Original Research Article

Residue Number System-Based Approach to Minimise Energy Consumption in Wireless Sensor Networks

ABSTRACT

This study harnesses the useful number properties of the residue number system (RNS) to minimise energy consumption in a wireless sensor network. In a traditional cluster-based wireless sensor network, large bit representations of aggregated packets are transmitted to the base station. However, large bit patterns of packets are slower than smaller bits. With the proposed approach, aggregated data is split into a pre-specified number of transmission channels using a moduli set. Cheap energy cost routes from the cluster heads are computed to deliver the chunked aggregated data to the base station. Forward and reverse converters are proposed to encode data into RNS and decode the RNS data that reaches the base station. Simulation is done with MATLAB to implement the proposed data splitting method and evaluate performance. The experimental results suggest that the proposed method is more effective at minimising transmission energy when compared with traditional approaches in which complete packets are transmitted.

Keywords: Wireless Sensor Network; Residue Number System; Clustering; Transmission Energy

1 INTRODUCTION

Wireless sensor networks (WSN) are a collaborative means of monitoring and reporting essential parameters of a target environment using low-cost, battery-powered sensor nodes. WSN can detect variations in environmental parameters such as temperature, pressure, humidity, sound, intensity, vibration and motion. Because of the advancements and miniaturization of micro sensors, these

sensing technologies have become widespread. They have been put in place to timely and reliably give alert appropriate disaster management units on coming disasters like wildfires, floods, tsunamis, earthquakes, and hurricanes [1][2][4], as well as for assessing the strengths and weaknesses of built structures, including roads, buildings, and bridges [22]. The availability and purity of water supplies, particularly for home consumption, are critical to humans and animals' survival. WSNs have been used to monitor the quality of water supplies as well as the quality of air, which is just as vital to life as water. Other uses include monitoring plant growth and animal movement, making predictions about disasters and assessing disaster control measures, monitoring and evaluating responses to patient healthcare plans, surveillance for securing homes and offices, monitory and controlling inventory and stock control in store locations, and tracking military target[1[15].

The primary source of power for sensor nodes is a battery. The cost of replacing these batteries is high, if not impossible [16]. As a result, many WSN issues, such as energy, processing, and storage capacity constraints, result from power consumption. When battery life deteriorates exceedingly quickly, network lifespan drastically diminishes. To ensure the energy efficiency of sensor nodes, there are competing needs to be resolved. Rigorous hardware and software, and route control is required to keep sensor batteries alive and the network running for as long as possible.

In a wireless sensor network, minimising energy consumption is one of the most critical issues of interest to researchers. The most energy-consuming activity of a sensor node is data transmission. Up to about 80% of a sensor node energy is expended in data reception and transmission put together [25]. Accordingly, it will take thousands of processing operations to expend the same amount of energy as transmitting just a bit of data. Based on this, the processor unit of a sensor node can be assigned additional processing tasks such as data encoding and still be less energy-intensive. The primary data transmission structure in traditional cluster-based wireless sensor network is to transmit large bit representations of an aggregated packets. However, large bit representations of messages transmit slower compared to those of smaller bits [8]. The residue number system is one of the available number representations for designing systems with high-speed computations and low power consumption. Though yet to receive a widespread application in general-purpose computing, its use is found in many digital signal processing systems such as digital filters and transforms [29]. Therefore, this study harnesses the RNS to split packets into smaller residual parts with the view to minimising transmission energy expended by cluster heads during transmission of packets to the base station.

The rest of the paper is presented as follows; background information on the residue number system is presented in Section 2. In Section 3, a review of some relevant RNS applications is provided. The methodology and proposed technique to achieve the research aims is highlighted in Section 4. Section 5 presents analysis of results from the proposed technique. The paper concludes in Section 5.

2 BRIEF BACKGROUND OF RESIDUE NUMBER SYSTEM

Residue Number System (RNS) has its roots in the ancient book of Sun Tzu [9]. However, its revival started in the 1950s as an alternative number system for applications requiring fast arithmetic and fault-tolerant operations [26]. The RNS encodes a number as its remainders with respect to a specified set of relatively prime moduli. Many of its natural features make it valuable and attractive for special-purpose computations. For example, RNS provides no carry mechanism, allowing parallel addition and multiplication operations without interaction between digits. Also, the allowance of parallel and carry-free computations benefits faster arithmetic operations than straight binary encoding [27]. Furthermore, because residues reveal no weight information, error in any residue positions is not propagated to other digit positions. RNS also offers some valuable properties for error detection and correction in computerised systems. These and many of its inherent features helps in building fault-tolerant systems required of many data communications systems including in digital filtering

[18],[13],[24],[30][20], convolution [12] and Discrete Cosine Transform [17]; communication engineering, cryptography, image processing and speech processing [26], stenographic and cryptographic schemes [6] and Rain fade mitigation [3]. Other areas of application of RNS are; direct digital frequency synthesis, Discrete Fourier Transform, and Fast Fourier Transform [18]. Bottlenecks exist in a smooth RNS realisation that limits the general implementation, especially in general-purpose computing. Authors in [28] and [21] were the first to look at the arithmetic limitations of the residue number system, including; overflow detection, reverse conversion, magnitude comparison, moduli selection, and division.

A residue number system is defined by a set of pair-wise relatively prime integers called the moduli set. The moduli set is denoted as $\{m_1, m_2, \ldots, m_n\}$, $i = 1, \ldots, n$, such that $gcd(m_i, m_j) = 1$ for $i \neq j$, where $gcd(m_i, m_j)$ means the greatest common divisor of m_i and m_j. Each integer can be represented as a set of smaller integers called the residues. The residue-set is denoted as $\{x_1, x_2, \ldots, x_n\}$, where x_i is the i^{th} residue.

Each residues x_i, of X is defined as the least positive remainder when X is divided by the modulo m_i. This relation can be notationally written based on the congruence in Equation (2.1).

$$x_i \equiv X mod m_i \tag{2.1}$$

For example, the residues of $X = 23$ with respect to the moduli set $\{8, 7, 3\}$ is $\{7, 2, 2\}$.

A given RNS is capable of uniquely representing all integers that lie in its dynamic range (DR), M (Taylor, 1984). Given the moduli set $\{m_1, m_2, \ldots, m_n\}$, the dynamic range for positive integers is denoted in Equation (2.2) below and corresponds to a range of all positive integers from 0 to M–1.

$$M = \prod_{i=1}^{n} m_i \tag{2.2}$$

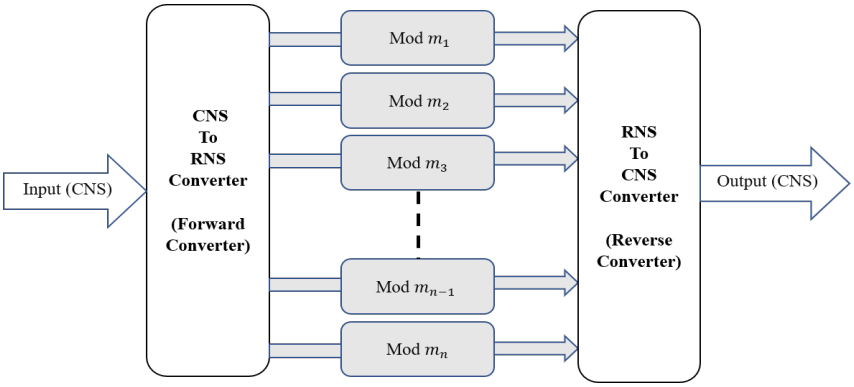

Figure 1: Structure of an RNS processor

Encoding a decimal number into an RNS code (called forward conversion) is often not a challenge. However, applications based on RNS require conversion of data in residue representation to a weighted number system – binary or decimal – to use the encoded data. This process is a significant bottleneck [14] in successfully realising an RNS implementation. The typical structure of an RNS processor that supports conversion from a conventional number system (CNS) – binary or decimal – to an RNS number is as in Figure 1. Known RNS to weighted number systems conversion techniques exists, including the Chinese Remainder Theorem (CRT) and the Mixed Radix Conversion (MRC) [23], [29],[27]. The MRC approach is adapted to perform the reverse conversion in this study.

