An Efficient RNS Arithmetic in Bioinformatics sequences

[1]Olatunbosun Lukumon Olawale
.I. C. T, Department of Computer science. Federal University of Agriculture,
Abeokuta, Nigeria
[2]Lawal Tunde Dauda.
Department of Computer Science. The Federal Polytechnic
Offa, Nigeria
[3]Gbolagade Kazeem Alagbe.
I.C.T, Department of Computer science, Kwara state University,
Malete. Nigeria.
Tel: +2348029290875; + 2348056548519;:+2348136273074

ABSTRACT:
This paper investigates the features and the limitation in the implementation of effective RNS computing algorithms using an special method of data representation and conversion in Residue Number System arithmetic as a helpful tool for enhancing Bioinformatics sequencing computing and to reduce the complexity of calculation in many applications. As required in DNA sequencing involving very high speed sequence comparator (VHSC) with real-time processing requirements. We consider and analyze different stages of data processing in RNS. Based on this analysis, we describe the process of conversion of algorithms using RNS-SWA based comparator with the representation of a large integer using a set of smaller integers, called residues. Emphasis was made on RNS implementation and its inherent arithmetic advantages; data conversion algorithm technique from Chinese remainder theorem CRT, to Mix radix conversion MRC performance, and analysis of Smith Waterman Algorithm base on DNA sequence computing.
.

Keywords: RNS Arithmetic, Moduli-selection, WNS, Smith-Waterman Algorithm, DNA, Bioinformatics, CRT, MRC.

1. INTRODUCTION

The main interest of the Residue Number Systems is to distribute integer operations on evaluations with the residues values [12],[14] where an operation with large integers is made on the residues which are small numbers and such that computations can be executed independently for each modulo allowing a complete parallelization of the arithmetic.
The ability to perform fast arithmetic on large integers is a major issue for the implementation of genetic sequences, protein sequencing and digital signature, particularly from the hardware design point of view [16]. RNS has many advantages over weighted number system (WNS) in terms of encoding large numbers into a set of smaller numbers to speed up computations, the following are time-consuming operations in RNS which affect the wide spread application of RNS in areas like cryptography; overflow detection, sign detection, magnitude comparison and division. Among them, division has modular operations application as can be found in DNA sequence analysis [17]. Currently, fast hardware implementations of DNA sequence is under study while confidentiality and security requirements are becoming indispensable and more important in biotechnology communities.
.

RNS is defined by a set of relatively prime integers called the moduli. The moduli set is denoted as $\{m_1, m_2, m_3 \dots , m_n\}$ where n^{th} is the modulus [12]. Each integer X can be denoted as a set of smaller integers called residues. The set of residues are represented as $\{r_1, r_2, r_3, \dots , r_n\}$ where r_n is the n^{th} residue. The residue r_n is defined as the least positive remainder of an integer value X is divided by the modulus, $\mathbf{m_n}$ **[13]**. This notation based on congruence can be written as; X Mod $m_i = r_i$

Example 1. Let the co prime n-turple be (255, 256, 257), thus
M = 16776960,
a = 10000 is represented by (55, 16, 234) and
b = 300 is represented by (45, 44, 43).

We can verify that
a × b = 3000000 is represented by (180, 192, 39)
Where 180 = 55 * 45 mod 255,
192 = 16 * 44 mod 256 and
39 = 234 * 43 mod 257.

All operations in RNS can be divided into two classes: non-positional (modular) operations, allowing parallelism and non-modular operations related to the need to compute a positional characteristic of a number. The first class includes multiplication and addition. The second class of operations complicates the high-performance computing in RNS. The most important operations of this class are magnitude comparison of numbers and division. In this context, RNS is most often used to solve a certain class of problems in which the number of non-modular operations is minimized. Examples of such problems are digital filters of different nature [3], image processing [4], and large numbers processing as in Genetic sequences [1, 2].

2. BACKGROUND

This modular arithmetic work was originated in the ancient day of the fifth century. The ancient study begins with a mathematical riddle from a third-century book, by the Chinese mathematician Sun Tsu in which puzzle was illustrated as follows: such that how can we determine a number that has the remainders 2, 3, and 2 when divided by the numbers 7, 5, and 3, respectively? The riddle goes like: We have things of which we do not know the number; If we count them by three, the remainder is 2; If we count them by five, the remainder is 3; If we count them by seven, the remainder is 2; How many things are there? Sun Tzu gave the solution to this riddle with a rule called Tai Yen, which was later generalized in 1247 by another Chinese Mathematician Qin Jiushao. The procedure of obtaining the solution to the puzzle was first proposed and known as the Chinese Remainder Theorem. **[1], [2], [3], [4], [5]** but the use of this arithmetic to represent number was introduced only in 1959 by H.I. Garner

