[1]Lakshmi V S

RSDA: A Reliable and Secure Data Aggregation Scheme for Smart Grid

Abstract: **-** Smart grids are modern power distribution systems that utilize advanced technologies to optimize energy generation, distribution, and consumption. The huge amount of data generated by various sources, such as smart meters, sensors, and other devices, can be aggregated for efficient energy management, supporting demand response etc. However, this data can contain sensitive information about individual's energy usage patterns, behaviour, and preferences, which raises several privacy concerns. Data reliability is another major issue, as the data send from the smart meters to the aggregators may get corrupted due to errors introduced by the communication channel. In this paper, a private-key additive homomorphic encryption scheme based on polar codes is proposed for privacy preserving data aggregation. The proposed scheme not only ensures data confidentiality but also provide data reliability. The security of this scheme relies on the difficulty in decoding the codeword and is achieved by embedding security in the generator matrix and error vector. Security analysis shows that the proposed scheme is resistant against all known attacks against private-key code-based cryptosystems. The performance analysis is evaluated in terms of error performance, communication overhead and computational complexity. The performance comparison with related additive homomorphic encryption scheme shows the efficiency of the proposed scheme.

Keywords: Data Aggregation, smart grid, polar codes, code-based cryptosystem, homomorphic encryption

I. Introduction

Smart Grid (SG) emerged as the next generation of power grid, which combines electrical network and advanced information and communication technology (ICT), to provide more efficient and economic generation, transmission and distribution of electricity [1]. Compared to the traditional power grid which is centralized and unidirectional, smart grid is decentralized and supports bidirectional communication. Smart meters are major components of smart grid, which are two-way communication devices, deployed at consumer's premises to record power consumption periodically. This huge amount of data collected from individual consumers can be utilized by the smart grid controller for various purposes such as dynamically adjusting power supply demand, and detecting failures in the power system in real time. In many cases, the smart grid controller requires only the aggregate of collected data over a geographical area such as sum or mean, instead of the exact power consumption data from each smart meter. Data aggregators such as gateways deployed in different geographic regions can be used for sending the aggregated data to the controller.

Data privacy or confidentiality is a major concern, when highly sensitive data like energy consumption are transmitted to the gateways, especially when processing operations are to be performed on these data [2]. This is due to the fact that the fine-grained energy consumption of different consumers may reveal a lot of information about consumers, e.g., excerpts from the day that there are no individuals in the household, arrival and departure times, or rest periods, which reveal their behaviour patterns. In order to ensure data privacy, local gateways should not be able to access the consumer's data.

Homomorphic encryption schemes can be used for data aggregation since the decryption of the computed ciphertext should be same as the computations performed on plaintext. Encryption schemes with additive homomorphism can support majority of the data aggregation operations as it involves only linear operations. Public key encryption schemes like Paillier possess additive homomorphism and can support smart grid data aggregation [3, 4]. But the computational complexity of Paillier scheme is very high because of the exponentiation operations. Another public key cryptosystem that supports additive homomorphism is the Elliptic curve ElGamal (EC-EG) scheme and several smart grid data aggregation works have been proposed based on this method [5-6]. But in EC-EG scheme, decryption of messages of larger length is practically difficult due to the hardness in solving the elliptic

[1] *Corresponding author: Assistant Professor, ECE Department, Sree Chitra Thirunal College of Engineering, Thiruvananthapuram, Kerala, India. Email id: lakshmivs23@gmail.com

curve discrete logarithmic problem for message retrieval. As smart meters that have restricted computation capabilities need to perform encryption, encryption schemes with additive homomorphism which offers relatively low computational complexity is suitable for these applications. Moreover, data reliability is another issue in smart grid due to the wireless connectivity between smart meters and gateways and is often dealt with the incorporation of error correcting codes. This is the motivation behind the design of error correcting code based encryption scheme which not only provides security during data processing but also offers error correction.

