j = \delta_{k,j}, k=1,\ldots,b \\ S_D(0) \leftarrow \{0,\ldots,b-1\} \end{cases}$$

In (15), each word w is a binary shift of length b moving cyclically and applied element-wise as an array to each previous subset. Sequences like (13) and (15) serve also as prototypes for more general functional compositions. Indeed, one may write an arbitrary recursion of the form

$$(16)\quad S(n+1) \leftarrow \{S(n), F_1(S(n)),\ldots,F_k(S(n))\}$$

In (16) we find a similar structure with that of the abstract dynamical system defined in (3) so that it may also serve as a prototype of a universal language given k computational primitives. The unique characteristic of (16) is that in this version the sequence of operations obtained includes all the images with exponents given again by the k-ary digit-sum function as in (13). Hence, a set of k precomputed trajectories of each F suffices for all productions of (16) as a sampling of the principal trajectories.

Another such primitive which leads back to the subject of block analysis is the so called leading-zeros function giving the first block of zeros of every word for any associated integer. This can be given in the binary case after restricting $L(n,2)$ to its even subset via the recursion

$$(17)\quad \begin{cases} S_D(2(\kappa+1)) \leftarrow \{S_D(2\kappa), S_D(2\kappa)+w\} \\ S_D(2) \leftarrow \{1\}, w \leftarrow [0,0,\ldots,1] \end{cases}$$

In (17), w is again a simple binary shift of length equal to that of the previous interval ($2^{\kappa-1}$). It is again possible to solve (17) as

$$(18)\quad S_D(2\kappa) = \sum_{k=1}^{n} h(\kappa,2^k,1), \kappa \in [0,\ldots,2^{n-1}]$$

In all of the above, self affinity of the original construction is reflected in the final productions. It is natural to ask how this principle generalizes for the various different block structures inside each and every word in an exponential interval.

We notice that the integer sequence resulting from the expansion in Table 1 is self-reflective and can indeed be reproduced by the recursion

$$(19)\quad \begin{cases} d_B(n+1) \leftarrow \{S, \hat{\mathbf{m}}(S) + 1(n < n_{\max}/2)\} \\ \\ d_B(1) \leftarrow \{0,1\} + 1 \end{cases}$$

In (19) $\hat{\mathbf{m}}$ stands for a "mirror" reflection or left-right flip operator while the logical operator reflects the internal mirror symmetry of every binary Lexicon which reproduces the other half without addition. Let us note that for certain trivial cases the structure of the principal trajectories is also trivial as for instance in the case of the two operators coming from the set of Pauli matrices σ_i, σ_j. One then gets the 2-period sequence $v, 0, v, 0...$ with v being i or j respectively. If we were to consider the full SU(2) basis though we would get the 4-period $v, -I, -v, I...$ Even a periodic set when randomly sampled results again in a random sequence but this is not the case here. Hence, the final set of block dimensions as well as the overall block structure will exhibit strong regularities albeit of an increased complexity in the limit of large expansion orders. The same regularities will be induced in the final mixing set of trajectories $\{\mathbf{e}(\sigma_{ij}) \cdot \mathbf{t}_j\}$ in (11).

In order to explore the mixing properties in the full combinatoric powerset and its possible fractality we must now invent a tool that will allow us to have a kind of microscopy over the complete block structure of any and all such sequences that stand for the path-wise read interference terms. To this aim we introduce the *Block Polynomials* with a maximal order given by the associated block dimension for each sequence. It is easier to present this in the binary case first. By an exact sequential count of every separate block of same symbols of 0s or 1s in a row we can associate any binary word in a 1-1 mapping with a monic polynomial of the form

$$(20)\quad n \in [0, ..., 2^N - 1] \rightarrow w_n : [\sigma_1, ..., \sigma_N] \rightarrow (\pm 1, c_1, ..., c_{d_B(n)}), c_i \in [1, ..., N]$$