2.1 The Mixed Radix Conversion

An alternative reverse conversion approach to the Chinese Remainder Theorem is the Mixed Radix Conversion which sequentially computes an integer from its residues.

Definition 2.1. Given the moduli set m_i for $i = 1, \ldots, k$, the equivalent decimal number X can be calculated from its residues set $\{x_1, x_2, \ldots, x_k\}$ using the MRC [27]as follows:

$$X = a_1 + a_2 m_1 + a_3 m_1 m_2 + \cdots + a_k m_1 m_2 m_3 \cdots_{(k-1)} \tag{2.3}$$

where $\{a_i\}_{(i=1,n)}$ are the Mixed Radix Digits (MRDs) and are given as follows:

$$\begin{aligned}
a_1 &= x_1 \\
a_2 &= |(x_2 - a_1)|m_1^{-1}|_{m_2}|_{m_2} \\
a_3 &= |\Big(\big((x_3 - a_1)|m_1^{-1}|_{m_3}\big) - a_2\Big)|m_2^{-1}|_{m_3}|_{m_3} \\
&\vdots \\
a_n &= \Big|\Big(\Big(\big((x_n - a_1)|m_1^{-1}|_{m_n}) - a_2\big)|m_2^{-1}|_{m_n}\big)\Big) \cdots - a_{n-1}\Big)|m_{n-1}^{-1}|_{m_n}\Big|_{m_n}
\end{aligned} \tag{2.4}$$

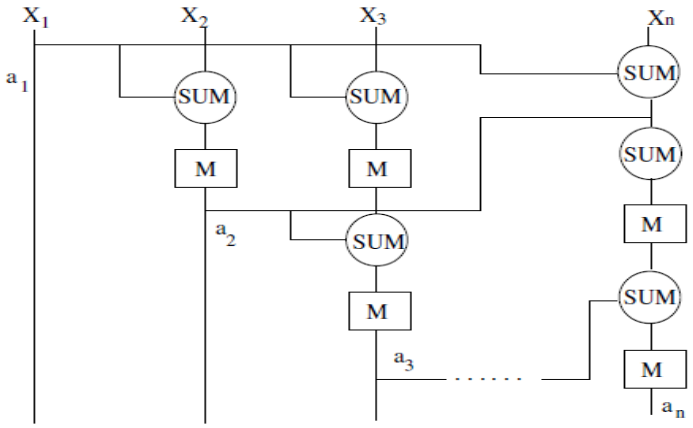

Figure 2: RNS reverse conversion using MRC

3 SOME RELATED WORKS

In the recent past, research has demonstrated the application of RNS to achieve fast signal processing in communication systems and for reliable transmission, data encryption and compression, and cryptographic and stenographic schemes [32],[5],[6] The high degree of computational parallelism and carry-free operations inherent in the system offers a new avenue for energy efficiency in sensor nodes, data security, increased data transfer rate, and better data storage.

Wenyori [31] applied RNS to the Huffman method of a secured data encryption algorithm. The method encrypts data using the moduli set $\{2^n - 1, 2^n, 2^+1\}$ based on the rate of occurrence of

each data. The proposed scheme achieves the required security of data and enhances the speed of transmission of data.

In bioinformatics, a common challenge is achieving accuracy and speed of sequence alignment. A popular technique in dealing with the sequence alignment challenge is via the Smith-Waterman (SW) algorithm. However, the algorithm is computationally costly. Bagyere [9] studied the computational challenge of the SW algorithm using the inherent properties of RNS. The hardware implementation of the RNS-based Smith-Waterman algorithm is done using a VHDL. Experimental results show that the approach used by the author achieves desired reduction in computational cost and sequencing time.

Roshanzadeh and Saqaeeyan [26] gave some insight into the applicability of RNS in WSN. The authors claim to have achieved in their studies reduced traffic rate in a wireless sensor network with a decreased amount of data transmission and, by extension, a reduction in power consumption of sensor nodes. The work in [26] also offers error detection and correction steps for single errors. Authors did not establish multiple error corrections.

Alhassan and Gbolagade [7] suggested an enhancement of the security of a digital image using the Moduli set $\{2^n-1, 2^n, 2^{n+1}\}$. Pixel scrambling is fused with RNS encoding to enhance the security of cryptographic systems. The advantages of their work include: the requirement of fewer bits to represent pixels; resistance to statistical attacks; only user discretion is required to specify decryption iterations; and it is highly sensitive.

Perceptual video encryption technique is presented in [8]. A cipher video is encrypted using the moduli set $\{2^n - 1, 2^n, 2^{n+1}\}$. into three residual videos of smaller pixel values. Their scheme reduces the number of transmissions by using only two of the three transmission channels. The approach provides enhanced transmission speed and security of cipher video during transmission over communication networks.

Agbedemnab et al. [5] exhibited a new image encryption and decryption technique using a genetic algorithm and residue number system. In order to achieve text encryption and decryption, genetic algorithm operators (cross over and mutation) are combined with RNS in a three-layered arrangement. First, forward conversion is done to encode a Unicode or ASCII value into RNS. Next, crossover and mutation operations are executed on the encoded values to encrypt them. Finally, to decrypt the encrypted data RNS to decimal number conversion is done. Simulation results show that their scheme is chaotic by sight, robust and generates a better throughput rate. Their work is also able to encrypt both smaller and larger messages.

In [3], the residue number system is applied to achieve rain fading mitigation on the KU Band satellite communication link. The emphasis of their work was to reduce the bit energy, which is an essential factor in classifying the performance of a communication link. Their work performs better compared with traditional rain fading mitigation methods.

4 METHODOLOGY

This section presents the proposed RNS-based data splitting scheme and the proposed forward and reverse converters to realise implementation in a wireless sensor network.

4.1 Forward converter for the moduli set $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$

The integer representation of messages and attribute values, often in decimal or binary form, is represented in RNS by performing forward conversion. The forward converter for the moduli set is presented here. The selected moduli set is enough to represent a legitimate binary/decimal number of width $(6n+2)$-bits. Such a number has three sub-blocks of $(2n+2)$-bits, $(2n+1)$-bits and $(2n)$-bits wide respectively.

