THE GENETIC CODE: A CASE OF RESIDUE NUMBER SYSTEM (RNS)

Joshua Apigagua Akanbasiam[1,2], Kwame Osei Boateng[2] and Matthew Glover Addo[3]

[1,2]Department of Electrical/Electronics Engineering, Dr. Hilla Limann Technical University, Wa, Ghana
[2]Department of Computer Engineering, Kwame Nkrumah University of Science and Technology, Ghana
[3]Department of Theoretical and Applied Biology, Kwame Nkrumah University of Science and Technology, Ghana

ABSTRACT

Cracking the code of life marked an interesting and significant beginning of life science. The genetic code maps the 20 amino acids to all 64 possible arrangements of the four (4)nitrogenous bases. Since its deciphering, various designs in varied fields have been proposed. This work presents a Residue Number System (RNS)-based genetic code using the concept of number tree. In conformity with Gamow's postulation, (4^3= 4 × 4 × 4 = 64) we present a three moduli set RNS model of the genetic code. The design sorts the codes into their "family box" or "four-codon box". The application of RNS in bioinformatics is not new - notably its application to Smith-Waterman Algorithm. RNS has been used to successfully generate the genetic code and this advances the cause of representing the genetic code in a quaternary number system.

KEYWORDS

Residue Number System, RNS-genetic Code, Genetic Code, Number Tree, Amino-acids.

1. INTRODUCTION

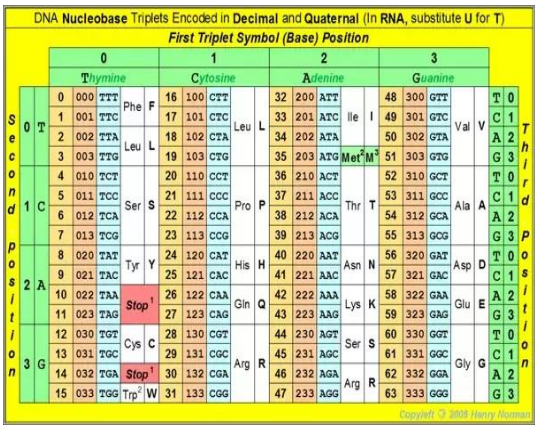
The genetic code drives its relevance from the saying that "A code without translation makes sense but translation without a code does not make sense". Francis Crick proposed the existence of a "genetic code," which is the set of all codons that specify the 20 amino acids.[1]. The concept of "pseudo-wild" led him to provide the first experimental evidence in support that the codons of the genetic code are organized in triplets [2]. George Gamow [1954] offered the first mathematical architecture of the genetic code (4 × 4 × 4 = 4^3 = 64)[3]. The first of the 64 triplet codons to be deciphered was UUU – Phenylalanine and by 1966 [4] all the 64-codons of the genetic code were cracked [4][5]. Cracking the code of life paved the way for modern molecular biological research [3]. The most prevalent representation of the canonical genetic code, found in[6][7], is a three-dimensional matrix, where each dimension corresponds to one of the three positions in the triplet code [4]. Some researchers have related various concepts of some fields of study to understanding the features of the genetic code [8][9][10][11][12]. The attempts to offer mathematical or computational approach to the genetic code have been the assignment of either binary [13]or quaternary number system[9]. Binary number system is used because is the conventional number system used in modern digital computations and quaternary system because the number of known nitrogenous bases are four (4).The genetic code representations have been extensively researched into to take advantage of digital signal processing techniques. An innovative approach that is mathematically structured to suite implementation of computational algorithms will therefore aid modern molecular biology. It is also evident that the use of computers and mathematics is revolutionizing the field of molecular biology. And therefore an enhanced digital representation of the genetic code has the potential of transforming molecular biological processes and analysis. This approach uses Residue Number System (RNS) to generate the code of life using the concept of number trees. The triplets of bases in an amino acid in the genetic code are represented with each digit of a three (3) moduli set residue number system. In agreement with Gamow for the use of $4 \times 4 \times 4 = 4^3 = 64$, we thus used three moduli sets, m1, m2 and m3, requiring the truncation of higher moduli sets values (>= 4) in the three moduli set selected. Just as codon families have unique characteristics like being hydrophobic, polar or non-polar, so do the decimal values generated in each block of code. They have unique sequential numbering based on the moduli sets chosen and the order of these moduli sets. The nature of the tree of residue digits can enhance further analysis on the genetic codes. The capability of RNS to generate the genetic code signals a buoyant relationship between molecular biology and RNS. The application of RNS in the field of bioinformatics is not new [14]–[16]. This design can improve speed, offer smaller digital foot print, and lower cost of some bioinformatics or molecular biological designs, modeling and analysis. Simply, it has the potential of taking advantage of the benefits RNS offers contemporary digital applications [17].