In (20), the leading sign coefficient represents the starting bit being 0 or 1 respectively. One can start with an alternating polynomial of order $d_B(n)$ given a change of encoding like $\sigma_i \to 2\sigma_i - 1 \in \{-1,1\}$ and then count subsequent blocks with their sign before extracting the sign bit in front. The mapping given by (20) or by their equivalent roots in the complex plane can be proven to be a 1-1 correspondence due to its connection with the problem of restricted integer partitions [20] in the binary case. Indeed, as every combination of blocks must be realized their summand must always satisfy

$$(21) \qquad \sum_{k=1}^{d_B(n)} c_k = n$$

Recent findings by Ono [21] also justify the fact that the set of integer partitions hide a self-similar structure. A related Fast Integer Partition (FIP) generator will be part of the forthcoming software package.

The true advantage of this representation is that it does not lose the details that remain hidden by simple Shannon entropic measures. For instance, simple use of the enumeration formula for binomial coefficients like $2^{\mathrm{N}} = \sum_{\kappa=0}^{\mathrm{N}} \binom{\mathrm{N}}{k}$ shows that entropic measures can only take into account the separate factorial groups and not the exact block structure of each of them hence they lack in vision of the inside complexity and possible regularity of large words. The polynomial mapping is a more refined view of complexity thus making block analysis an alternative complexity estimator.

We may now introduce a more general type of a complex polynomial mapping for arbitrary k-ary alphabets using the *k* roots of unity $\omega_n = \exp[2\pi\mathbf{i}(n/b)], n \in [0,...,b-1]$ as representatives of each symbol for every distinct block which then takes the form

$$(22) \qquad n \in [0,...,b^N - 1] \to w_n : [\sigma_1,...,\sigma_N] \to (c_1\omega_i,...,c_{d_B(n)}\omega_j), c_i \in [1,...,Nb]$$

Turning back to the problem of the operator sequences, we see that the above helps in greatly simplifying their evaluation in case of any cyclic group. For any such group of order say, *p*, the amplitude c_i of every coefficient is identical with the image exponent in any trsjectory and can be taken in *mod* p arithmetic. This fact endows the above mappings with a natural ring structure and allows to also make certain connections with the vast subject of algebraic varieties [22] as well as with arithmetic and Diophantine geometries, a theme that goes beyond our scope here. Regarding algorithmic evaluation of the above polynomial mappings, we notice that careful use of the recursions shown can be utilized to provide all the block coefficients without actually producing large strings. A suitable MATLAB toolbox is in preparation for all the above which will be announced elsewhere.

5. The case of Liouville-Von Neumann dynamics

In the modern algebraic formulation of quantum mechanics [23] time evolution is given in terms of the wavefunction density matrix through the definition of the Von Neumann superoperator $\hat{L} = \lfloor \hat{H}, \rho \rfloor$ with the dynamics been given by the so called, *Liouville-Von Neumann equation* in the form $\partial\rho / \partial t = -(\mathbf{i}/\hbar)(\hat{L})(\rho)$. A formal solution is then given as

$$\rho(t) = \exp\left(-\mathbf{i}\hat{H}t/\hbar\right)\rho(t_0)\exp\left(\mathbf{i}\hat{H}t/\hbar\right)$$

Supplementary conditions for ρ to be a proper density matrix are often given as $Tr(\rho) = 1, \rho \geq 0$. For n identical copies of entangled particles, one usually considers a Hamiltonian as a tensor product of independent Hamiltonians for every subsystem as $\otimes_{i=1}^{n} \hat{H}_i$ while the density matrix can always be decomposed in an appropriate linear $SU(n)$ basis with elements Σ_{i} with $Tr(\Sigma_i) = 0$ as $\rho = (1/n)\left(\mathbf{I} + \sum_{i=1}^{n^2-1} \lambda_i \Sigma_i\right)$ so that the dynamics is reduced in the study of the trajectories given by $\exp\left(-\mathbf{i}\hat{H}t/\hbar\right)\Sigma_i \exp\left(\mathbf{i}\hat{H}t/\hbar\right)$. The basis elements are themselves also given as tensor products generalizing the ordinary $SU(2)$ basis of Pauli matrices. In this final form, we recognize the crucial factor in any possible structure and regularities of the eigenvalue spectrum as well as the eigenvectors of the collective Hamiltonians.