2.1. Bioinformatics in Literature

Bioinformatics is the application of information technology and computer science to the field of molecular biology in those areas of genomics involving large-scale DNA sequencing. It entails the creation and advancement of databases, algorithms, computational and statistical techniques, with activities including mapping and analyzing DNA and protein sequences, aligning different DNA and protein sequences to compare them and creating, viewing 3-D models of protein structures including sequence alignment, gene finding, genome assembly, protein structure alignment and prediction, prediction of gene expression and protein-protein interactions, genome-wide association studies and the modeling of evolution Comparative genomics, modeling biological systems, protein-protein docking. Analysis of mutations in cancer and regulation. It is focus on developing and applying computationally intensive techniques e.g., pattern recognition, data mining, machine learning algorithms, and visualization [4].

3. Basic RNS Arithmetic Models

Residue representation:

The RNS which is an unweighted number can uniquely represent all integer numbers, X, that lie in its dynamic range [17]. The dynamic range of an RNS is determined by the moduli set and it is denoted as M, where; in equation (1) However, RNS provides unique representation for all integers in the range between 0 and M − 1. In a situation where the integer X is greater than M − 1, RNS repeats itself. As an integer system capable of supporting high speed concurrent arithmetic [6] an integer is decomposed into a set of smaller integers i.e. with shorter binary representations, which can be processed independently and in parallel [12],[15]. This basis of RNS is set of pair wise prime integers $S = \{m_1, m_2, . \quad , m_n\}$, where gcd $(m_i, m_j) = 1$ for i = j. The set S is called the moduli set and the dynamic range of the number system is [0,], where M is the product of all moduli m_i in S [17]. Any integer X within the dynamic range has a unique RNS representation given by an ordered set of residues

$$X \rightarrow \{x_1, x_2 . . . x_n\},$$

$$xi = \left| X \right|_{mi} . \qquad \{1\}$$

where $(X)_{mi}$ denotes X mod mi. The most important characteristic of the RNS representation is that it is a non-weighted number system, which facilitates parallel computing. (CRT) If integers **A** and **B** have RNS representations $\{a_1, a_2 . . . a_n\}$ and $\{b_1, b_2, . . . , b_n\}$ respectively. then the RNS representation of

$$C = A \circledast B \text{ such that } C \rightarrow \{C_1, C_2 . . . C\mathbf{n}\} \qquad \{2\}$$

$$Ci = \left| a_i \circledast b_i \right| \; m_i \qquad \{3\}$$

Where $\circledast$ denotes **addition, subtraction, multiplication** or any combination of the three. The computation of **ci** depends upon **ai, bi** and *mi* only. Hence, each ***ci*** can be computed using a separate arithmetic unit, often called a channel. The reconstruction **X** from $\{\mathbf{x_1, x_2 . . . x_n}\}$ is based on the Chinese Remainder Theorem (CRT) The RNS which is an un weighted number can uniquely represent all integer numbers, **X,** that lie in its dynamic range [17]. The dynamic range of an RNS is determined by the moduli set and it is denoted as **M,** where;

$$M = \Pi^{n}_{n_i} m_i \qquad \{4\}$$

4. Parallel Computations in RNS

The ability to perform fast arithmetic on large integers is still a major issue for the implementation of genetic sequences, protein sequencing and digital signature, particularly from the hardware design point of view [16]. RNS has many advantages over weighted number system (WNS) in terms of encoding large numbers into a set of smaller numbers to speed up computations, the following are time-consuming operations in RNS which affect the wide spread application of RNS in areas like cryptography; overflow detection, sign detection, magnitude comparison and division. Among them, division has modular operations application as can be found in DNA sequence analysis [17]. Currently, fast hardware implementations of DNA sequence is under study while confidentiality and security requirements are becoming more and more important.

The main advantage of RNS in the context of high performance computing is the opportunity to do some operations in parallel. Each number X from the interval [0, M] is substituted by n number x_i each of which belongs to the interval $[0, m_i]$ for i = 1,2,3,..n. Since M is the product of all m_i, then X is larger than each x_i. If X is of N bit length then each x_i (under the condition of balanced moduli set m_i) is of N /n bit length.

Addition and multiplication have asymptotical complexity **0 (N) and 0 (N^2)** respectively. In the best case, multiplication has complexity 0 (Nlog$_2$ N log$_2$ log$_2$ N) [17]. Using this property and combining addition and multiplication, we can achieve a dramatic increase in speed of computations, which are done in parallel in RNS.