1.1. Nucleic Acids (DNA And RNA)

The information required to produce life is all contained in the molecule DNA (deoxyribonucleic acid), which is likely the most significant molecule for life. [18, p. 255]. Nucleotide polymers make up DNA and RNA (ribonucleic acid), with the former controlling the creation of RNA through a process known as transcription [19, pp. 3–4][20]. DNA differs from RNA in that, deoxyribose sugars are used instead of ribose sugar therefore[21, p. 75 and 83];the hydroxyl (OH) group in ribose sugar is replaced with hydrogen (H) in deoxyribose sugar at position 2’ of the ribose sugar [22].DNA stores information as codes made up of four chemical bases, which are grouped as purines and pyrimidines. Adenine (A) and Guanine (G) are the two pyrimidine bases of DNA, while Cytosine (C) and Thymine are the two purine bases (T). The RNA has similar purines but differ in their pyrimidines – Thymine (T) is replaced with Uracil (U). The basic tenet of genetics or gene expression is that bases are the essential building blocks of amino acids and subsequently proteins. The well-known Chargaff's rule is used to pair the bases; A pairs with T and C pairs with G. The amino acids in a tabular form constitute the genetic code [23]. The genetic code can be represented in DNA and RNA form since they have nearly similar bases except for the replacement of thymine in DNA with uracil in RNA.

1.2. The Genetic Code

The genetic code is a set of rules that determine how a nucleotide sequence is converted into the amino acid sequence of proteins [24][25].The genetic code is made of some characteristics that define its nature. It is made of triplets of codons that is non-overlapping, commaless and non-ambiguous. It is also degenerate, has polarity and is universal [26]. There are $4^3 = 4 \times 4 \times 4 = 64$ codon combinations from four bases and each codon codes for one type of amino acid. The blocks on the table are separated by the same bases in the first two positions. The genetic code was first thought of as universal, but the mitochondria genome has different genetic codes. George Gamow [1954] unsuccessfully represented the genetic code in his “diamond code”[5][27] – Figure 3. The most prevalent representation of the canonical genetic code has a three-dimensional matrix with each dimension corresponding to one of the three locations in the triplet code.[28] – Figure 4.



DNA **Nucleobase** Triplets Encoded in **Decimal** and **Quaternal** (In **RNA**, substitute **U** for **T**)

Second Position		0 Thymine				1 Cytosine				2 Adenine				3 Guanine				Third Position	
0	T	0	000	TTT	Phe F	16	100	CTT	Leu L	32	200	ATT	Ile I	48	300	GTT	Val V	T	0
		1	001	TTC		17	101	CTC		33	201	ATC		49	301	GTC		C	1
		2	002	TTA	Leu L	18	102	CTA		34	202	ATA		50	302	GTA		A	2
		3	003	TTG		19	103	CTG		35	203	ATG	Met[2] M[3]	51	303	GTG		G	3
1	C	4	010	TCT	Ser S	20	110	CCT	Pro P	36	210	ACT	Thr T	52	310	GCT	Ala A	T	0
		5	011	TCC		21	111	CCC		37	211	ACC		53	311	GCC		C	1
		6	012	TCA		22	112	CCA		38	212	ACA		54	312	GCA		A	2
		7	013	TCG		23	113	CCG		39	213	ACG		55	313	GCG		G	3
2	A	8	020	TAT	Tyr Y	24	120	CAT	His H	40	220	AAT	Asn N	56	320	GAT	Asp D	T	0
		9	021	TAC		25	121	CAC		41	221	AAC		57	321	GAC		C	1
		10	022	TAA	Stop[1]	26	122	CAA	Gln Q	42	222	AAA	Lys K	58	322	GAA	Glu E	A	2
		11	023	TAG		27	123	CAG		43	223	AAG		59	323	GAG		G	3
3	G	12	030	TGT	Cys C	28	130	CGT	Arg R	44	230	AGT	Ser S	60	330	GGT	Gly G	T	0
		13	031	TGC		29	131	CGC		45	231	AGC		61	331	GGC		C	1
		14	032	TGA	Stop[1]	30	132	CGA		46	232	AGA	Arg R	62	332	GGA		A	2
		15	033	TGG	Trp[2] W	31	133	CGG		47	233	AGG		63	333	GGG		G	3

Copyleft © 2006 Henry Norman

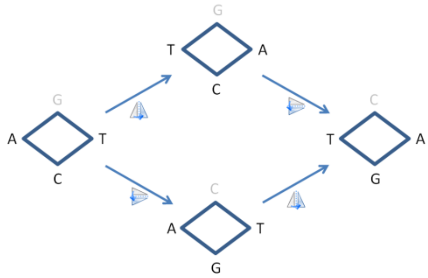


Figure 3: Quaternary Genetic Code [9]Figure 4: George Gamow's diamond code [3]

This work considers the standard or otherwise canonical genetic code [7]. Mathematically, the4^3 explains the number of bases as four (4) and the number three (3), as that which constitute a codon and its evaluation, as the number of codons in the entire genetic code. The expanded form of4×4×4 further expresses how these codes are arranged in tabular form, where each of the 4's represents the first, second and third bases in the codon table. Table 1 presents how an RNS-genetic code is developed from a three (3) moduli set RNS system.