At this point the simplest observation is self-explanatory. The mere fact that tensor products were made to guarantee the connection of every previous element with every other also guarantees their self affine nature when reinterpreted in the language of the combinatoric Lexicon hierarchies. For this to become obvious one needs to introduce a new encoding guaranteeing the existence of an isomorphic map between the elements of any such square matrix and an $L(n^2, b)$ matrix of unfolded elements. We first try an abstract example with arbitrary 2 x 2 Hermitian matrices. These can always be produced as the logarithms of unitary matrices which in their most general form can be parametrized by two complex numbers in $\boldsymbol{C}^2$ as

$$\hat{U} = \begin{pmatrix} z_1 & z_2 \\ -\bar{z}_2 & \bar{z}_1 \end{pmatrix} : \hat{H} \leftarrow \left(\log \hat{U} + (\log \hat{U})^T\right)/2$$

Taking the Kronecker product in such a parametrization will produce a matrix form which is necessarily self-similar. In fact, we can now introduce a mapping between any such Kronecker product and a multinomial expansion of the previous section with the use of a quaternary alphabet ($b = 4$) so as to distinguish between all elements of the resulting Hermitian matrix. Use of a parsing map as before will again reproduce the same combinatory mixture of the principal trajectories $\{\hat{H}^n_{ij}\}$ with exactly the same products as those contained in the original Kronecker product.

The sole additional ingredient for such a complete map is the introduction of a special reordering in reading any such Kronecker matrix which can be given as a discrete version of a so called, *Lebesque Z-order* curve. This is shown in the below schematic as a sequence of steps for a 2nd order product containing only products of two terms

$$\begin{pmatrix} 1 & 2 & 5 & 6 \\ 3 & 4 & 7 & 8 \\ 9 & 10 & 13 & 14 \\ 11 & 12 & 15 & 16 \end{pmatrix}$$

At the 3rd step one needs to refine the same numbering scheme for each new sub-block in which case one is led to a recursive scheme which exhibits a primitive form of self-similarity when unfolded in a one dimensional graph. To do so, we use a scheme where any pair of indices (*i,j*) in its equivalent unfolded form becomes $i + 2^n j$. At the end of any row, (mod *n*) arithmetics and a double increment suffices for cyclically covering the whole matrix. One can construct this way a Look-Up Table (LUT) as a one-dimensional graph of precomputed addresses for any such operation. Such an operation is of course algorithmically invertible hence there always exist a 1-1 discrete fractal map that turns any product of Hamiltonians into its equivalent multinomial $(H_{00}+...+H_{11})^n$ and the associated Lexicon. In this scheme, the combinatoric set of mixed trajectories has a direct correspondence with simple commuting complex products

$$\{\mathbf{e}(\sigma_{ij})\cdot \mathbf{t}_j\} \to \prod_{\substack{k=1 \\ i,j\in\{0,1\}}}^{n} H_{ij}$$

If certain initial configurations for some Hamiltonians exist which inherit the natural self-similarity and the recursive structure introduced in the previous two sections then it would be important for applications to know under what conditions this property perseveres during Liouvillian evolution. To elucidate this we need to enlarge our set of mappings in such a way that not only the original static version of a Kronecker product but also the complete evolution equation will take a new unfolded form so as to be amenable to the same treatment as in the previous sections. For this reason, we will have to introduce a pair of interrelated operators acting on matrices performing a reordering according to a certain "Multiplexing" scheme. Such operators actually stand for counting algorithms instead of analytical objects and their realization will be part of a forthcoming software package.