RNS is based on modular non-positional representation of numbers. Each digit in RNS is a residue of the division by a number called a base. All bases for each digit are form a moduli set. The uniqueness of the representation of numbers in RNS is only guaranteed under the condition that all moduli are pair wise co prime.Let (m_1,m_2,m_3,.m_n) be a given moduli set. It determines a unique RNS. A number X in this RNS can be represented as follows:

$X = (x_1,x_2,x_3,\ldots..x_n)$, where $x_i = |X|_{mi}$ {5}
for all i = 1,2,3,..n.

With CRT X should belong to the interval [O,M] where M = m_1,m_2,m_3,....m_n is dynamic range of the RNS. With this representation, addition and multiplication can be done in parallel. Let X = (x_1,x_2,x_3,.x_n) and Y = (y_1,y_2,y_3,..y_n) be numbers in RNS then their sum S = X + Y = such that S_i = (s_1,s_2,s_3,...s_n) and product Q = X*Y = such that Q_i = (q_1,q_2,q_3,...q_n) can be computed using following formulas

$s_1 = |x_i + y_i|\, m_i$ and $q_i = |x_i \cdot y_i|\, m_i$, {6}
for all I = 1, 2, 3…. n

These operations do not require carries between digits as in positional number systems. This property allows performing these operations independently.

The simplicity and speed are the advantages of RNS arithmetic and is paramount in DNA sequencing. The Fig.1 below depicts the structure of an adder, subtractor and multiplier for RNS arithmetic [6]. This representation solves the problem of carry propagation with 6-bits residues.

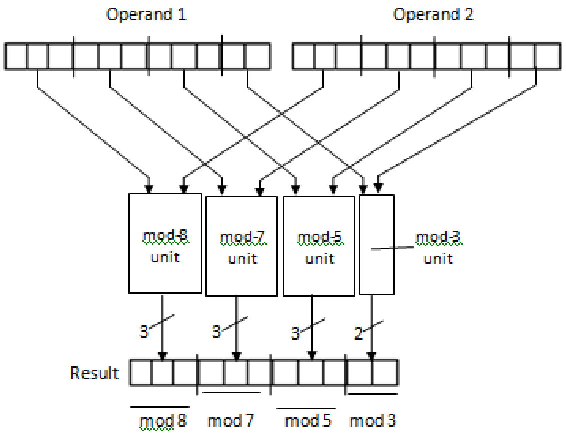

Fig.1. Operands operation for RNS (8/7/5/3)

5. Data conversion in residue number system

Data conversion from RNS to weighted number system vis-à-vis encoding and decoding is an important operation in RNS based system design [7],[14]. The achievement of hardware realization depends on both data conversion and choice of moduli set. There are different approaches to this and include among other conversions the Chinese Remainder Theorem and the Mixed Radix Conversion operation,[6]

In this section, we look into
1. The conversion from Decimal/Binary to RNS.
2. Conversion from RNS to mixed radix form.
3. Conversion from RNS to Binary/Decimal.

Remark: The mixed-Radix representation allows us to detect the sign of a number and compare the magnitudes of two RNS numbers. This feature is very useful in molecular sequence computation when there is likelihood mutation in the cells sequences