Table 1: An RNS – Amino acid Representation

	4	5	7	Amino Acid		4	5	7	Amino Acid		4	5	7	Amino Acid
0	0	0	0	TTT	47	3	2	5		94	2	4	3	
1	1	1	1	CCC	48	0	3	6		95	3	0	4	
2	2	2	2	AAA	49	1	4	0		96	0	1	5	
3	3	3	3	GGG	50	2	0	1	ATC	97	1	2	6	
4	0	4	4		51	3	1	2	GCA	98	2	3	0	AGT
5	1	0	5		52	0	2	3	TAG	99	3	4	1	
6	2	1	6		53	1	3	4		100	0	0	2	TTA
7	3	2	0	GAT	54	2	4	5		101	1	1	3	CCG
8	0	3	1	TGC	55	3	0	6		102	2	2	4	
9	1	4	2		56	0	1	0	TCT	103	3	3	5	
10	2	0	3	ATG	57	1	2	1	CAC	104	0	4	6	
11	3	1	4		58	2	3	2	AGA	105	1	0	0	CTT
12	0	2	5		59	3	4	3		106	2	1	1	ACC
13	1	3	6		60	0	0	4		107	3	2	2	GCC
14	2	4	0		61	1	1	5		108	0	3	3	TGG
15	3	0	1	GTC	62	2	2	6		109	1	4	4	
16	0	1	2	TCA	63	3	3	0	GGT	110	2	0	5	
17	1	2	3	CAG	64	0	4	1		111	3	1	6	
18	2	3	4		65	1	0	2	CTA	112	0	2	0	TAT
19	3	4	5		66	2	1	3	ACG	113	1	3	1	CGC
20	0	0	6		67	3	2	4		114	2	4	2	
21	1	1	0	CCT	68	0	3	5		115	3	0	3	GTG
22	2	2	1	AAC	69	1	4	6		116	0	1	4	
23	3	3	2	GGA	70	2	0	0	ATT	117	1	2	5	
24	0	4	3		71	3	1	1	GCC	118	2	3	6	
25	1	0	4		72	0	2	2	TAA	119	3	4	0	
26	2	1	5		73	1	3	3	CGG	120	0	0	1	TTC
27	3	2	6		74	2	4	4		121	1	1	2	CCA
28	0	3	0	TGT	75	3	0	5		122	2	2	3	AAG
29	1	4	1		76	0	1	6		123	3	3	4	
30	2	0	2	ATA	77	1	2	0	CAT	124	0	4	5	
31	3	1	3	GCG	78	2	3	1	AGC	125	1	0	6	
32	0	2	4		79	3	4	2		126	2	1	0	ACT
33	1	3	5		80	0	0	3	TTG	127	3	2	1	GAC
34	2	4	6		81	1	1	4		128	0	3	2	TGA
35	3	0	0	GTT	82	2	2	5		129	1	4	3	
36	0	1	1	TCC	83	3	3	6		130	2	0	4	
37	1	2	2	CAA	84	0	4	0		131	3	1	5	
38	2	3	3	AGG	85	1	0	1	CTC	132	0	2	6	
39	3	4	4		86	2	1	2	ACA	133	1	3	0	CGT
40	0	0	5		87	3	2	3	GAG	134	2	4	1	
41	1	1	6		88	0	3	4		135	3	0	2	GTA
42	2	2	0	AAT	89	1	4	5		136	0	1	3	TCG
43	3	3	1	GGC	90	2	0	6		137	1	2	4	
44	0	4	2		91	3	1	0	GCT	138	2	3	5	
45	1	0	3	CTG	92	0	2	1	TAC	139	3	4	6	
46	2	1	4		93	1	3	2	CGA	140				

2. NUMBER SYSTEM

A method for representing numbers in the architecture of a computer system is known as a number system[29]. The performance of computer arithmetic is frequently constrained by the carry chain of the traditional binary number system[14]. Digital electronics is achieved as voltage transitions and these are understood as either HIGH or LOW – the main idea behind binary numbers as the ideal notation for representing digital electronic circuits. Some of the major constraints in the application of binary number system in digital electronics had been sign representation and the carrying and borrowing in digital arithmetic. As the demand for high speed data processing and transmission and also smaller digital footprints of digital designs in various fields of digital computing has become necessary, Residue Number System (RNS) has been re-invented[30]. This is as a result of the benefit it provides in overcoming some difficulties that prevent binary data representation from being a viable option for some fast computing applications.

2.1. RNS and RRNS

The greatest common divisor of the set of k pairwise relative co-prime positive integers that make up the residue number system (RNS) is 1, $(mi, mj) = 1$ with $i \neq j$ and $m_1, m_2 \ldots, m_{k-1}, m_k$, called moduli[17]. The interval [0, M], also known as the valid range or dynamic range, which determines the practical computational range of the number system, is represented by the product (m_1 x m_2 x ……. m_{k-1} x m_k) that is,

$$M = \prod_{i=1}^{K} mi \quad (1)$$

There is a distinct K-tuple representation for each integer in the residue class Zm which is given by

$X \rightarrow (r_1, r_2, \ldots\ldots\ldots\ldots, r_k)$ Where $ri \equiv |X|mi$ or $(X \bmod mi)$

RNS can represent signed numbers; for all scenarios of even and odd number dynamic range, the range of representable integers is divided into two equal intervals.

$\left[\frac{-(M-1)}{2}, \frac{M-1}{2}\right]$ If M is odd, and

$\left[\frac{-M}{2}, \left(\frac{M}{2}\right) - 1\right]$ If M is even.