We then define the $\hat{V},\hat{M}$ operators and their inverses as follows. For any matrix $H = H_1 \otimes H_2 \otimes ... \otimes H_n$ we define *P* as a reordering that faithfully sends all elements of *H* to the elements of an associated vector $\vec{h}$ and its inverse as $H = V^{-1}(\vec{h})$. This is realized by the "z-curve" algorithm of the previous paragraph. For any such operation, we define its complementary operation denoted by *M* as the algorithm who maps a standard commutator of *n* x *n* matrices to a standard matrix-vector product at the cost of introducing a new $n^2 \times n^2$ sparse matrix which will be called the *Interaction Kernel*. The combined action of both maps is then defined by the following equivalence on any matrix-matrix product as

$$(23) \quad V(A \cdot B) = M_L(A)V(B) = M_R(B)V(A)$$

It is trivial to check that *V* is a linear map satisfying $V(A+B) \to \vec{a}+\vec{b}$. It is then possible to rewrite the Von Neumann superoperator in the two dual representations

$$(24)\quad \begin{cases} \hat{V}\left(\left[\hat{H},\rho\right]\right) = K_L(H)\vec{\rho} = K_R(\rho)\vec{h} \\ K_L(H) = M_R(H) - M_L(H) \\ K_R(\rho) = M_L(\rho) - M_R(\rho) \end{cases}$$

Using the simplification provided by (24) direct integration of the Liouville-Von Neumann equation gives

$$(25)\quad \dot{\vec{\rho}} = (-\mathbf{i}/\hbar)K_L(H)\vec{\rho} \rightarrow \vec{\rho} = \exp\left((-\mathbf{i}/\hbar)K_L(H)t\right)\rho(t_0)$$

It is also possible to inspect the inner workings of this evolution by virtue of the *SU(n)* decomposition of ρ in the original superoperator which becomes an average over the elementary commutators $\left[\hat{H},\Sigma_i\right]$. Given that we can also extract a 1st order ODE system for the λ_i coefficients we once again arrive at the expressions

$$(26)\quad \begin{cases} \lambda(t) = \exp\left(-\mathbf{i}\Omega/\hbar\right)\lambda_0 \\ \Omega = \vec{\Sigma}_i^{-1} K_L(\hat{H})\vec{\Sigma}_i \end{cases}$$

The frequencies Ω are now given by the square forms contracting the elements of the total interaction kernel after transforming the basis elements accordingly thus simplifying the exploration of any regularities present during evolution. For an initial Hamiltonian produced by a Kronecker product we may use the dual expression $\vec{h} = V(\hat{H})$ to inspect the order in the elements of the interaction kernel via a pair of left and right homeomorphisms given as

$$(27)\quad \begin{cases} V(K_L(H)) = \vec{h}_R - \vec{h}_L \\ \vec{h}_R = (V \circ M_R \circ V^{-1})(\vec{h}) \\ \vec{h}_L = (V \circ M_L \circ V^{-1})(\vec{h}) \end{cases}$$

The full study of such mappings for certain important cases will be reported after completion of the necessary software and will be reported in a subsequent publication.

6. Discussion and Conclusions

"He who thinks freely, thinks well"
Rigas Pheraeos

From the previous argumentation the author hopes to have provided enough evidence for the significance of the existence of certain regular and recursive structures that permeate many important fields from the mathematical toolbox used in physics as well as in computational theory. The existence of such a unifying tool allows exploring certain complex processes from a different perspective. In fact, the existence of regularities in all combinatoric powersets at least for discrete dynamics implies a type of reduction from dynamics to statics similar to that perceived via the so called D' Alembert principle of virtual works in classical mechanics. It is in essence a Platonic reduction of time to a simple reordering of preexisting abstract elements. We may separate the important lessons of the whole excursion by separating them in two main categories, one dealing directly with the possibility of an additional algorithmic structure underlying natural laws and a more technical one dealing with possible benefits for computational sciences.