In recent years, various cryptosystems based on different error correcting codes have been proposed. McEliece public key cryptosystem [7] based on Goppa codes was the first code-based cryptosystem. However, it has large computation overhead due to the large key size and block length. The private key version of code-based cryptosystem known as Rao and Nam (RN) scheme [8] was based on simple error correcting codes and reduced the keysize and computational complexity. In this scheme, the security is introduced through hiding the structure of generator matrix and addition of intentional error vectors to the codewords. The error vectors are chosen from the syndrome error table and this makes the scheme susceptible to different chosen plaintext attacks, since the number of error vectors is less [9, 10]. In recent years, various cryptosystems based on different error correcting codes such as low density parity check (LDPC) codes and Polar codes have been proposed [11-13]. Polar codes are better compared to LDPC codes if the data to be transmitted is of shorter length, especially data such as energy consumption in smart grid applications [14]. However, these schemes do not provide data reliability as they do not retain the error correction capability of the underlying code. Also, these schemes cannot be directly extended for the design of additive homomorphic encryption schemes as these are built based on binary codes. To ensure correctness of homomorphic computation, non-binary codes are required. In this work, an additive homomorphic encryption scheme based on non-binary polar codes is proposed. To the best of our knowledge, this is the first work on homomorphic encryption scheme based on Polar codes. The main contributions of this paper can be summarized as

1. An additive homomorphic encryption scheme based on non-binary polar codes which simultaneously provide data confidentiality and reliability is proposed for smart grid data aggregation.
2. The required levels of security can be achieved through decoding problem by hiding the structure of the generator matrix and through addition of intentional error vectors.
3. The design of intentional error vectors based on LFSR is presented that preserves the randomness properties during encrypted domain processing.
4. Detailed cryptanalysis is performed to show that the proposed cryptosystem resists attacks against code-based cryptosystems.
5. The performance evaluation of the proposed cryptosystem is performed in terms of bit error rate performance, computational complexity and communication overhead. The comparison with related works proves that the proposed system outperforms other schemes.

II. SYSTEM MODEL AND THREAT MODEL

A. System Model

The system model of secure smart grid data aggregation scheme is as shown in Fig. 1, which involves three entities: smart meters, aggregator and smart grid controller. Smart meters (SM) will be installed in each user’s house, and has limited storage and computation capabilities. A smart meter is linked to each consumer and it is responsible for sending the encrypted meter reading at specific time intervals (say per day) to the aggregators/ gateways (GW). The encrypted data from all consumers at regular intervals are aggregated by the gateways and send to the smart grid controller for detailed analysis and billing. Therefore, homomorphic encryption schemes which support secure data aggregation is needed. Forward error correction mechanisms are required to handle the transmission channel errors.

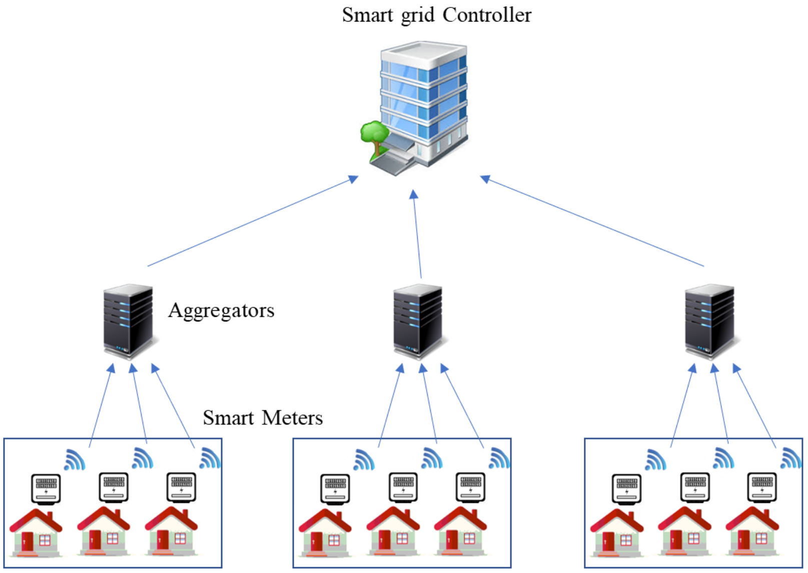

Fig. 1 System Model

B. *Threat Model*

In our threat model, we consider passive adversaries who can eavesdrop on the data transmitted between smart meters and gateways. The goal of the adversary is to breach the security of the data transmitted by smart meters, through launching different attacks such as brute-force attack, known plaintext attack and chosen plaintext attack.