2.2. Moduli Set

The choice of moduli sets plays a significant role in designing an RNS system. The form and quantity of the chosen moduli play a significant role on the speed and complexity of internal RNS arithmetic circuits as well as the forward and reverse conversion[17, p. 8]. The moduli set selection also affect the dynamic range, the speed, as well as the VLSI implementation of RNS systems [31]. The powers of two related moduli can simplify the required arithmetic operations and this result in efficient hardware implementation of the RNS systems by using usual binary hardware [32]. For RNS implementations, a variety of moduli sets, including the two, three, four, etc., have been proposed. The three moduli sets have received some considerable research attention and the moduli set; $2^n - 1, 2^n, 2^n + 1$ is the most projected because of its simplicity and balance moduli [33].The RNS-Genetic code proposed requires a three moduli set, for each of the bases in a triplet of codon, and each modulus must be greater than or equal to four (4), to adequately represent the four (4) bases in each position of the codon. The dynamic range must be greater than or equal to 64 to make up for all codons in the codon table. Customarily if the first two conditions are met then the third condition will suffice. For many important operations, including forward and reverse hardware converters, some unique sets of moduli result in hardware simplifications and a decrease in latency. The approach considered for this work would work with any set of relatively prime moduli sets considered hence a more open approach to the selection of moduli sets for hardware implementation.

2.3. Converters: Forward and Reverse Converters

The forward conversion is considered as the algorithms that transform any conventional weighted number system to RNS and reverse conversion is any algorithm which transforms RNS to the weighted number system [30, pp. 49&213]. The residue-to-binary conversion is a crucial step for any successful RNS application. The traditional technique of this conversion is based on the Chinese Remainder Theorem (CRT). The reverse conversion from residue to the weighted number has some performance bottlenecks whereas the forward conversion process is relatively easier[34]. In recent years, the Chinese Remainder Theorem (CRT) or Mixed-Radix Conversion has served as the foundation for reverse conversion algorithms. In contrast to MRC, which is a sequential operation that frequently necessitates the use of numerous lookup tables, CRT frequently entails the use of a huge modulo adder, which is the product of all moduli. The new CRT I and II proposed, have simpler algorithms, improved speed and reduced hardware complexity. The research work on converters is actively on-going and the proposed RNS-Genetic code would comply with any improved converter design.

3. THE NUMBER TREE (FOREST)

A tree is a discrete structure that represents hierarchical relationship between individual elements or nodes [35]. Typically, the root is the "starting" node; it has no parents, and every edge either directly or indirectly emanates from it. The predecessor node is the parent and the successor nodes are the children. Nodes without children are leaves[36] and Figure 5, shows an illustration of the nature of number trees.

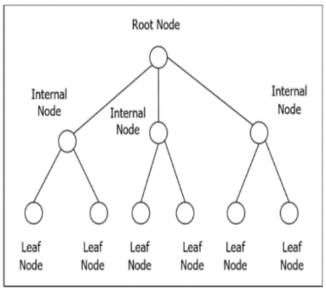


Figure 5: Nature of tree of numbers[35]

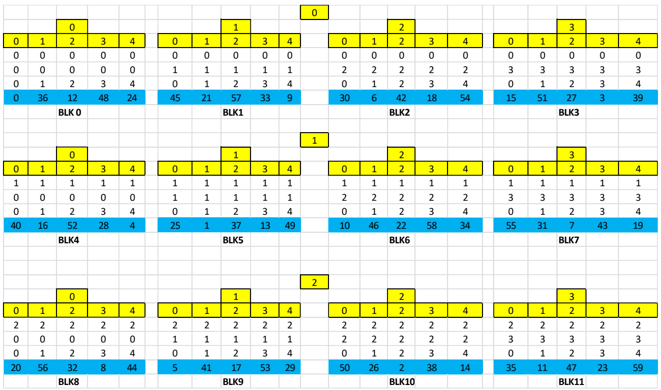


Figure 6:Tree (forest) of Residue digits

The binary tree, in which each node has two (mod 2) offspring or less, is a specific case of a number tree. The concept of number trees has been used in developing algorithms like the binary search tree, itis also employed in[37]to knit a Fibonacci tree. This work has constructed an RNS-Tree (forest)–Figure 6, and used it to generate the genetic code. The RNS based tree (forest) model has the number of nodes at each level determined by the moduli set at that level – m1, m2 or m3. The number of "levels" is determined by the number of moduli sets chosen, thus 2 moduli sets give two levels, three moduli sets, three levels and so on. In an RNS based genetic code, a three moduli set is chosen since we require three levels where each level represents a particular base in a triplet of codons.