Data Conversion Forms	**Comparison Between RNS and Mixed Radix Computation**	
	Decimal/Binary to RNS	**RNS to mixed radix**
Problems Procedure	Given a number y with respect to the moduli m_i $0 \leqq i \leqq k_{-1}$ If y is an unsigned binary number Then : (($y_{k-1}\dots y_1y_0)_{two})$ $m_i = ((2^{k-1}y_{k-1})\ m_i + \dots + (2y_1)_{mi} + (y_0)m_i)m_i$ (?) With $(2^j)_{mi}$ for each **i** and **j**, residue x_i of y (Mod $_{mi}$) is computed.	**RNS**($M_{k1} \vert \dots \vert m_2 \vert m_1$)→MRNsystem.**MRS**($M_{k1} \vert \dots \vert m_2 \vert m_1 \vert m_0$) k-digit'$M_{k-2..}/m_2m_1m_2\dots m_2m_1m_0\dots m_1m_0\ m_0$) k-digit. Y = $(X_{k-1} \vert \dots \vert x_2 \vert x_1 \vert x_0)_{RNS} = (Z_{k-1} \vert \dots \vert z_2 \vert z_1 \vert z_0)_{MRS}$.Y=$Z_{k-1}(m_{2..}m_1m_0)+\dots + z_2(m_1m_0) + z_1(m_0) + z_{0.}$ **$Z_0 = x_0$,** subtracting **$z_0 = x_0$** from both **RNS and MRS** y– x_0 = $(x^{\vert}_{k-1} \vert \dots \vert x^{\vert}_2 \vert x^{\vert}_1 \vert 0)_{RNS} = (z_{k-1} \vert \dots \vert z_2 \vert z_1 \vert 0)_{MRS.}$ where $x^{\vert}_j = (x_j - x_0)\ m_{j.}$If we divide both RNS and MRS by m_0 yield: $(x^{\Vert}_{k-1} \vert . \vert x^{\Vert}_2 \vert x^{\Vert}_1)_{RNS} = (z_{k-1} \vert \dots \vert z_2 \vert z_1)_{MRS}$
Example Solution	Let y = $(1010\ 0100)_2 = (164)_{10}$ in RNS (8/7/5/3) yMod8 = $x_3 = (y_2\ y_1\ y_0)_2 = (100)_2 = 4$. But, y = $2^7 + 2^5 + 2^2$and Mod7, Mod5, Mod3 respectively. For j = 7,5, 2. $X_2 = \vert y \vert_7 = \vert 2+4+4 \vert = 3$. $X_1 = \vert y \vert_5 = \vert 3+2+4 \vert_5 = 4$.$X_0 = \vert y \vert_3 = \vert 2+2+1 \vert_3$=TheRNS(8/7/5/3)=$(164)_{10}$→$(4/3/4/2)_{RNS}$	Convert y = $(0 \vert 6 \vert 3 \vert 0)_{RNS}$ to MR $z_0 = x_0 = 0$.$(0 \vert 6 \vert 3 \vert 0)_{RNS} = (0 \vert 6 \vert 3 \vert 0)_{RNS}$* $(3 \vert 5 \vert 2 \vert$ = $(0 \vert 2 \vert 1 \vert -)_{RNS}$. **If z_1 = 1**.replace 1 and divide by 5. $(7 \vert 1 \vert 0 \vert -)_{RNS} = (7 \vert 1 \vert 0 \vert -)_{RNS} * (5 \vert 3 \vert - \vert -)_{RNS} = (3 \vert 3 \vert - \vert -)_{RNS}$. **If z_2 = 3**, replace 3 and divide by 7 $\underline{(0 \vert 0 \vert - \vert -)}_{RNS} = (0 \vert 0 \vert - \vert -)_{RNS} * (7 \vert - \vert - \vert -)_{RN} = (0 \vert - \vert - \vert -)_{RNS}$. **If z_3 = 0.** Then y = $(0 \vert 6 \vert 3 \vert 0) = (0 \vert 3 \vert 1 \vert 0)_{MRS}$ = **48**
Remark	The worst case requires k modular addition by a **k bit** number **(000/011/001/00)$_{MRS}$ and (000/011/000/00)$_{MRS}$**	

Sensitivity Analysis: Table 1 did shows the comparison between the average times for the two different moduli sets, and it was observed that they both have almost the same average time. **$Y^{|} = (0|6|3|0)_{RNS} = 48$** and **$(5|3|0|0)_{RNS} = 45$** Comparing the binary coded equivalent of MRS with RNS thus: the equivalent mixed Radix Representations $(0/3/1/0)_{MRS}$ and $(0/3/0/0)_{MRS}$ **or** $(000/011/001/00)_{MRS}$ and $(000/011/000/00)_{MRS}$ when coded in binary can be compare as ordinary numbers

5.1. Analysis of the structure of algorithms in RNS

In this section, we discuss the main features of the implementation of RNS systems, based on the model presented in Figure 1. All operations can be divided into forward conversion of Weighted Number System (WNS) to RNS, and inverse conversion, computations in each digit and non-modular operations. We consider each type of operations separately.

5.2. Conversion from RNS to Binary/Decimal

Residue to decimal conversion has some speed limitations, and is derived from

i. The weights for the RNS directly based on the Chinese Remainder Theorem. The magnitude of an RNS number can be obtained from the CRT formula.

ii. The Mixed Radix representation of the RNS number uses the weight radix of the mixed radix position to complete the conversion.