III. Proposed Homomorphic Encryption Scheme based on Polar Codes

A. *Binary Polar Codes*

Polar codes are a new class of capacity achieving linear error correcting codes with low encoding and decoding complexity [15]. The main idea behind polar codes is to transform ordinary channels into extreme channels, i.e., perfect channels with maximum channel capacity and useless channels with minimum channel capacity. This is achieved through channel polarization. The idea behind channel polarization is detailed below.

Let $W: X \to Y$ be a binary discrete memoryless channel (B-DMC), with input alphabet X, output alphabet Y and channel transition probabilities, $W(y|x)$, where $x \in X$ and $y \in Y$. Then $I(W)$ denotes the symmetric capacity of the channel W, which is used as a measure of rate and is given by (1).

$$I(W) \triangleq \sum_{x\in X}\sum_{y\in Y}\frac{1}{2}W(y|x)\,log\frac{W(y|x)}{\frac{1}{2}W(y|0)+\frac{1}{2}W(y|1)} \qquad (1)$$

The measure of reliability can be represented in terms of Bhattacharya parameter, $Z(W)$, which can be expressed as,

$$Z(W) \triangleq \sum_{y\in Y}\sqrt{W(y|0)W(y|1)} \qquad (2)$$

These two parameters of the channels $I(W)$ and $Z(W)$ takes values in [0, 1]. In general, I(W) ≈ 1 iff Z(W) ≈ 0, and $I(W) \approx 0$ iff $Z(W) \approx 1$. If W is a binary erasure channel with erasure probability ε, then $Z(W) = \varepsilon$ and $I(W) = 1 - Z(W)) = 1 - \varepsilon$.

Channel polarization is the process by which a set of polarized channels $\left\{W_N^{(i)}: 1 \le i \le N\right\}$are obtained from N independent copies of a given B-DMC W, where $N = 2^n, n \ge 0$. It is shown in that as N becomes large, the capacity, $I\left(W_N^{(i)}\right)$ and Bhattacharya parameter, $Z\left(W_N^{(i)}\right)$ of polarized sub-channels, except for a fraction of them, approaches towards 0 or 1. Channel polarization includes two phase, a vector channel, $W_N: X^N \to Y^N$ is produced by combining N copies of B-DMC W in a recursive manner. This is followed by channel splitting phase, which splits W_N back into N binary-input co-ordinate channels, $W_N^{(i)}: X \to Y^N \times X^{i-1}, 1 \le i \le N$. The basic idea of polar coding is that, through channel polarization, we can create a channel coding scheme, where one can access each coordinate channel $W_N^{(i)}$ individually and send data only through those for which $Z_N^{(i)}$is near 0.

1) Encoding

The channel combining operation is fully decided by the transformation kernel, *F*, which is as shown in (3).

$$F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \qquad (3)$$

Then the N × N encoding matrix, G_N for generating polar code of size, N can be obtained recursively from *F* as shown in (4).

$$G_N = B_N F^{\otimes n} \qquad (4)$$

where ⊗ denotes the kronecker power of a matrix and B_N is the *N* × *N* bit reversal permutation matrix, which can be computed recursively using *N* × *N* reverse shuffle permutation matrix R_N as shown in (5), with $B_2 = I_2$, where I_2 is the identity matrix.

$$B_N = R_N \begin{bmatrix} B_{N/2} & 0 \\ 0 & B_{N/2} \end{bmatrix} = R_N(I_2 \otimes B_{N/2}) \qquad (5)$$

where N × N reverse shuffle permutation matrix R_N is defined as

$$(s_1, s_2, ..., s_N)R_N = (s_1, s_3, ..., s_{N-1}, s_2, s_4, ..., s_N) \qquad (6)$$

Now, in order to construct an (N, K) polar code, we have to identify *K* channels with lowest Bhattacharya parameter, $Z(W_N^{(i)}), 1 \leq i \leq K$. For BEC, $Z(W_N^{(i)}), 1 \leq i \leq N$ for all *N* channels can be computed using (7).

$$Z(W_N^{(2l-1)}) = Z(W_{N/2}^{(l)})^2$$
$$Z(W_N^{(2l)}) = 2Z(W_{N/2}^{(l)}) - Z(W_{N/2}^{(l)})^2 \qquad (7)$$

where $1 \leq l \leq n$ and $N = 2^n$. A permutation $\pi_N = (i_1, i_2, \dots, i_N)$ is to be formed using the indices of the channel (1, 2, ..., *N*), in such a way that $Z\left(W_N^{(i_j)}\right) \leq Z\left(W_N^{(i_k)}\right)$, where $1 \leq j < k \leq N$. Ths implies that i_1 and i_N are the indices of the channel with minimum and maximum Bhattacharya parameter respectively.