4. METHODOLOGY

When considering the design of the RNS-genetic code, we proceeded with the rudimentary conventions that there are only 4 nucleotides and 20 encoded amino acids and that each codon is a triplet of nucleotides [7]. RNS can be designed as a tree and combination of trees constitute a forest of residue numbers. The classical RNS can be generated in tabular form, given the moduli set chosen, as shown in Table 1. The concept here is to design a forest of residue numbers from the generated residue numbers with appropriate roots, nodes (internal nodes) and leaves. This design uniquely categorizes the residue numbers into blocks of data. These blocks display certain unique characteristics that have been exploited to generate the genetic code.If the moduli set of RNS are chosen such that there are at least three (3) moduli sets - m1, m2 and m3 and their dynamic range greater than or equal to 64 thus m1 x m2 x m3≥ 64 and each of m1, m2 and m3 greater than or equal to four (4). A tree (forest) of residue numbers can be constructed such that with appropriate truncation and digit-baseassignment, the genetic code or amino acid table can be constructed from the block (tree) of residue numbers. The genetic code representationally has four (4) bases for each row, four (4) bases for each column and four (4) other bases within each block - thus a 4 x 4 x 4 or 4^3 = 64 codons. The moduli sets are chosen such that each codon in the genetic code is uniquely represented by each of the three moduli sets; m1, m2 and m3. The moduli set m1 is the root nodes of each tree (forest) and represent the first letter of each codon, the second moduli set, m2, is the child node and represents the second letter of each codon, the third moduli set, m3, is the leaves nodes of each tree and represents the third base of each codon. Thus, a collection of root, child and leaf; or a concatenation of each digit of the three chosen moduli sets constitutes a codon of the genetic code.

The number of roots of the tree (forest) is determined by the first moduli set m1, the number of nodes is represented by the second moduli set, m2, and the third moduli set m3, which is the leaf node concludes the three (3) moduli set representation of the genetic code. Since there are four (4) bases, each moduli set is truncated to four (4)residue digits, thus 0, 1, 2 and 3. The RNS-Genetic code table is completed with residue digits assigned appropriate nitrogenous bases. Thus, Thymine (T) is assigned residue digit zero (0), Cytosine (C) is assigned residue digit one (1), Adenine (A) is assigned residue digit two (2) and Guanine (G) is consigned residue digit three (3).In the case of an RNA-Genetic code Thymine (T) is replaced with Uracil (U) and assigned residue digit zero (0). The algorithm for generation of the RNS-Genetic code can be seen in section 4. 1.The genetic code has some properties and therefore any model that seeks to generate it must reflect these properties. Every codon block (quartet) has the same initial two nucleotides, and they are more crucial for the specific codon-anticodon identification. This is illustrated in figure-6, where the RNS sorted into a tree, orders the first two digits of each codon block (quartet) as the same digits. The third digit changes incrementally within each block – and this explains the wobble hypothesis[38].

4.1. Design Flow – Algorithm

An algorithm is a procedure or formula for solving a problem or systematic method for producing a specified result. The five (5) key properties of algorithms are answered in the quest to generate the RNS-genetic code. The inputs are the RNS moduli sets selected (defined) and the output is the genetic code produced as results. The definite nature of the algorithm is specified by the sequence of operations defined for turning the inputs into output. It is deemed effective since it’s doable and finally stops offering the output set out to be achieved, RNS-Genetic Code. This is very relevant since it offers a codified, platform independent for arriving at our solution. Below is the algorithm required to generate the RNS-Genetic Code.

RNS Genetic Code Table(m_1,m_2,m_3)

Input: Three relativey prime numbers m1, m2, m3
Output: RNS Genetic code Reference Table

```
M ← m1 * m2 * m3
N ← 64    // number of codons
Array GeneticCode[N, 6]
count ← 0
for i ← 0 to M-1
    if ((x < 4) && (y < 4) && (z < 4))
        GeneticCode[count, 0] ← x
        GeneticCode[count, 1] ← y
        GeneticCode[count, 2] ← z
        GeneticCode[count, 3] ← AminoAcid(x,y,z)
        GeneticCode[count, 4] ← CodonName(x,y,z)
        GeneticCode[count, 5] ← i
        count ← count + 1
    x ← x + 1
    y ← y + 1
    z ← z + 1
    if (x = m1)
        x ←0
    if (y = m2)
        y ←0
    if (z = m3)
        z ←0
return GeneticCode
```

5. Results and Discussion of Model

This is a novel approach of representing the Genetic code. The results of this findings, RNS-Genetic code with moduli sets 4, 5, and 7 are shown below. This work further draws the relationship of the findings to the canonical genetic code. There are two nucleic acids (DNA and RNA) and generally, the genetic code can be generated in either form. As discussed earlier the two have nearly similar nitrogenous bases except for the replacement of Thymine (T) with Uracil for the case of RNA. The RNS-Genetic code can therefore be generated as a table of DNA or a table of RNA as shown in the simulation results of Figure – 7and 8 (RNS-DNA Genetic code) and (RNS-RNA Genetic code) respectively. The choice of moduli selection plays a major role in the selection of RNS for any digital implementation. In the design, three moduli set RNS is considered since we required three residue digit concatenations for each of the 64 degenerate amino acids on the genetic code table. The design accepts any three (3) relatively prime moduli set implementation of the RNS-genetic code. The use of the number tree (forest) , suggests that the genetic code would not be affected by any change in the three (3) relatively prime moduli sets selected for implementation. As such the design can take advantage of any effectively implemented three (3) moduli sets – since the research on moduli selection and converters is on-going. With the use of residue number trees, each of the digits that make up a particular block of code will always be in the same block irrespective of the moduli sets selected. The major difference noticed in any change of moduli sets is a change in decimal values of each of the concatenated residue digits that make up each amino acid. In Figure 7, the moduli set chosen are 4, 5 and 7 and as such the decimal digits for the first block of codons are seen as 0, 120, 100 and 80. When these moduli sets were reversed, thus 7, 5 and 4, the decimal digits for the first block of codons are 0, 105, 70 and 15 – Figure 8.