6. Chinese Remainder Theorem:

We consider a n-tuple of co prime numbers $(m_1, m_2, .m_n)$. We

note M = [$\prod$ mi,] If we consider then-tuple $(x_1, x_2, . . , x_n)$ of integer such that xi < mi. Then there exits an unique X which verifies:

$$X = \left| \sum_{i=1}^{n} M_i \left| M_I^{-1} \times x_i \right| m_i \right|_M \qquad \{7\}$$

where $_M$ = Dynamic range $^n\prod_{i=1} m_i$ such that $m_{i\ = 1.2.3\dots n}$ and x_i= Residues

mi = Moduli set; $M_{i\ =\ M\,/\,m_i;}$ M_i^{-1} = multiplicative inverse of m_i

$_{X\ =}$ Binary / Decimai equivalent of x_i

Example1: Let X = $\{0, 3, 0\}_{RNS}$ {4/4/3}

The value of X can be computed from Equation { }

$$\text{CRT}\quad X = \left| (12* |3*0|_{5}) + (15* |3*3|_{4}) + (20* |2*0|_{3}) \right|_{60}$$

$$X = |0+15+0|_{60} \qquad X = |15|_{60}$$

$$X = 15$$

The value of X is computed in Parallel for the CRT conversion while in MRC the value of X done in Sequential. This accounts for advantage over each process.

7. Mixed Radix conversion

Another alternative schemes is the Mixed Radix conversion which does not involve active range of the large modulo M with calculations approaching a low yield complexity of 0(n) when compared to the CRT computation whose complexity is of order $0(n^3)$.

Given an RNS number $(x1,x2,x3\dots xk)$ for the moduli set $(m1,m2,m3\dots.mk)$ with the constant ai,j where $1 \le i,j \le K$ as $ai,j\ mi=1\ mod\ mj$, the decimal equivalent of it can be computed as [1]

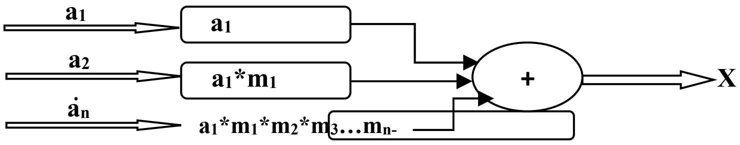

Fig.2: the schematic diagram of the MRC

$z_1, z_2 \ldots\ldots z_n = a_1, a_2 \ldots\ldots\ldots a_n$

$z_1 = y_1 \ (mod\ m_1)$ {8}

$z_2 = (y_2 - z_1) a_{12} \ (mod\ m_2)$ {9}

$z_3 = (y_3 - z_1)\ a_{13} - z_2)_{23} \ (mod\ m_3)$ {10}

$X = a1 + a2m1 + a2m1m2 + a3m1m2m3 + \ldots + anm1m2m3m\text{n-1}$ {11}

the mixed radix digit ai =1, k can be computed as follows:

$a1 = x1:$ {12}

$a2 = (x2 - a1) \left| \ m1^{-1} \right|_{m2} \left| \ m2 \right| :$ {13}

$a3 = (x3 - a1) \left| \ m1^{-1} \right|_{m3} - a2 \left|_{m3} \right| m3$ {14}

.

.

$a_n = (((x_k - a_1) \left|_{m1}^{-1} \right| m_k - a_2) m_{2-1}\, m_k - a_k - 1) m_{k-1} m_k m_k \quad \ldots$ {15}

The equations above can be directly utilized, verify by Euclidean algorithm if the moduli set $\{m1m2m3\ldots mk\}$ are relatively prime and that gcd $(mi,) = 1$, for i ≠ j. **[2][4]**

Example 6: For a moduli set (**7, 8, 9**), the integer X = **10** can be represent in the residue format as(**3, 2, 1**).Using the MRC Conversion method, we can recover back the integer **10** from these residues

MRC semantic model iteration [a$_n$]	Moduli Computation/Iteration		
	A_{12}	A_{13}	A_{23}
$z1 = y1\ (mod\ m1)$ $z2 = (y2 - z1)a12$ $(mod\ m2)$ $z3 = (y3 - z1)$ $a13 - z2)23(modm3)$	$m1 = 1$ $mod\ m2$ $7\ a_{12} = 1$ $mod\ 8$ $7 * 6\ mod$ $8 = 1$ $a_{12} = 6$	$m1 = 1$ $mod\ m3$ $7\ a_{13} = 1$ $mod\ 9$ $7\ a_{13} = 1$ $mod\ 9$ $7 * 4\ mod$ $9 = 1$ $a_{13} = 4$	$m2 = 1$ $mod\ m3$ $8\ a_{23} = 1$ $mod\ 9$ $8 * 8$ $mod\ 8$ $= 1$ $a_{23} = 8$

Table 2. MRC semantic model iteration for moduli set (7, 8, 9) computation.