Regarding the significance of the plan laid out in section ***4***, for general open quantum systems, we may here notice a previous extraordinary finding by Diederik Aerts [24]-[27] on the inseparability of quantum entities where these are not considered to be parts of the standard Euclidean space used in our models and equations. Regarding the particular property of complementarity inherent with the distinction between two types of construction for the same hierarchy of combinatoric objects we find that there is a peculiar analogy with problems like the double-slit experiment for which reason we distinguished between a phase-wise and a path-wise traversing of the same objects. For the case of so called, delayed choice experiments in particular we find an extreme similarity with the case of what is known as the "*Lazy Evaluation*" technique used in functional languages where actual calculation of any results is delayed for the end of the computation. Assuming that any photon path especially in the Feynman sense of all possible paths is actually in a symbolic state before any measurement as a "representative" of an underlying primitive entity and its "collapse" as forcing the extraction of a certain numeric result, it is not surprising that it appears as if "travelling backwards in time" when measured before arrival at a slit. This is perhaps better understood as a change from a "phase-wise" mode of existence to a "path-wise" one. By this we actually mean that all such expressions used till now may be just bad inferences based on the misleading understanding offered by such representations of "particles" vs. "waves".

This appealing equivalence between generic classes of universal computations and the particulars of certain types of quantum dynamics brings about other previously acclaimed similarities of the universe with some kind of computing machinery that became known in some circles as the Zuse-Fredkin-Wolfram (ZFW) Conjecture named after the main authors that mostly promote this idea in some deeper technical level [28] - [30]. At the moment we can only argue that even if such an analogy has any merit it cannot be in the simplicial sense of some cellular or other automata first tried by Fredkin and next explored by Wolfram. This is the reason for mentioning the difficulties with the distinction between a symbolic and a numeric substrate which is able to incorporate also non-classical phenomena which cannot be present in a classical model of computation. Further development of the above program may reveal otherwise but we find it appropriate to add here that if a simple device as an interferometer is capable of altering the state of any underlying algorithmic substrate from what we termed a symbolic, abstract to a specific, numerically concrete state of existence, it is then possible that there could be more refined versions of machines that perhaps by forcing some type of so called, Hyper-Turing computation could serve as a kind of "operating system" thus allowing direct commands to be send to the said substrate, at least locally.

At this point a fundamental research question remains to be answered. With respect to what has been described as a geometric representation of all possible computations on an integer lattice in section 2. In particular, it is of great importance the finding of any particular set of primitive functors such that the cloud of precomputed elements over all possible data in the said lattice is self-similar or generally recursive in some manner. It would then be a separate discussion whether or not the particular universe we inhabit may already have made use of such an advantage.

We may also provide a hint which is definitely important as it is connected with the deeper question of an underlying universal order behind all computations as well as discrete physical processes. An important part of the theory of varieties concerns the so called *"Toric"* varieties [31] and their connections with Polytope geometry where certain integer periodicities arise in a k-dimensional lattice. Such a connection strongly depends on the appropriate choice of either computational primitives or operator group structures.

Philosophical ramifications aside, in practical terms the research program proposed may show several benefits in deepening our understanding of such issues as emergent complexity and computational complexity. Moreover, in the case one could find ways to endow the space of solutions of certain difficult problems with self – similar recursive structure this could make possible to at least infer the existence of solutions in a vastly larger space by simply inferring the exact nature of the recursive law underlying the whole space without actually performing each separate computation ("path-wise"). This might represent an important step in many optimization problems thus giving the above program a value on its own at least for purely technical reasons. Further research will require the completion of the specialized software package under preparation which will be reported in a subsequent publication.

References

[1] Raptis, T. E., "*Polynomial Representations of the Integers and Primordial Self-Similarity in the Hierarchy of Universal Lexicons*" DEMO 2011/1 Report. DOI:

[2] Raptis, T. E. "*Arithmetic Fractal Structures and Number Theoretic Properties in the Measurement Protocols of the CHSH and Leggett-Garg Inequalities.*" DEMO 2015/1 Report, DOI: 10.13140/RG.2.1.1736.8408.

[3] Baris, R. D. et al, "Two Computational Primitives for Algorithmic Self-Assembly: Copyning and Counting", Nano-Letters 5(12), (2005) 2586 – 2592.

[4] Hamel, M. et al., "Arithmetic Structure in Sparse Difference Sets", J. Num. Theory, 130(7), (2010), 1581–1589, arXiv:1004.2723.

[5] Abel Z. *et al., "Shape Replication through Self-Assembly and RNase Enzymes"*, Proc. 21st Annual Symposium on Discrete Algorithms, (2010).