RNS-Genetic Code

TTT	Phe	0	TCT	Ser	56	TAT	Tyr	112	TGT	Cys
TTC	Phe	120	TCC	Ser	36	TAC	Tyr	92	TGC	Cys
TTA	Leu	100	TCA	Ser	16	TAA	Stop	72	TGA	Stop
TTG	Leu	80	TCG	Ser	136	TAG	Stop	52	TGG	Trp
CTT	Leu	105	CCT	Pro	21	CAT	His	77	CGT	Arg
CTC	Leu	85	CCC	Pro	1	CAC	His	57	CGC	Arg
CTA	Leu	65	CCA	Pro	121	CAA	Gln	37	CGA	Arg
CTG	Leu	45	CCG	Pro	101	CAG	Gln	17	CGG	Arg
ATT	Ile	70	ACT	Thr	126	AAT	Asn	42	AGT	Ser
ATC	Ile	50	ACC	Thr	106	AAC	Asn	22	AGC	Ser
ATA	Ile	30	ACA	Thr	86	AAA	Lys	2	AGA	Arg
ATG	Met	10	ACG	Thr	66	AAG	Lys	122	AGG	Arg
GTT	Val	35	GCT	Ala	91	GAT	Asp	7	GGT	Gly
GTC	Val	15	GCC	Ala	71	GAC	Asp	127	GGC	Gly
GTA	Val	135	GCA	Ala	51	GAA	Glu	107	GGA	Gly
GTG	Val	115	GCG	Ala	31	GAG	Glu	87	GGG	Gly

Fig. 7: The RNS-Genetic code (DNA) - Moduli 4 5 7

RNS-Genetic Code

UUU	Phe	0	UCU	Ser	56	UAU	Tyr	112	UGU	Cys
UUC	Phe	105	UCC	Ser	21	UAC	Tyr	77	UGC	Cys
UUA	Leu	70	UCA	Ser	126	UAA	Stop	42	UGA	Stop
UUG	Leu	35	UCG	Ser	91	UAG	Stop	7	UGG	Trp
CUU	Leu	120	CCU	Pro	36	CAU	His	92	CGU	Arg
CUC	Leu	85	CCC	Pro	1	CAC	His	57	CGC	Arg
CUA	Leu	50	CCA	Pro	106	CAA	Gln	22	CGA	Arg
CUG	Leu	15	CCG	Pro	71	CAG	Gln	127	CGG	Arg
AUU	Ile	100	ACU	Thr	16	AAU	Asn	72	AGU	Ser
AUC	Ile	65	ACC	Thr	121	AAC	Asn	37	AGC	Ser
AUA	Ile	30	ACA	Thr	86	AAA	Lys	2	AGA	Arg
AUG	Met	135	ACG	Thr	51	AAG	Lys	107	AGG	Arg
GUU	Val	80	GCU	Ala	136	GAU	Asp	52	GGU	Gly
GUC	Val	45	GCC	Ala	101	GAC	Asp	17	GGC	Gly
GUA	Val	10	GCA	Ala	66	GAA	Glu	122	GGA	Gly
GUG	Val	115	GCG	Ala	31	GAG	Glu	87	GGG	Gly

Fig. 8: The RNS-Genetic code (RNA) - Moduli 7 5 4

6. CONCLUSION

Our model provides the method for producing the genetic code, which has been the subject of numerous attempts to create an easier method. We successfully constructed a genetic code using the concepts of residue number system (RNS). This is the first and advances the cause of representing the genetic code in a quaternary number system. RNS promises to offer some benefits over the conventional number system (binary) in some application specific digital applications. With the fast-evolving field of bioinformatics, it is our firm belief that the RNS-Genetic code model proposed would help revolutionize the field of molecular biological research and applications.

7. RECOMMENDATIONS AND FUTURE WORKS

There is the need to perform a digital (hardware) implementation of the RNS-Genetic code model developed and compare its speed and area with existing models. The number of digital components required to implement the RNS based genetic table could also be assessed. This work could be further extended to consider DNA/RNA sequence (string) comparison and the concept of mutation, as viewed as biological errors. There are many similarities in the way biological and digital errors manifest; thus, the causes of errors - environmental or internal and the nature of the errors - single digit (a base), and burst (chromosomes).

REFERENCES

[1] "The Genetic Code: Francis Crick's Legacy and Beyond | Enhanced Reader." .

[2] C. Yanofsky, "Establishing the Triplet Nature of the Genetic Code," Cell, vol. 128, no. 5, pp. 815–818, 2007, doi: 10.1016/j.cell.2007.02.029.

[3] V. Nanjundiah, "George Gamow and the genetic code," Resonance, vol. 9, no. 7, pp. 44–49, 2004, doi: 10.1007/bf02903575.

[4] J. Marshall, "The genetic code," Proc. Natl. Acad. Sci. U. S. A., vol. 111, no. 16, p. 5760, 2014, doi: 10.1073/pnas.1405652111.