Substitute the respective variables in eq.4 yield: $z_1 = 3\ (mod\ 7) = \mathbf{3} : z_2 = (2-3) * 6\ (mod\ 8) = \mathbf{2} :$
$z_3 = (1-2) * (4-2) * 8\ (mod\ 9)\ z_3 = \mathbf{6}$
$: \ X = z_1 + z_2 m_1 + z_3 m_1 m_2$

MRC is an ideal scheme when considering the computational complexity since it only requires operations of modulo m_i. Since mixed radix (MR) system is a weighted system, the comparison can be accomplished with comparing the corresponding coefficients of two integers, respectively

7.1. The pseudo code of the scheme for Mixed Radix Conversion (MRC)

```
Input: Given residues of X as ($x_1$; $x_2$. $x_n$) of moduli {$m_1$;$m_2$;….., $m_n$}.
Output: $|X|_m = \prod_{I=1}^{n} p_i \sum_{j=0}^{i-1} m_j$ |where $m_o = 1$.
For i = 1 : n do
    $p_i = x_i$;
    For j = (i + 1) : n do
        $X_j = \left| \left| x_j - x_i \right| m_i^{-1} \right|_{m_j}$
    end
end
$|X|_M = \sum_{I=1}^{n} p_i \prod_{j=0}^{I=1} m_j$
```

This shows that MRC is an iterative method where comparison of two residue vectors requires n (n-) modula multiplications

8. Moduli set selection

The number of bits generally required in RNS is greater than that of weighted number systems because RNS gives the number of residues same as the cardinality of the moduli set, increasing the number of bit required to express it in RNS. A number system is said to have higher bit efficiency if the bit required to represent a particular dynamic range is lower. There are many important parameters that determine the efficiency of RNS and bit efficiency is one of them .The bit efficiency depends on the choice of the moduli set [7]. The choice of moduli selection plays the paramount roles and dictates the speed up of computations, the more the dynamic range of the moduli set the faster the system and the slower its reverse conversion There are many different options for a moduli set, each of them may be used for certain applications. In the selection of a moduli set, a number of factors should be taken into account. These factors will influence the further computations in RNS. For a given platform for the implementation, concurrency and independence of computations over each modulus, it is necessary to:

- Select balanced moduli set with respect to their magnitude and complexity of computations. If the moduli set is unbalanced, it may lead to long waiting time for some threads (nodes). There are several Techniques for moduli set generation reported in the literature **$\{2^n, 2^n - 1, 2^{n-1} - 1\}$ and $\{2^{2n} + 1, 2^n + 1, 2^n - 1\}$.**
- Minimize the redundancy of RNS representation. If P_{WNS} is the number of bits of a number in WNS, and P_{WNS} is number of bits

required to represent that number in RNS, then, $P_{WNS} < P_{RNS}$ .Different moduli sets have different redundancy where redundancy leads to additional storage, transmission and processing costs effective.

- Select the modules with a minimum computational cost for the given problem, taking into account the implementation of modular operations and considering the difficulty of forward and reverse RNS conversions.

- They must be relatively primed i.e they should be pairwise prime. With $gcd\ (m_i,m_j) = 1$ for all $m_i \neq m_j$.
- The moduli m_is should imply simple binary to RNS and RNS to binary conversions as well as simple RNS arithmetic.
- The smaller the moduli, the faster the arithmetic operations and Moduli numbers can be restricted to power of 2 with optimum large dynamic range to avoid overflow.
- The higher the dynamic range of the moduli set, the faster its forward conversion and the slower its reverse conversion.
- Efficiency of the RNS moduli should be considered and high efficiency is more desirable, example the RNS (15|13|11) - require 12 bits - it can represent $2^{12} = 4096$, whereas only 2145 numbers are presented - the efficiency is 52% **[4, 6.17]**
- Select prime numbers in sequence until a desired dynamic range is obtained i.e. the moduli product should be large enough to implement the desired dynamic range.
- Each moduli m_i should be as small as possible so that operations modulo m_i require minimum computational time with a well balanced decomposition of the dynamic range. This means that the difference in word length between the moduli should be as small as possible

9. LIMITATIONS OF RESIDUE NUMBER SYSTEM:

RNS has limitation in applications involving magnitude comparisms, sign detection, overflow detection, division, reverse conversion etc High speed is achieve in the operation with addition, subtraction, multiplication by supporting carry free addition, borrow free subtraction and digit to digit multiplication without partial product [4],[7]. such that the system emerges as a better option for processing data in the advancing Biotechnology [13].