[6] Post, E.: "*Formal reductions of the combinatorial decision problem*", American Journal of Mathematics, 65 (2), (1943) 197–215.

[7] A. Church, "*A set of postulates for the foundation of logic*", Annals of Mathematics, Series 2, 33 (1932) 346–366.

[8] Kumari, K., "*Expansion of a Function of Noncommuting Operators*", J. Math. Phys. **6** (1965).

[9] H. S. A. Potter, "*On the latent roots of quasi-commutative matrices.*", this MONTHLY 57 (1950) 321-322.

[10] M.P. Schutzenberger, "*Une interpretation de certaines solutions de l'´equation fonctionnelle: F(x+y) =F(x)F(y).*" C. R. Acad. Sci. Paris 236, (1953), 352–353.

[11] Popescu, G. "*Characteristic Functions for Infinite Sequences of Noncommuting Operators*", J. Operator Th. **22** (1989) 51 – 57.

[12] M. Suzuki, "*On the convergence of exponential operators: the Zassenhaus formula, BCH formula and systematic approximants.*", Comm. Math. Phys. 57 (1977), 193-200.

[13] Smirnov, V. A., "*Analytic Tools for Feynman Integrals*" (2012) Springer.

[14] Morton, P., Mourant, W. J., "*Paper Folding, Digit Patterns and Groups of Arithmetic Fractals.*", Proc. London Math. Soc. (1988) **59**(3) (1989) 253 – 293.

[15] Lapidus, M. L., Frankenhuijsen, M., "*Fractal Geometry and Applications: a Jubille of Benoit Mandelbrot.*", Annual Meeting of AMS 2002 Proc. (2004).

[16] Rastegar, A., arXiv:math/0404498

[17] Aerts, D. *et al.*, "*Crystallization of space: Space-time fractals from fractal arithmetic.*"

[18] Czahor, M., "*Relativity of Arithmetic as a fundamental symmetry of physics.*" J. Q. Studies: Math. Found. (2015) 1-11.

[19] Borwein, J. M.*;* Borwein, P. B. *(1992),* "*Strange series and high precision fraud.*", Am. Math. Monthly **99**(7) 622-640.

[20] Kelleher, J., "*Encoding Partitions as Ascending Compositions.*", PhD Thesis, Univ. College Cork (2006), arXiv: 0909.2331.

[21] A. Folsom *et al.*, **"***ℓ-Adic properties of the partition function.*", Adv. Math**.** **229**(3) (2012) 1586 – 1609.

[22] Robin, H. "*Algebraic Geometry*", (1977) Springer-Verlag.

[23] Heinosaari, T., Ziman, M. "*The Mathematical Language of Quantum Theory.*", (2012) Cambridge Univ. Press.

[24] Aerts, D., "*Description of many physical entities without the paradoxes encountered in quantum mechanics.",* Foundations of Physics, **12**, (1982) 1131-1170.

[25] Aerts, D., "*The missing elements of reality in the description of quantum mechanics of the EPR paradox situation.*", Helvetica Physica Acta, **57** (1984) 421-428.

[26] Aerts, D. *et al.*, "*Probing the Structure of Quantum Mechanics: Nonlinearity, Nonlocality, Probability and Axiomatics"*, Singapore: World Sci. 20 – 46.

[27] Aerts, D. "*An attempt to imagine parts of the reality of the micro-world.*", J. Mizerski, A. Posiewnik, J. Pykacz and M. Zukowski (Eds.), *Problems in Quantum Physics*. (1990) 3 - 25.

[28] Zuse, K., "*Rechnender Raum.*", (1969) Braunschweig: Friedrich Vieweg & "*Calculating Space*", MIT Technical Translation AZT-70-164-GEMIT, MIT(Project MAC),

[29] Whitworth, B., " *Simulating space and time*", Prespacetime Journal , **1**(2), (2010) 218-243.

[30] Wolfram, S., "*A New Kind of Science*", (2002) Champaign, IL: Wolfram Media, Inc.

[31] Cox, A. D., Little, J. B., Schenck, H., "*Toric Varieties*" (2011) AMS ed.