Given the moduli set $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$, $\forall n > 1$ integer X of width $6n+2$, can be represented as follows;

$$X \longrightarrow \underbrace{X_{6n+1}\ldots X_{4n+3}X_{4n+2}}_{B_3,(2n)-bits} | \underbrace{X_{4n+1}\ldots X_{2n+2}X_{2n+1}}_{B_2,(2n+1)-bits} | \underbrace{X_{2n}\ldots X_1X_0}_{B_1,(2n+1)-bits} \tag{4.1}$$

Where, B_1, B_2 and B_3 are binary numbers given as:

$$\begin{aligned} B_1 &= \sum_{i=0}^{2n} x_i 2^i, \\ B_2 &= \sum_{i=2n+1}^{4n+1} x_i 2^{i-2n+1}, \\ B_3 &= \sum_{i=4n+2}^{6n+1} x_i 2^{i-4n+2}. \end{aligned} \tag{4.2}$$

Thus X can be computed as;

$$X = B_1 + 2^{2n+1}B_2 + 2^{4n+2}B_3 \tag{4.3}$$

The residues of X w.r.t to the moduli set $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$, can be derived as follows:

$$\begin{aligned} x_1 &= |X|_{2^{2n+1}}, \\ &= ||B_1|_{2^{(2n+1)}} + |2^{2n+1}B_2|_{2^{2n+1}} + |2^{4n+2}B_3|_{2^{2n+1}}|_{2^{2n+1}}, \\ &= B_1. \end{aligned} \tag{4.4}$$

$$\begin{aligned} x_2 &= |X|_{2^{2n+1}-1}, \\ &= ||B_1|_{2^{2n+1}} + |2^{2n+1}B_2|_{2^{2n+1}-1} + |2^{4n+2}B_3|_{2^{2n+1}-1}|_{2^{2n+1}-1}, \\ &= |B_1 + B_2 + B_3|_{2^{2n+1}-1}. \end{aligned} \tag{4.5}$$

$$\begin{aligned} x_3 &= |X|_{2^{2n}-1}, \\ &= ||B_1|_{2^{2n}-1} + |2^{2n+1}B_2|_{2^{2n}-1} + |2^{4n+2}B_3|_{2^{2n}-1}|_{2^{2n}-1}, \\ &= |B_1 + 2B_2 + 2^2B_3|_{2^{2n}-1}. \end{aligned} \tag{4.6}$$

Following from above, the proposed forward converter for the chosen moduli set - $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$ - is shown in Figure 3. It is made up of two carry save adders (CSA) of $(2n+1)$-bits and $2n$-bits wide respectively and two carry propagate adders (CPA) of $(2n+1)$-bits and $2n$-bits wide respectively.

4.2 Reverse converter for the moduli set $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$

The validity of a given RNS moduli set is premised on the requirement that its members are co-prime. We thus will test that the elements of the chosen moduli set are pair-wise relatively prime and can be used to design an associated reverse converter.

The moduli set $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$, is made up of pair-wise relatively primes.

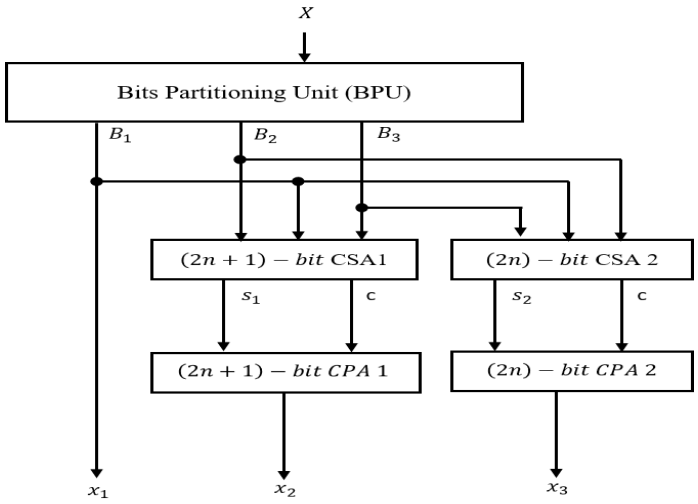

Figure 3: Proposed forward converter for the moduli set $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$.

Proof. The proof of co-prime is done using the Euclidean theorem, which is expressed as:

$$gcd(a,b) = gcd(b, |a|_b).$$

Considering moduli m_1 and m_2, it follows that,

$$\begin{aligned} gcd(2^{2n+1}, 2^{2n+1}-1) &= gcd(2^{2n+1}-1, |2^{2n+1}|_{2^{2n+1}-1}), \\ &= gcd(2^{2n+1}-1, 1), \\ &= 1. \end{aligned}$$

Next considering moduli m_1 and m_3, it follows that

$$\begin{aligned} gcd(2^{2n+1}, 2^{2n}-1) &= gcd(2^{2n}-1, |2^{2n+1}|_{2^{2n}-1}), \\ &= gcd(2^{2n}-1, 2), \\ &= 1. \end{aligned}$$

For the moduli m_2 and m_3, the co-primality proof appears in [10].
This completes the proof. □

Let $\{m_1, m_2, m_3\}$ be an RNS moduli set such that $m_1 = 2^{2n+1}$, $m_2 = 2^{2n+1}-1$, and $m_3 = 2^{2n}-1$, $\forall\ n \geq 1$. Then,

$$|m_1^{-1}|_{m_2} = 1 \tag{4.7}$$

$$|m_1^{-1}|_{m_3} = 2^{2n-1} \tag{4.8}$$

$$|m_2^{-1}|_{m_3} = 1 \tag{4.9}$$

hold.

Proof. Let k be the multiplicative inverse of $|m_1|_{m_2}$ such that

$$k = |(2^{2n+1})^{-1}|_{2^{2n+1}-1}$$

which can be rewritten as;

$$|k(2^{2n+1})|_{2^{2n+1}-1} = 1$$

since

$$|(2^{2n+1})|_{2^{2n+1}-1} = 1$$

It follows that

$$|k(2^{2n+1})|_{2^{2n+1}-1} = |k(1)|_{2^{2n+1}-1} = 1.$$

Since the multiplicative inverse of 1 is 1, it implies that k = 1.

Similarly let k be the multiplicative inverse of $|m_1|_{m_3}$ such that

$$k = |(2^{2n+1})^{-1}|_{2^{2n}-1}$$

which can be rewritten as;

$$|k(2^{2n+1})|_{2^{2n}-1} = 1$$

but

$$\begin{aligned} |2^{2n+1}|_{2^{2n}-1} &= |2(2^{2n}-1)+2|_{2^{2n}-1} \\ &= 2 \end{aligned}$$

It follows that

$$|k(2^{2n+1})|_{2^{2n}-1} = |k(2)|_{2^{2n}-1} = 1$$

solving above for k, we have

$$\begin{aligned} k &= |2^{-1}|_{2^{2n}-1} \\ &= |(2^{-1})2^{2n}2^{-2n}|_{2^{2n}-1}, \\ &= |(2^{2n-1})(1)|_{2^{2n}-1}, \\ &= 2^{2n-1}. \end{aligned}$$

Finally let k be the multiplicative inverse of $|m_2|_{m_3}$ such that

$$k = |(2^{2n+1}-1)^{-1}|_{2^{2n}-1}$$

which can be rewritten as;

$$|k(2^{2n+1}-1)|_{2^{2n}-1} = 1$$

but

$$\begin{aligned} |2^{2n+1}-1|_{2^{2n}-1} &= |2(2^{2n}-1)+1|_{2^{2n}-1} \\ &= 1 \end{aligned}$$

It follows that

$$|k(2^{2n+1})|_{2^{2n}-1} = |k(1)|_{2^{2n}-1} = 1$$

solving above for k, we have

$$k = 1.$$

These complete the proof. □

4.2.1 Some useful properties

- **Property 1.** Modulo multiplication of a residue number by 2^k in modulo $(2^n - 1)$, the result is computed by performing k-bits left rotation of the given number. Given $\langle 2^2(10)\rangle_3 \xrightarrow{\text{Binary}}$ $\langle 2^2(1010)\rangle_3$=$\langle 1010\rangle_3 \xrightarrow{\text{Decimal}} 1$.
- **Property 2.** Modulo $(2^p - 1)$ of a negative number is computed by subtracting the specified number from $(2^p - 1)$. Such an operation in binary form is equivalent to performing a one's complement of the number. Given$\langle -10\rangle_3 \xrightarrow{\text{Binary}} \langle -(1010)\rangle_3 = \langle 0101\rangle_3 \xrightarrow{\text{Decimal}} 2$.
- **Property 3.** The sum of a residue number x_a and $2^n x_b$ is computed as x_b concatenation x_a, if and only if x_a is an $n - bit$ wide number.