But arithmetic operation like scaling, overflow, division, magnitude comparison, and sign detention and are very difficult in RNS.The complexity of the system to implement and the coding overhead due to large number of bits that are required to denote the whole set of moduli in computation are also drawback.

10. Sequence Alignment in Bioinformatics

Many fast algorithms for computing sequence alignment, such as **FASTA** [6]**, [10]** and **BLAST** [4]**, [17]** compete with **SWA** at expense of accuracy. The SWA is an optimal method for homology searches and sequence alignment in genetic databases and makes all pair wise comparisons between two strings of DNA. It achieves high sensitivity as all the matched and near-matched pairs are detected; at the expense high computation time. We believe RNS as a tool; can be used to address the computational time limitation of SWA..**[4],[6], [14].**

In calculating the local alignment, the matrix H(i ,j) is used to keep track of the degree of similarity between the two sequences **(A_i;B_j)** that are aligned. Each element of the matrix H(i; j) is calculated according to the following:

$$H_{(I,J)} = \max \begin{cases} \mathbf{0} \\ H\,(i-1, j-1) + S(i,j) \\ H(i-1, j) - d \\ H(i, j-1) - d \end{cases} \qquad \mathbf{\{16\}}$$

Where

H (i, j) is the maximum similarity score between the two sequences compared. **S(i, j**) is the similarity score incomparing sequence **A_i** to sequence **B_j** and **d** is the gap penalty for a mismatch in the comparison.

Steps involves

- **The initialization :-** Matrix H is initialized by setting H(0,j) = 0 and H(i,0) = 0 for all i and j.
- **Matrix fill :-** This step is done to fill in all the entries of the matrix using equation $X^{|}{}_{\mathbf{1}}, X^{||}{}_{\mathbf{1}\text{and}}\ X^{|||}{}_{\mathbf{1}}$ where X = $(x_1, x_{2 \ldots} x_n)$ and $n > 0$ computationally intensive where hardware acceleration runs.
- **Trace back: -** Where the matrix scores entered are trace back to inspect optimal score local alignment.

11. Application of Residue Number System in DNA Sequence

The study DNA sequence alignment and its analysis is very important in bioinformatics and provides the following benefits:

- Allows correlation of DNA information diseases. A relation between diseases and inheritance can be studied. Example genes identified to be involved in breast cancer [**5].**
- Because of the trade-offs between word length and hardware size, and between propagation delay and accuracy, various types of number representation have been proposed and adopted in which the Residue Number System (**RNS)** will require a very high speed sequence comparison has been evaluated with

emphasis on its arithmetic advantages in real life application in bioinformatics. RNS has been successfully used in many applications that involve high computations because of the carry propagation absence in the RNS embedded processor.[4][5]

- Allows us to trace evolutionary trend.: Example, if Bioinformatics are able to find the similarity between any two sequences, they will be able to trace and understand evolutionary trend between them **[6]**

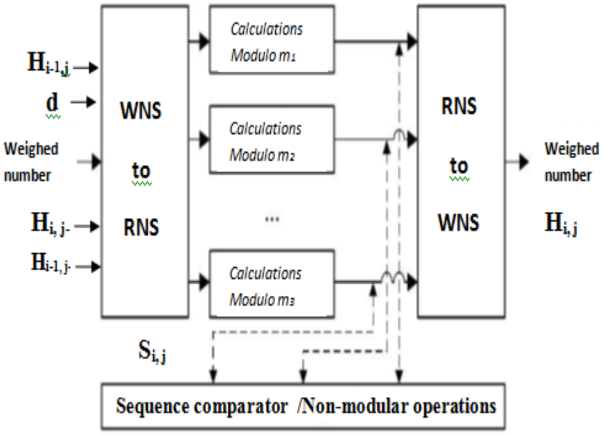
$H_{i-1,j}$
d
Weighed number
$H_{i,j-}$
$H_{i-1,j-}$
WNS to RNS
Calculations Modulo m_1
Calculations Modulo m_2
...
Calculations Modulo m_3
RNS to WNS
Weighed number
$H_{i,j}$
$S_{i,j}$
Sequence comparator /Non-modular operations

Fig. 3. – **Schematic** Model of RNS-SWA Based computations in DNA Sequences

12. RNS-SWA Based Architecture

In this, RNS-SWA architecture the hardware realization is achieved base on the number of specified moduli set that is defined e.g. **2^{2n} -1,2^n, 2^{2n} + 1,** inputs from the SWA local alignment of the form in **eq 14.** Follow this process.