Now, let A be a K element subset of 1, 2, ..., N and Ac be its complementary set of *A*. Then A is referred to as the information set, if the indices in A corresponds to the leftmost K elements in π_N, and A_c is referred to as the frozen set, which is formed by the last (N − K) indices in π_N .

Then the codeword, *x* of length *N*, corresponding to message vector *u* of length *N* can be generated as,

$$x = u.G_N \qquad (8)$$

where $u = (u_A, u_{A^c})$ and G_N is given by (4). The main idea is to construct a message vector *u* where the elements u_i with $i \in A \subseteq \{1, 2, ..., N\}$ carry information and the other elements u_j with $j \in A^c$ contain values known at the transmitter and receiver (frozen bits - fixed to 0's). In otherwords, the codeword, *x* can also be expressed as,

$$x = u_A G_N(A) \oplus u_{A^c} G_N(A^c) \qquad (9)$$

where $G_N(A)$ denotes the submatrix of G_N formed by the rows with indices in *A*. Thus, polar codes are specified in terms of four parameters {*N*, *K*, *A*, u_{A^c}}, where K is the code dimension, specified by the size of *A*. The code rate is defined as *R* = K/N. Hence, the size of information set u_A and frozen set u_{A^c} can be represented by *K* or *NR* and (*N* − *K*) or *N*(1 − *R*) respectively.

2) Successive Cancellation Decoding

Let *y* be the channel output corresponding to the transmitted codeword *x*, which is to be decoded using SC decoding. The main task of SC decoder is to generate an estimate of information bits, $\hat{u}_i$ of u_i given knowledge of *A*, u_{A^c} and *y*. Since the decoder knows the frozen bits, the decoding error in the frozen part can be avoided by setting, $\hat{u}_{A^c} = u_{A^c}$, and thus the real task of decoding reduces to finding the estimate of $\hat{u}_A$ of u_A. The information bits are estimated successively using SC decoder as follows:

$$\hat{u}_i = \begin{cases} u_i & , \ if\ i \in A^c \\ h_i(y_1^N, \hat{u}_1^{i-1}), & if\ i \in A \end{cases} \qquad (10)$$

where the decision functions, $h_i: Y^N \times X^{i-1}, i \in A$ are computed for all $y_1^N \in Y^N, \hat{u}_1^{i-1} \in X^{i-1}$ as,

$$h_i\left(y_1^N, \hat{u}_1^{i-1}\right) \triangleq \begin{cases} 0\,, \ if \dfrac{W_N^{(i)}\left(y_1^N, \hat{u}_1^{i-1}|0\right)}{W_N^{(i)}\left(y_1^N, \hat{u}_1^{i-1}|1\right)} \geq 1 \\ 1, \qquad\qquad if\ otherwise \end{cases} \qquad (11)$$

The estimate of the information bits, $\hat{u}_i, i \in A$ are generated one by one, based on the channel output vector, y and the estimate of the previous information bits, $\hat{u}_1^{i-1}$. A block error is said to occur if the decoder output, $\hat{u}_A$= u_A. The error probability of SC decoder is upper bounded as follows [15].

$$P_e \leq \sum_{i \in A} Z\left(W_N^{(i)}\right) \qquad (12)$$

In order to improve the error correction performance of the SC decoder, the successive cancellation list (SCL) decoding algorithm was proposed [16]. In SCL decoding, the L most likely paths u_1^{i-1} are tracked. When decoding u_i for $, i \in A^c$, the decoder extends each path into two paths exploring both possibilities $u_i = 0$ and $u_i = 1$. If the number of obtained paths exceeds L, the decoder picks L most likely paths as surviving ones and prunes the rest based on a certain Path Metric (PM).

B. Non-Binary Polar Codes

In the case of non-binary polar codes [17, 18], the message vector u to be encoded can be elements from Galois field, F_q, where $q = 2^r$. The polarization kernel, F for non-binary polar codes can be defined as shown in (13), where $\alpha \in F_q$.

$$F = \begin{bmatrix} 1 & 0 \\ \alpha & 1 \end{bmatrix} \qquad (13)$$

In general, α needs to be a primitive element of F_q to achieve channel polarization. The recursive nature of binary polar codes is preserved in the case of non-binary codes also and this facilitates the use of SCL decoder. The convolution of probabilities required to find the transition probabilities and decision function for q-ary symbols can be efficiently computed using fast Fourier transform (FFT) over Galois field [19].