Let $\{m_1, m_2\}$ be a moduli set such that $m_1 = 2^{2n+1}$, and $m_2 = 2^{n+1} - 1$, and $\{x_1, x_2\}$ the residue subset with respect to the moduli set $\{m_1, m_2\}$. Then $\forall n \geq 1$, the decimal equivalent of $\{x_1, x_2\}$ is derived as follows:

$$X_p = x_1 + 2^{2n+1}\beta \tag{4.10}$$

where

$$\beta = \Big|(x_2 - x_1)\Big|_{2^{2n+1}-1} \tag{4.11}$$

holds.

Proof. Applying Equation (4.7) to (2.4) for the moduli set $\{m_1, m_2\}$ yields

$$\begin{aligned} X_p &= x_1 + 2^{2n+1}\Big|(x_2 - x_1)(1)\Big|_{2^{2n+1}-1} \\ &= x_1 + 2^{2n+1}\Big|(x_2 - x_1)\Big|_{2^{2n+1}-1} \end{aligned} \tag{4.12}$$

Substituting Equation (4.11) into (4.12) yields

$$X_p = x_1 + 2^{2n+1}\beta$$

□

This completes the proof Let $\{m_1 m_2, m_3\}$ be a moduli set such that $m_1 m_2 = (2^{2n+1})(2^{2n+1} - 1)$ and $m_3 = 2^{2n} - 1$. Then for the residue subset $\{X_P, x_3\}$ with respect to the set $\{m_1 m_2, m_3\}$, its decimal equivalent is derived as follows:

$$X = X_p + 2^{4n+2}\gamma - 2^{2n+1}\gamma \tag{4.13}$$

where

$$\gamma = \Big|(x_3 - X_p)(2^{2n-1})\Big|_{2^{2n}-1} \tag{4.14}$$

holds.

Proof. Applying Equation (4.7) to (2.4) for the moduli set $\{m_1 m_2, m_3\}$ yields

$$\begin{aligned} X &= X_p + 2^{2n+1}(2^{2n+1} - 1)\Big|(x_3 - X_p)(2^{2n-1})\Big|_{2^{2n}-1} \\ &= X_p + 2^{4n+2}\Big|(x_3 - X_p)(2^{2n-1})\Big|_{2^{2n}-1} - 2^{2n+1}\Big|(x_3 - X_p)(2^{2n-1})\Big|_{2^{2n}-1} \end{aligned} \tag{4.15}$$

Substituting Equation (4.14) into (4.15) yields

$$X = X_p + 2^{4n+2}\gamma - 2^{2n+1}\gamma$$

□

This completes the proof

Equation (4.13) will be simplified and implemented using the properties above.

Considering the moduli set $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$, the residues $\{x_1, x_2, x_3\}$ have the following binary representation.

$$x_1 = (\underbrace{x_{1,2n}x_{1,2n-1}\ldots x_{1,1}x_{1,0}}_{(2n+1)-bits}) \tag{4.16}$$

$$x_2 = (\underbrace{x_{2,2n}x_{2,2n-1}\ldots x_{2,1}x_{2,0}}_{(2n+1)-bits}) \tag{4.17}$$

$$x_3 = (\underbrace{x_{3,2n-1}x_{3,2n-2}\ldots x_{3,1}x_{3,0}}_{2n-bits}) \tag{4.18}$$

Equation (4.11) can be simplified as

$$\beta = \beta_1 + \beta_2 \tag{4.19}$$

where

$$\beta_1 = \Big|x_2\Big|_{2^{2n+1}-1} \tag{4.20}$$

and

$$\beta_2 = \Big|-(x_1)\Big|_{2^{2n+1}-1} \tag{4.21}$$

From Equation (4.20)

$$\begin{aligned}\beta_1 &= \Big|x_2\Big|_{2^{2n+1}-1} \\ &= \Big|(x_{2,2n}x_{1,2n-1}\ldots x_{1,1}x_{1,0})\Big|_{2^{2n+1}-1}\end{aligned} \tag{4.22}$$

The binary representation of β_1 is therefore specified as

$$\beta_1 = \underbrace{\beta_{1,2n}\beta_{1,2n-1}\ldots\beta_{1,1}\beta_{1,0}}_{(2n+1)-bits} \tag{4.23}$$

From Equation (4.21)

$$\begin{aligned}\beta_2 &= \Big|-(x_1)\Big|_{2^{2n+1}-1} \\ &= \Big|\bar{x_1}\Big|_{2^{2n+1}-1} \\ &= \Big|(\bar{x}_{1,2n}\bar{x}_{1,2n-1}\ldots\bar{x}_{1,1}\bar{x}_{1,0})\Big|_{2^{2n+1}-1}\end{aligned} \tag{4.24}$$

The binary representation of β_2 is therefore specified as

$$\beta_2 = \underbrace{\beta_{2,2n}\beta_{2,2n-1}\ldots\beta_{2,1}\beta_{2,0}}_{(2n+1)-bits} \tag{4.25}$$

Substituting (4.23) and (4.25) into Equation (4.10) gives:

$$\begin{aligned}X_p &= x_1 + 2^{2n+1}\beta \\ &= x_1 + 2^{2n+1}(\beta_{1,2n}\beta_{1,2n-1}\ldots\beta_{1,1}\beta_{1,0} + \beta_{2,2n}\beta_{2,2n-1}\ldots\beta_{2,1}\beta_{2,0}) \\ &= x_1 + 2^{2n+1}\beta_{2n}\beta_{2n-1}\ldots\beta_1\beta_0\end{aligned} \tag{4.26}$$

Applying property (3) to (4.26) , X_p can be expressed as an $4n+2$ bit number as follows:

$$\begin{aligned} X_p &= \beta_{2n}\beta_{2n-1}\dots\beta_1\beta_0 \otimes x_{1,2n}x_{1,2n-1}\dots x_{1,1}x_{1,0} \\ &= X_{p,4n+1}X_{p,4n}\dots X_{p,1}X_{p,0} \end{aligned} \tag{4.27}$$

where, $\otimes$ is a concatenation operator.
Equation (4.14) can be simplified as

$$\gamma = \gamma_1 + \gamma_2 \tag{4.28}$$

where

$$\gamma_1 = \left|2^{2n-1}x_3\right|_{2^{2n}-1} \tag{4.29}$$

$$\gamma_2 = \left| -(2^{2n-1}X_p)\right|_{2^{2n}-1} \tag{4.30}$$

From Equation (4.29)

$$\begin{aligned} \gamma_1 &= \left|2^{2n-1}x_3\right|_{2^{2n}-1} \\ &= \left|(x_{3,0}x_{3,2n-1}\dots x_{3,1})\right|_{2^{2n}-1} \end{aligned} \tag{4.31}$$