- Equation 6 **:** H (i-1,j), H (I,j-1), H (i-1,j-1), d and S (i,j) is input and the binary to residue converter is assigned and executed by forward conversion to RNS processor .MOD 1

- The result is sent to the RNS processor/converter MOD2 where arithmetic computation is done in the embedded processor taking the advantage of the inherent properties of the carry free propagation in RNS and enableling the realization of high-speed and low-power consumption.

- The corresponding final result X(i,j) is then Converted from residue to binary converter as binary/decimal number.
- However, we can build system in these areas capable to design RNS based general purpose architectures that can manage the above problems, also converters are required to interface a RNS data path to the conventional systems based on binary representation.

13. Conclusions

The use of novel moduli set, as guideline for a Reconfigurable SWA-DNA-RNS based Processor is a task to process some predefined functions. RNS minimizes the carry propagation in this work; the scheme employed with its conversion technique has been implemented in DNA sequences where high speed computations are required**.**

The RNS arithmetic advantages address the computational challenge of the SWA. The availability of embedded RNS processor, and realization of high-speed and low-power consumption with accuracy, as required in bioinformatics field operation was achieved. Future research could employ use of dedicated moduli set selection technique to futher build a specialized RNS-SWA based architecture with the implementation capability via FPGA tools to address mutations, detection and correction of errors in molecular sequence computing.

References

[1]. H. M. Yassine, W. R Moore, [1991] "Improved mixed-radix conversion for residue number system architectures", IEE Proc.-G, vol. 138, no. 1, pp. 120-124.

[2]. Bin Cao, Chip-Hong Chang. S.Thambipillai. [1994]. “Adder based residue to binary converters for a new balanced 4-moduli set” Proceedings of the 3rd Inte S..J.Piestrak, "IEEE Trans. Comp., vol. 43, pp.68-77’

[3]. A..B..Premkumar, [2013] "Improved memory less RNS forward converter based on periodicity of residues, "IEEE Trans. Circuits and Systems-II, express Briefs, vol. 53, no.2.

[4]. Hassan Kehinde Bello and Kazeem Alagbe Gbolagade [2017] “A Survey of Human Deoxyribonucleic Acid” British Journal of Applied Science & Technology. 21(5): 1-10.

[5]. K.A Gbolagade, R. Chares, L. Sousa, S.D Cotofana [2009] “Residue –to- binary converters for the {22n+1-1, 22n, 2n-1} moduli set. 2nd IEEE International conference in adaptive science & technology. Accra Ghana Pp 26 – 33.

[6] L.O. Olatunbosun, A.A. Adam, K.A. Gbolagade. [2020] “RNS Bases in Computer Architecture for DNA Sequence Application “ijecs.vol 9 issue 7

[7] R. Conway and J. Nelson, “New CRT-based MRC converter using restricted moduli set,” IEEE Transactions on Computers 52(5), pp. 572–578, 2003.

[8] A. P. Shenoy and R. Kumaresan, [1989] “Fast base extension using a redundant modulus in RNS,” IEEE Transactions on Computer 38(2), pp. 292–296.

[9] D. Knuth,[1981]. Semi numerical Algorithms, vol. 2 of The Art of Computer Programming, Addison-Wesley,

[10]. K.A. Gbolagade [2009c]. A shorter algorithm for efficient residue to decimal conversion. Advances in computer Science and Engineering, 3(2), pp. 147-156.

[11]. K.A. Gbolagade[2011].” New adder-base RNS-to-binary converters for the {2^{n-1}+1,2^{n-1}-1,2^n} modoli set”. ISRN Signal Processing Journal, 7,1-7

[12]. K.A. Gbolagade [2013] “An efficient MRC base RNS-to-binary converters for the {2^{2n}-1,2^n, 2^{2n+1}-1} modoli set”. International Journal of Advance Research in Computer Engineering and Technology 2(10), 2661-2664.

[13] B..Parhami. and H.F.Lai [1993] “Alternate Memory Compression Schemes for modular Multiplication “IEEE Trans. Signal Processing Vol. 41, pp.1378-1385..

[14] W. David. Mount, [2004] “Bioinformatics; sequence and genome analysis” (second edition). Cold spring Harbor Laboratory Press,

[15]. Dan Gusfied. [1997] “Algorithms on Strings.Trees and Sequences: Computer Science and Computational Biology” Cambridge University Press,

[16] International Journal of VLSI design and Communication Systems (VLSICS) Vol.3, No.5, October 2012 176

[17] Behrooz Parhami [2000] “Computer Arithmetic Algorithms and Hardware Designs“Oxford University Press. Pg 56-61.