[5] B. Hayes, "The Invention of the Genetic Code," Am. Sci., vol. 86, no. 1, pp. 8–14, 1998, doi: 10.1511/1998.17.3338.

[6] F. H. C. Crick, "The origin of the genetic code," J. Mol. Biol., vol. 38, no. 3, pp. 367–379, Dec. 1968, doi: 10.1016/0022-2836(68)90392-6.

[7] E. V. Koonin and A. S. Novozhilov, "Origin and evolution of the genetic code: The universal enigma," IUBMB Life, vol. 61, no. 2, pp. 99–111, 2009, doi: 10.1002/iub.146.

[8] "amazed knitter's patterns: Code patterns (DNA and other codes)." http://amazedknitterspatterns.blogspot.com/2009/12/code-patterns-dna-and-other-codes.html (accessed Dec. 28, 2020).

[9] "Is our DNA coded like a computer? - Quora." https://www.quora.com/Is-our-DNA-coded-like-a-computer (accessed Dec. 29, 2020).

[10] E. N. Trifonov, "Genome and language-two scripts of heredity," 2013.

[11] S. P. Fedotov, "Simple association of the genetic code with hexagrams of the Book of Changes (I Ching)," Cardiometry, no. 9, pp. 32–43, 2016, doi: 10.12710/cardiometry.2016.9.3243.

[12] E. K. Levy, "Contemporary art and the genetic code: New models and methods of representation," Art J., vol. 55, no. 1, pp. 20–24, 1996, doi: 10.1080/00043249.1996.10791734.

[13] L. R. Nemzer, "A binary representation of the genetic code," BioSystems, vol. 155, pp. 10–19, 2017, doi: 10.1016/j.biosystems.2017.03.001.

[14] H. Kehinde and K. Alagbe, "Residue Number System: An Important Application in Bioinformatics," Int. J. Comput. Appl., vol. 179, no. 10, pp. 28–33, 2018, doi: 10.5120/ijca2018916106.

[15] H. Kehinde Bello and K. Alagbe Gbolagade, "Acceleration of Biological Sequence Alignment Using Residue Number System," Asian J. Res. Comput. Sci., vol. 1, no. 2, pp. 1–10, 2018, doi: 10.9734/ajrcos/2018/v1i224735.

[16] K. O. Boateng and E. Y. Baagyere, "A Smith-Waterman Algorithm Accelerator Based on Residue Number System," Int. J. Electron. Commun. Eng., vol. 5, no. 1, pp. 99–112, 2012, [Online]. Available: http://www.irphouse.com.

[17] K. A. Gbolagade, Effective Reverse Conversion in Residue Number System Processors. 2010.

[18] K. Ahern, "Biochemistry and Molecular Biology How Life Works," pp. 275–279, 2019.

[19] J. Pavsner, Bioinformatics and Functional Genomics 2nd Edition, 2nd ed. .

[20] S. T. de Farias and F. Prosdocimi, "RNP-world: The ultimate essence of life is a ribonucleoprotein process," Genet. Mol. Biol., vol. 45, no. 3, pp. 1–10, 2022, doi: 10.1590/1678-4685-GMB-2022-0127.

[21] J. D. W. with A. Berry, DNA - The secret life. 2003.

[22] "What is DNA?: MedlinePlus Genetics." https://medlineplus.gov/genetics/understanding/basics/dna/ (accessed Oct. 12, 2021).

[23] M. E. B. Yamagishi and R. H. Herai, "Chargaff's 'Grammar of Biology': New Fractal-like Rules," no. December, 2011, [Online]. Available: http://arxiv.org/abs/1112.1528.

[24] "Universal Genetic Code and its Natural Variations: Kimitsuna Watanabe, National Institute of Advanced Industrial Science and Technology, Tokyo, Japan Tsutomu Suzuki, University of Tokyo, Tokyo, Japan - Google Search." .

[25] Y. Omachi, N. Saito, and C. Furusawa, "Rare-Event Sampling Analysis Uncovers the Fitness Landscape of the Genetic Code," pp. 1–8, 2022, [Online]. Available: http://arxiv.org/abs/2210.09666.

[26] "9 Most Important Properties of Genetic Code | Biology." https://www.yourarticlelibrary.com/biology/9-most-important-properties-of-genetic-code-biology/6393 (accessed Oct. 17, 2021).

[27] I. Trenchev, M. Traykov, M. Trencheva, M. Stoyanova, and I. Trenchev, "Mathematical Models for Studying the Properties of the Genetic Code Mathematical Modeling of Optimal TGCs," pp. 40–54, 2017.