The binary representation of γ_1 is therefore specified as:

$$\gamma_1 = \underbrace{\gamma_{1,2n-1}\gamma_{1,2n-2}\dots\gamma_{1,1}\gamma_{1,0}}_{(2n)-bits} \tag{4.32}$$

From Equation (4.30):

$$\begin{aligned} \gamma_2 = \left| -(2^{2n-1}X_p)\right|_{2^{2n}-1}, &= \left|2^{2n-1}\bar{X}_p\right|_{2^{2n}-1} \\ &= \left|2^{2n-1}(\bar{0}\bar{0}\dots\bar{0}\bar{0}\bar{X}_{p,4n+1}\bar{X}_{p,4n}\dots\bar{X}_{p,1}\bar{X}_{p,0})\right|_{2^{2n}-1}, \\ &= \left|2^{2n-1}(11\dots11\bar{X}_{p,4n+1}\bar{X}_{p,4n}\dots\bar{X}_{p,1}\bar{X}_{p,0})\right|_{2^{2n}-1}, \\ &= \left|\gamma_2^1 + \gamma_2^{11} + \gamma_2^{111}\right|. \end{aligned} \tag{4.33}$$

where

$$\begin{aligned} \gamma_2^1 &= \left|(2^{2n-1}(11\dots11\bar{X}_{p,4n+1}\bar{X}_{p,4n})\right|_{2^{2n}-1}, \\ &= \bar{X}_{p,4n}11\dots11\bar{X}_{p,4n+1}. \end{aligned} \tag{4.34}$$

$$\begin{aligned} \gamma_2^{11} &= \left|2^{2n-1}(\bar{X}_{p,4n-1}\bar{X}_{p,4n-2}\dots\bar{X}_{p,2n})\right|_{2^{2n}-1}, \\ &= \bar{X}_{p,2n}\bar{X}_{p,4n-1}\dots\bar{X}_{p,2n+1}. \end{aligned} \tag{4.35}$$

and

$$\begin{aligned} \gamma_2^{111} &= \left|2^{2n-1}(\bar{X}_{p,2n-1}\bar{X}_{p,2n-2}\dots\bar{X}_{p,0})\right|_{2^{2n}-1}, \\ &= \bar{X}_{p,0}\bar{X}_{p,2n-1}\dots\bar{X}_{p,1}. \end{aligned} \tag{4.36}$$

Substituting (4.34), (4.35) and (4.36) into (4.33) yields the binary representation of γ_2 as

$$\gamma_2 = \underbrace{\gamma_{2,2n-1}\gamma_{2,2n-2}\ldots\gamma_{2,1}\gamma_{2,0}}_{(2n)-bits} \tag{4.37}$$

From the preceding, (4.13) can be simplified as

$$X = X_p + 2^{4n+2}\gamma - 2^{2n+1}\gamma = X_1 + X_2 \tag{4.38}$$

where

$$\begin{aligned} X_1 &= X_p + 2^{4n+2}\gamma \\ &= \underbrace{\gamma_{2n-1}\gamma_{2n-2}\ldots\gamma_1\gamma_0}_{(2n)-bits}\underbrace{X_{p,4n+1}X_{p,4n}\ldots X_{p,1}X_{p,0}}_{(4n+2)-bits} \end{aligned} \tag{4.39}$$

$$\begin{aligned} X_2 &= -(2^{2n+1})\gamma \\ &= 2^{2n+1}\bar{\gamma} \\ &= \underbrace{11\ldots11}_{(2n+1)-bits}\ \underbrace{\bar{\gamma}_{2n-1}\bar{\gamma}_{2n-2}\ldots\bar{\gamma}_1\bar{\gamma}_0}_{(2n)-bits}\ \underbrace{11\ldots11}_{(2n+1)-bits} \end{aligned} \tag{4.40}$$

4.3 Hardware realisation for the proposed reverse converter

The hardware realization for the proposed reverse and forward converters work for values of $n \geq 1$. The underlying hardware design for the reverse converter is based on Equations (4.10) and (4.38). At the outset, an operand preparation unit (OPU1) is used to manipulate residues x_1, x_2 and x_3 to produce β_1 and β_2, given in Equation 4.20 and 4.21 respectively. These are then added using $2n+1-$bit carry propagate adder (CPA) 1 to produce β via modulo addition. The result is then used to obtain X_p via a simple concatenation operation which requires no further hardware resource. Next OPU 2 reduces X_p into γ_1, γ_2^1, γ_2^{11} and γ_2^{111}. The binary numbers, γ_1, γ_2^1, γ_2^{11} are added using a $2n$ bit CSA 1 to produce sum s_1 and carry c_1. Another 2n bit CSA 2 is then used to produce s_2 and c_2 from the modulo addition of s_1, c_1 and γ_2^{111}. Carry propagate adder CPA 2, a 2n-bit adder computes γ from s_2 and c_2. Equation (4.38), is then derived for X by performing modulo addition of X_1 and X_2 both of which are of length 6n+2 bit using a CPA 3.

The proposed conversion from residue to binary as depicted in Figure 4 requires the use of carry-save adders and carry-propagate adders. The hardware requirement of the proposed reverse converter in area terms is $(14n+3)\ \triangle_{FA}$. The estimated delay of the reverse converter is $(20n+8)\ \triangle_{FA}$.

4.4 Proposed RNS-based data splitting scheme

In the method proposed here, the residue number system inherent property of parallelism is used to transform WSN data into smaller parts for transmission. The subparts are derived relative to a set of chosen coprime numbers whose product must be greater than the data to have a legitimately working system. Cheap energy cost routes are computed to deliver data from source to destination nodes by considering links that expend the least energy. MATLAB 7.5.0 (R2007b) is used to implement the data splitting method proposed in this research. The proposed RNS based data splitting and transmission protocol here is obtained by modifying the energy model in [19] by fitting in an RNS encoder and decoder layer as in Figure 5.

The idea is to decompose an original sense attribute value (such as text or image, or video) into smaller parts at the sender node. Splitting is done with an RNS encoder (forward converter) built into the senor node architect and placed after the analogue to digital converter (ADC). Bitstreams of