[28] B. Ostash and M. Anisimova, Visualizing Codon Usage Within and Across Genomes: Concepts and Tools, no. May. 2020.
[29] R. Lal, “Number system,” Infosys Sci. Found. Ser. Math. Sci., no. October, pp. 55–91, 2017, doi: 10.1007/978-981-10-4253-9_3.
[30] O. Amos and B. Premkumar, Residue Number System Theory and Implementation. 2007.
[31] N. Z. Haron and S. Hamdioui, “Redundant residue number system code for fault-tolerant hybrid memories,” ACM J. Emerg. Technol. Comput. Syst., vol. 7, no. 1, pp. 1–19, 2011, doi: 10.1145/1899390.1899394.
[32] P. V. Ananda Mohan and A. B. Premkumar, “RNS-to-binary converters for two four-moduli sets {2n - 1,2n, 2n + 1, 2n+1 - 1} and {2n -1,2n, 2n + 1,2n+1+1},” IEEE Trans. Circuits Syst. I Regul. Pap., vol. 54, no. 6, pp. 1245–1254, 2007, doi: 10.1109/TCSI.2007.895515.
[33] P. V. Ananda Mohan, “Comments on residue-to-binary converters based on new chinese remainder theorems,” IEEE Trans. Circuits Syst. II Analog Digit. Signal Process., vol. 47, no. 12, p. 1541, 2000, doi: 10.1109/82.899650.
[34] K. O. B. and K. A. G. E.Y. Baagyere, “Bioinformatics: An Importanc Application of Residue Number system,” Bioinformatics: An Importanc Application of Residue Number system, 2011. http://library1.nida.ac.th/termpaper6/sd/2554/19755.pdf (accessed Dec. 30, 2020).
[35] “Introduction to Trees - Tutorialspoint.” https://www.tutorialspoint.com/discrete_mathematics/introduction_to_trees.htm (accessed Jan. 14, 2021).
[36] R. Dhankhar, S. Kamra, and V. Jangra, “Tree concept in data structure,” vol. 1, no. 7, pp. 183–185, 2014.
[37] “Fibonacci Tree | Botanica Mathematica.” https://botanicamathematica.wordpress.com/2014/04/01/fibonacci-tree/ (accessed Oct. 19, 2021).
[38] F. H. C. Crick, “Codon-Anticodon Pairing: The Wobble Hypothesis,” pp. 548–555, 1966.

AUTHORS

Ing. Joshua Apigagua Akanbasiam is currently a PhD. Candidate in the Department of Computer Engineering, KNUST, and he holds a Masters in Telecommunications Engineering and BSc. in Computer Engineering, KNUST. He is a Senior Lecturer and currently the head of the department of Electrical/Electronics Engineering, Dr. Hilla Limann Technical University formerly Wa Polytechnic - Ghana. Prior to Lecturing he served as the Maintenance Manager of Mass Telecom Innovation (MTI) a company that maintained Telecom Power for MTN and Airtel cell sites in the three Northern Regions. He is a member of GhIE and IEEE. His research interest is in Broadband services, Computer Arithmetic and Applications, RNS and its application, Digital Signal Processing, and Bioinformatics.

Prof. Ing. Kwame Osei Boateng holds a Doctorate in Systems Engineering from the Graduate School of Engineering and Science of Ehime University, Japan, Masters in Computer Science from the same university and BSc. (Hons.) degree in Electrical/Electronic Engineering from the University of Science and Technology (now Kwame Nkrumah University of Science and Technology), Kumasi Ghana. Since March 2003, Prof. Ing. Boateng has been working for Kwame Nkrumah University of Science and Technology and has risen through the ranks as a Senior Lecturer, an Associate Professor and currently a Professor. He has held several positions as head of department of Computer Engineering, the dean of Faculty of Electrical and Computer Engineering of the College of Engineering and director of the Institute of Distance Learning and ICT consultant. Prof. Ing. Boateng is a member of the GhIE, IEEE and IEEE Computer Society. He has served on technical programme committee for the 11th IEEE (ATS’02) and IEEE VLSI Test Symposium (VTS), ICAST 2012, WTS 2013, and ESTE 2015. His present research interests are in the areas of design, test and diagnosis of logic and VLSI circuits, test of mixed-signal circuits, network security protocols, applications of residual number systems (RNS), image processing and smart metering.

Prof. Matthew Glover Addo; Mathew Glover Addo is a Professor of Microbiology/Molecular Biology at the Department of Theoretical and Applied Biology, Faculty of Biosciences at the Kwame Nkrumah University of Science and Technology (KNUST). Currently, he is the Director of the Institute of Distance Learning, KNUST. Prof Addo holds a BSc in Biological Sciences from KNUST-Ghana, an MSc in Biotechnology from the University of Bergen, Norway and Doctor of Science/Doctor of Philosophy degrees at the Université Paris Sud IX, Paris, France and KNUST, Kumasi. He completed the doctorate programme in May, 2011. During the doctorate research programme, Prof. Addo designed an efficient screening method for the identification of genes involved in the mitochondrial genome stability using Caenorhabditis elegans as a model organism. He worked with the Functions and Dysfunction of Mitochondrial group of the Institute of Genetics and Microbiology where for the first time, he identified four (4) new nuclear genes (Y105E8A.23, dnj-10, atad-3, and phi-37) involved in mitochondrial stability. Prof. Addo has has 20 years of teaching, collaborative research and consultancy experience. His specialisation is in Gene Expression, Clinical Infectious Microorganisms and Food and Water Microbiology. He has served the international and local communities in a number of capacities and has several peered reviewed publications in reputable international journals. He has also assessed a number of MSc/MPhil and PhD theses. Prof. Addo is a visiting Lecturer/Assessor at the ISA-Lille University, Lille, France. Before his appointment as Director of IDL, Prof Addo was the Dean of Faculty of Biosciences.