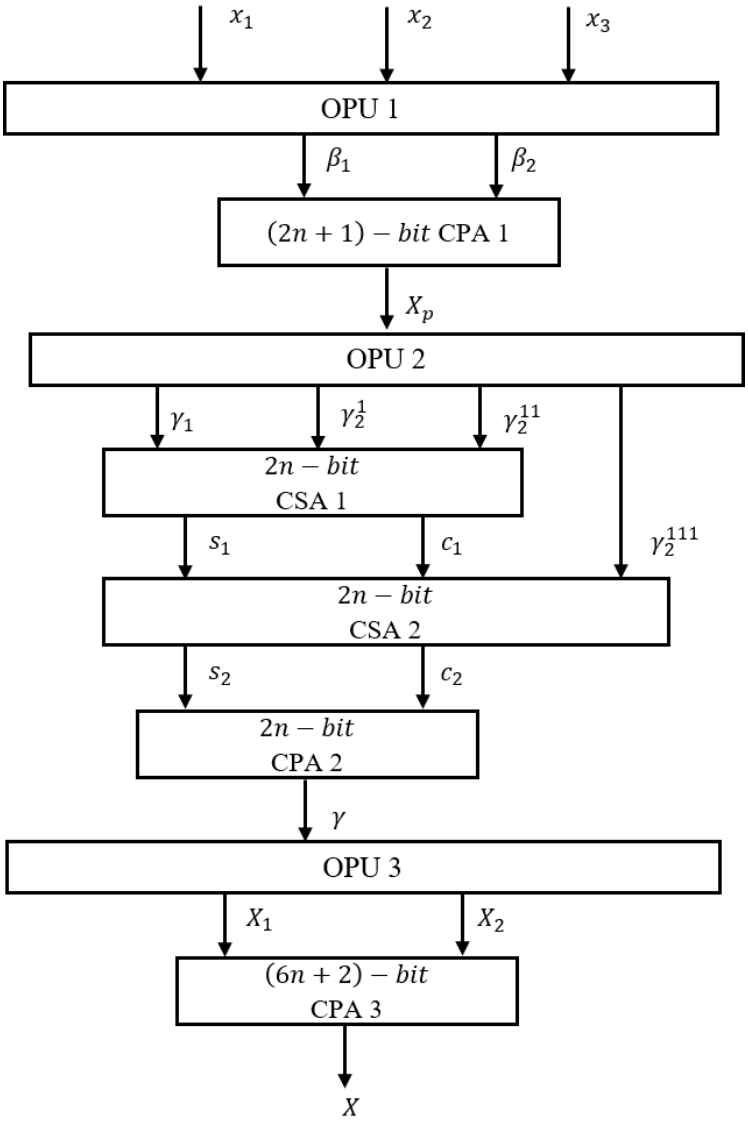

Figure 4: Proposed reverse converter for the moduli set $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$

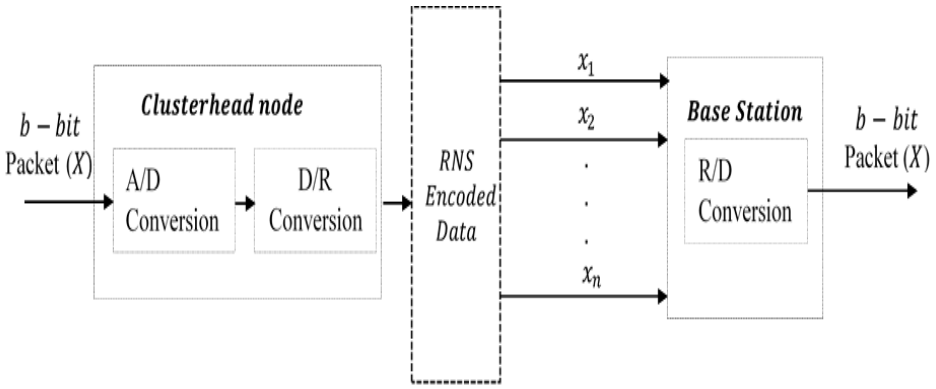

Figure 5: A block diagram for the proposed data routing scheme base on RNS.

the set of reduced attribute values are then passed from the forward converter of the sending sensor node to the receiving node via non-intersecting minimum transmission energy routes. The received bitstreams of the reduced attribute values are manipulated using the proposed reverse converter to recover the initially sensed data at the receiver node. Algorithms I and II presents the splitting and decoding steps of the proposed approach.

ALGORITHM I: RNS-BASED DATA SPLITTING

INPUT: message X, $\{m_i\}_{i=1,\ldots,l}$, number of bits n, channel width W_{m_i}
OUTPUT: residual packets $\{x_1, x_2, x_3\}$

1. Cluster head receives and aggregates data from cluster members.
2. The Cluster head then selects a set of relatively prime moduli, $m_1 = 2^{2n+1}$, $m_2 = 2^{2n+1} - 1$, $m_3 = 2^{2n} - 1$, such that $m_1 > m_2 > m_3$
3. The Cluster head computes actual moduli set $\{2^{2n+1}, 2^{2n+1} - 1, 2^{2n} - 1\}$, $n \geq 1$
4. The Cluster head determines the width of each moduli channel, W_{m_i}
5. The Cluster head computes the residual packets $\{x_1, x_2, x_3\}$ from X (aggregated message at cluster head) with respect to the moduli set $\{2^{2n+1}, 2^{2n+1} - 1, 2^{2n} - 1\}$
6. The Cluster head determines l non-intersecting minimum transmission energy route to the base station.
7. Finally, the Cluster head transmits residual packets $\{x_1, x_2, x_3\}$ via l MTE routes to the BS.
8. Steps 5 to 7 are repeated for all transmissions in a cluster while network is active.

ALGORITHM II: RNS MESSAGE DECODING AT THE BS

INPUT: residual packets $\{x_1, x_2, x_3\}$, $\{m_i\}_{i=1,\ldots,l}$
OUTPUT: message X

1. The base station receives and orders the residual packets $\{x_1, x_2, x_3\}$.
2. The base station then computes the moduli set $\{2^{2n+1}, 2^{2n+1} - 1, 2^{2n} - 1\}$ for $n \geq 1$
3. The base station decodes the residual packets $\{x_1, x_2, x_3\}$ to obtain the original message X using Equation (4.13)
4. The base station return original message X

5 ANALYSIS AND RESULTS

In this section we present and discuss the results from the performance analysis of the proposed data splitting scheme.

5.1 Illustrative example of proposed forward conversion technique

Given the information moduli set, $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$ and a sensed attribute value (given in integer) X = 8923, the forward conversion process is as follows; Let n = 2; consequently, the moduli set is reduced to $\{32, 31, 15\}$.
Next 8923 is converted into binary representation which gives 10001011011011_2. Since n = 2 and X is a $(6n+2)$-bits wide number, we partition X into three blocks: $B_1 = 11011$, $B_2 = 10110$ and $B_3 = 1000$.

From Equation (4.4)

$$\begin{aligned} x_1 &= B_1, \\ &= 11011, \\ &= 27. \end{aligned}$$

From Equation (4.5)

$$\begin{aligned} x_2 &= |B_1 + B_2 + B_3|_{2^{2n+1}-1}, \\ &= |(11011)_2 + (10110)_2 + (1000)_2|_{31}, \\ &= |27 + 22 + 8|_{31}, \\ &= 26. \end{aligned}$$

From Equation (4.6)

$$\begin{aligned} x_3 &= |B_1 + 2B_2 + 2^2 B_3|_{2^{2n}-1}, \\ &= |(11011)_2 + 2(10110)_2 + 2^2(1000)_2|_{15}, \\ &= |27 + 2(22) + 4(8)|_{15}, \\ &= |27 + 44 + 32|_{15}, \\ &= 13. \end{aligned}$$

Therefore, the residues of 8923 with respect to the moduli set $\{32, 31, 15\}$ is $\{27, 26, 13\}$
End of example.

5.2 Performance analysis of proposed RNS-based data splitting scheme

The analysis compares the energy cost in transmitting data from a cluster head to a base station in a conventional transmission approach and the proposed method in this study.

In the RNS-based splitting scheme for data transmission, the moduli set, $\{2^{2n+1}, 2^{2n+1}-1, 2^{2n}-1\}$ which has three channels will be used to split the sensed message into three binary residual parts using Equation (4.4 - 4.6). The residual parts are sent individually.

The energy cost required to transmit a b bit of aggregated data from a cluster head to the base station using the proposed RNS-based scheme is as in Equation 5.1.

$$
\begin{aligned}
E_{RNS} &= E(Rx) + E(DA) + E(Tx), \\
&= W_b \cdot \Big(\frac{N}{k} - 1\Big) E_{elec} + W_b \cdot N \cdot E_{DA} + W_{m_i} \cdot (E_{elec} + \varepsilon_{fs} \cdot r^2_{CHtoBS}), \\
&= W_b \cdot \Big(\frac{N}{k} - 1\Big) E_{elec} + W_b \cdot N \cdot E_{DA} + W_{m_i} \cdot E_{elec} + W_{m_i} \cdot \varepsilon_{fs} \cdot r^2_{CHtoBS}.
\end{aligned} \tag{5.1}
$$

where
E_{Tx} is the energy required to run the transmitter electronic,
E_{DA} is the energy required to aggregate data,
E_{Rx} is the energy required to run the receiver electronic,
r_{CHtoBS} is the distance between a cluster head and the base station,
W_b is the width of the the transmitted message,
W_{m_i} is the width of the ith channel.

Table 1: Network characteristics to be used for analysis

Operation	Energy requirement
Transmitter/Receiver Electronics	$E_{elec} = 50nJ/bit$
Data Aggregation	$E_{DA} = 5nJ/bit/packet$
Transmit amplifier of $r \leq r_o$	$\varepsilon_{fs} = 10pJ/bit/m^2$
Transmit amplifier of $r > r_o$	$\varepsilon_{mp} = 0.0013pJ/bit/m^4$
Initial energy of normal nodes	$0.5J$

5.3 Energy efficiency analysis

We simulate the energy required to transmit the residues of an aggregated packet and compare it with the energy required to transmit the original non-split packet. The widest channel of the moduli set, $\{2^{2n+1}, 2^{2n+1} - 1, 2^{2n} - 1\}$, is channel 1, which is $2n + 2$ bit wide. Channel 2 and 3 have width $(2n + 1)$-bits and $(2n)$-bits wide respectively. The message X itself is $(6n + 2) - bit$ wide. So we use the worst-case scenario in our analysis and comparison. Figure 6 and Figure 7 graphically depict the energy dissipation rate and the energy reserved in both methods.

It is instructively clear that using values of $n \geq 1$ (even values of n are used for illustrations); the proposed scheme betters the traditional method regarding reduction in transmission energy. On average, the energy dissipation in the conventional approach is $0.00020918j$ per transmitted data, compared with $0.00000304529j$ in the proposed RNS-based method. Consequently, residual energy after data transmission is more in the RNS-based scheme. This improved energy reserve means that data gathering and transfer will be done in an extended period than would have traditionally been allowed.

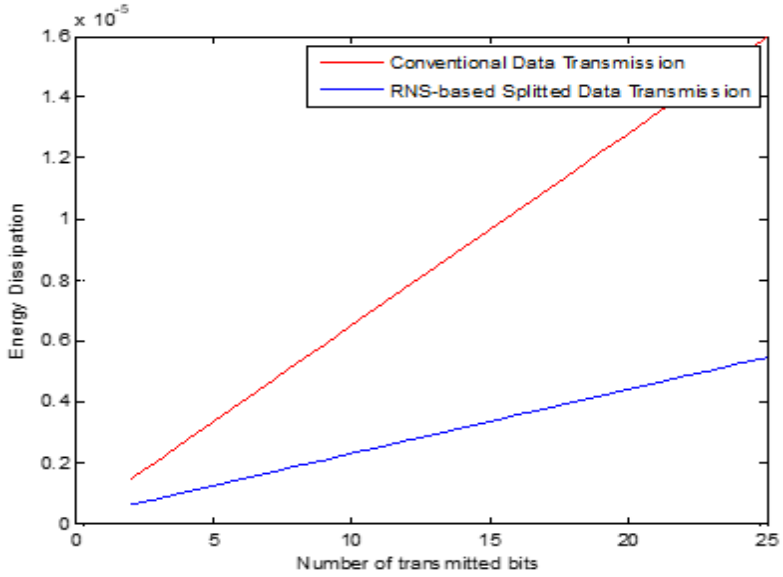

Figure 6: Energy dissipation per n bit of transmitted message

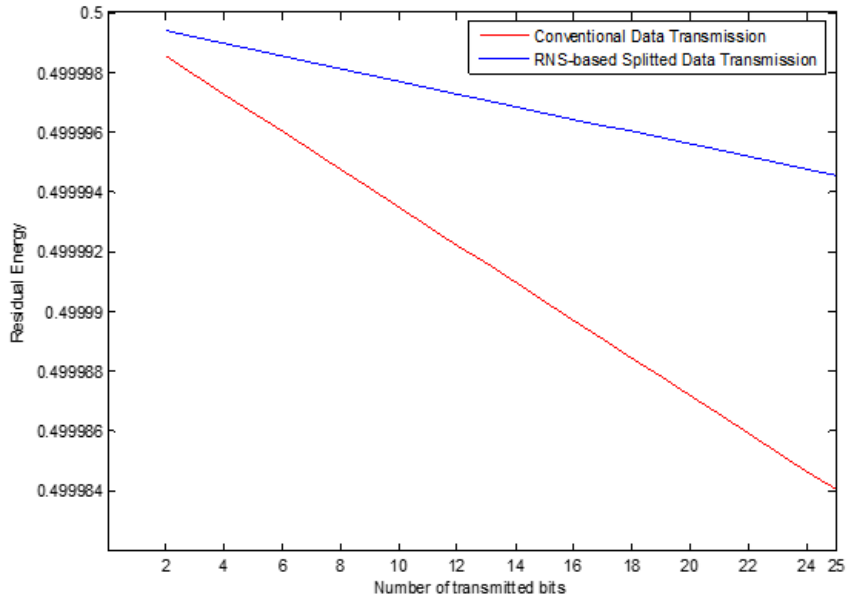

Figure 7: Residual energy per n bit of transmitted message

6 CONCLUSIONS

In the monitoring functions of a sensor node, data transmission accounts for most of its energy losses. Energy-saving mechanisms for sensor nodes abound in the literature, with varying degrees of success. This work proposes an enhanced scheme to reduce transmission energy in WSN using suitable number properties of the residue number system. The parallelism feature of RNS is relied on to split and then transmit sensed attribute values to a user station via selected moduli channels. Sensors send out residual values in digital form through routes that minimise the energy consumption of cluster heads and the network at large. Member nodes in a cluster only partake in data sensing and transmission to the cluster heads. Cluster heads perform data encoding and transmission afterwards to a user station for further analysis. The reverse conversion process is performed at the base station on the received residues to reproduce the original message. Consequently, forward and reverse converters are proposed for message encoding and decoding at the cluster head and base station, respectively. The proposed scheme is experimented with to test performance. Several simulation runs reveal improvement in network performance and energy reduction during data routing to the base station. Due to energy limitations, the study does not consider reverse conversion by sensor nodes. This issue will be interesting to investigate in the future.

References

[1] Adu-Manu, K. S., Adam, N., Tapparello, C., Ayatollahi, H., and Heinzelman, W. (2018). Energy-Harvesting Wireless Sensor Networks (EH-WSNs): A Review. *ACM Trans. Sen. Netw. 14*(2). (doi: 10.1145/3183338)

[2] Adu-Manu, K. S., Trapparello, C., Heinzelman, W., Katsriku, F. A., and Abdulai, J. D. (2017). Water quality monitoring using wireless sensor network: Current trends and future research directions. *ACM Transactions on Sensor Networks, 13*(1), 1-41. (doi: 10.1145/3005719)

[3] Akobre, S., Daabo, M. I., and Salifu, A.-M. (2020). Rain Fade Mitigation Technique Using Residue Number System Architecture on KU Band Satellite Communication Link. *Advances in Networks, 7*(2), 59-66. (doi: 10.11648/j.net.20190702.17)

[4] Akyildiz, I. F., Su, W., Sankarasubramaninam, Y., and Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks, 38*(4). (doi: 10.1016/S1389-1286(01)00302-4)

[5] Agbedemnab, P., A.-N., Baagyere, E., Y., & Daabo, M., I. (2019). A New Image Encryption and Decryption Technique using Genetic Algorithm and Residual Numbers. *IEEE AFRICON Conferencee*, 20-31. (doi: 10.1109/AFRICON46755.2019.9133919)

[6] Agbedemnab, P., A.-N, Baagyere, E., Y., and Daabo, M., I. (2020). Single and Multiple Error Detection and Correction using Redundant Residue Number System for Cryptographic and Stenographic Schemes. *Asian Journal of Research in Computer Science 4*(4), 1-14. doi: 10.9734/ajrcos/2019/v4i430123

[7] Alhassan, S. and Gbolagade, K. A. (2013). Enhancement of the Security of a Digital Image using the moduli set$\{2^n-1, 2^n, 2^n+1\}$. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 2*(7).

[8] Alhassan, S., Iddrisu, M. M. and Daabo, M. I. (2019). Improved Perceptual Video Encryption Technique using Residue Number System. *Earthline Journal of Mathematical Sciences, 2*(1). doi: 10.34198/ejms.2119.111125

[9] Baagyere, E. Y. (2011). Application of residue number system to Smith-Waterman algorithm. Kwame Nkrumah University of Science and Technology, *Unpublished masters thesis*.

[10] Bankas, E. K., (2013). Efficient residue to binary converters for some powers of two moduli sets. University for Development Studied, *Unpublished Ph.D thesis*.

[11] Bankas, E. K., and Gbolagade, K. A. (2013). New Efficient FPGA Design of Residue to Binary Converter. *International Journal of VLSI design & Communication Systems (VLSICS), 4*(6). doi:10.5121/vlsic.2013.4601

[12] Beckmann, P. and Musicus, B. (1993). Fast fault-tolerant digital convolution using a polynomial residue number system. *IEEE Transactions on Signal Processing, 41*(7), 2300-2313. doi:10.1109/78.224241

[13] Chalivendra, G., Hanumaiah, V., and Vrudhula, S. (2011). A new balanced 4-moduli set $\{2^k, 2^n-1, 2^n+1, 2^{n+1}-1\}$ and its reverse converter design for efficient FIR filter implementation. GLSVLSI'll, 139-144

[14] Conway, R., and Nelson, J. (1999). Fast converter for 3 moduli RNS using new property of CRT. *IEEE Transactions on Computers, 48*(8), 852 - 860. doi: 10.1109/12.795127

[15] El Khediri, S., Nasri, N., Wei, A., and Kachouri, A. (2014). A New Approach for Clustering in Wireless Sensors Networks Based on LEACH. *Procedia Computer Science, 32*. doi: 10.1016/j.procs.2014.05.551

[16] Estrin, D. Sayeed, A., and Srivastava, M. (2002). Wireless sensor networks. *In Proceedings of the Tutorial at the 8th ACM International Conference on Mobile Computing and Networking (MobiCom'02)*, 255. ACM, New York, NY.

[17] Fernandez, P., Garcia, A., Ramirez, J., and Iloris, A. (2000). Fast RNS-based 2D computation on field-programmable devices. *Proceedings of the IEEE Signal Processing Systems Workshop, LA, USA, 2*,365-373.

[18] Gbolagade, K. A. (2013). An Efficient MRC based RNS-to-Binary Converter for the $\{2^{2n-1}, 2^n, 2^{2n+1}-1\}$ Moduli Set. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 2*(4).

[19] Heinzelman, W. R., and Chandrakasan, A. (2000). Energy efficient communication Protocol for Wireless Microsensor Networks. In: IEEE Computer Society Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS '00), 8, 1-10. doi: 10.1109/hicss.2000.926982

[20] Jenkins, W. and Leon, B. (1977). The use of residue number system in the design of finite impulse response digital filters. *IEEE Trans. on Circuits and Systems, 24*, 191-200.

[21] Keir, Y. A., Cheney, P. W., and M., Tannenbaun, M. (1962). Division and overflow detection in residue number system. *IRE Transactions on Electronic Computers, EC-11*(4), 501-507. (doi:10.1109/TEC.1962.5219389)

[22] Kim, S., Pakzad, S., Culler, D.E., Demmel, D., Fenves, G., Glaser, S., and Turon, M. (2007). Health monitoring of civil infrastructures using wireless sensor networks. *Proceedings of the 6th International Conference on Information Processing in Sensor Networks.*

[23] Omondi, A. and Premkumar, B. (2007). Residue Number Systems: Theory and Implementation. *Imperial College Press*, UK.

[24] Pontarelli, S., Cardarilli, G., Re, M., and Salsano, A. (2010). Optimized implementation of RNS FIR filters based on FPGAS. *J Sign process Syst., 12*.

[25] Qi, W., Liu, W., Xuxun L., Liu, A., Wang, T., Xiong, N. N., and Cai, Z. (2018). Minimizing Delay & Transmission Times with Long Lifetime in Code Dissemination Scheme for High Loss Ratio and Low Duty Cycle Wireless Sensor Networks. *Sensors, 18*, 3516. (doi:10.3390/s18103516)

[26] Roshanzadeh, M., and Saqaeeyan, S. (2012). Error detection & correction in wireless sensor networks by using residue number systems. *International Journal of Computer Network and Information Security,* 4(2), 2935. doi:10.5815/ijcnis

[27] Szabo, N. S., and Tanaka, R. I. (1967). Residue Arithmetic and its Application to Computer Technology. *McGraw-Hill Book Co.*

[28] Tanaka, R. (1962). Modular arithmetic techniques. *Tech. Rep. ASTDR, Lockheed Missiles and Space Co., (2-38-62-1A).*

[29] Taylor, F. J. (1984). Residue arithmetic: A Tutorial with Examples. *Computer, 17*(5), 50-62. (doi:10.1109/MC.1984.1659138)

[30] Toivonen, T., and Heikkila, J. (2006). Video filtering with fermat number theoretic transforms based on residue number systems.*IEEE Trans. on Circuits and Systems for Video Tech, 16*(1), 92-101.

[31] Weyori, B. A., Akobre, S., and Armah, G. K. (2008). Application of RNS to Huffman's method of secured data encryption algorithm. *International Journal of Soft Computing, 4*(5), 197-200.

[32] Yang, L.-L., and Hanzo, L. (2001). Redundant residue number system based error correction codes. *IEEE Transaction on Circuits and Systems I: Regular Papers, 15*.