C. Proposed Homomorphic Encryption Scheme based on Polar Codes

The encryption scheme is designed based on Rao-Nam symmetric code-based cryptosystem [8]. In this scheme, the plaintext or information vector u is first post-multiplied by a scrambler matrix, S_M and is then followed by multiplication with the generator matrix, G of the underlying code. An intentional error vector e is then added to the resultant vector, which is then post-multiplied by a random permutation matrix, P_M. The ciphertext of length N generated using (N, K) non-binary polar code over F_q can be mathematically represented as,

$$c = (uS_M.G + e)P_M \qquad (14)$$

where u is of length K, S_M is a $K \times K$ matrix, G is of size $K \times N$, e is of length N and P_M is of size $N \times N$.

The decryption of u from c can be done by first post-multiplying with the inverse of permutation matrix, P_M^{-1}followed by the subtraction of the intentional error vector. Then, the iterative SCL decoding of non-binary polar codes need to be performed to extract uS_M, which is then followed by the multiplication with the inverse of scrambler matrix. The decoding operation provides reliability to the transmitted information vector, based on the error correcting capability of the underlying code.

For this encryption scheme to be homomorphic or to enable encrypted domain data processing, the decryption of the sum of the ciphertexts should result in the sum of corresponding information vectors. This can be mathematically represented as,

$$\begin{aligned} c_1 + c_2 &= (u_1 S_M.G + e_1)P_M + (u_2 S_M.G + e_2)P_M \\ &= (u_1 + u_2)S_M.G + (e_1 + e_2)P_M \qquad (15) \end{aligned}$$

The sum of information vectors $u_1 + u_2$ can be extracted using the above-mentioned decryption step, by using the sum of corresponding error vectors. Similarly, the decryption of the scalar multiplication of the ciphertext with an element, β in F_q should also result in the corresponding information vector multiplied with the same scalar.

$$\beta c = (\beta uS_M.G + \beta e)P_M \qquad (16)$$

From Eq. 15 and Eq. 16, it is clear that the error vector should also possess the homomorphic properties. In order to resist chosen plaintext attacks against RN cryptosystems, the intentional error vectors of Hamming weight at least $N/2$ need to be used. Linear feedback shift register (LFSR) based keystreams have good statistical and randomness properties needed for the intentional error vector. Moreover, they possess the additive homomorphic properties required to compute linear operations (sum, mean etc) in encrypted domain. In this work, we utilize the design of LFSR based random vector over F_q [20].

1) *Key Generation*

The LFSR secret key consists of the feedback polynomial, $f(x)$ which decides the feedback connections and initial state, k. Since the LFSR keystream possess linearity or additive homomorphic properties, linear combinations of keystreams should generate a new keystream with similar randomness properties. But due to the linearityproperty, during encrypted domain aggregation, linear combinations of keystreams result in a null vector. This spoils the security offered by the intentional error vector in the aggregated ciphertext. In order to prevent this situation, linearly independent keystreams need to be used as error vectors. If the initial states of LFSR are linearlyindependent, then keystreams will also be linearly independent. However, only L linearly independent initial statescan be generated by an LFSR of length L. A scheme for generating L linearly independent initial states from an initial key, k of length L through cyclic shifting operation is also proposed [20]. Since the initial states derived through cyclic shifts lacks security, a more secure method is proposed. The steps involved in the generation of linearly independent keystreams is detailed in Algorithm 1.

Algorithm 1 Generation of linearly independent keystreams

```
Input: $k_1, \gamma_1, f(x)$
Output: $e_i$
for $i = 1 : L$ do
    $e_i$ = LFSR1-PRNG($k_i$)
    $k_{i+1} = RS_i(k_1) \cdot \gamma_i$
    $\gamma_{i+1}$ = LFSR2- UPDATE STATE ($\gamma_i$)
end for
return $e_i$
```

The required keystreams are generated using two LFSRs, in which LFSR1 is used to generate keystreams for intentional error vectors and LFSR2 is used to derive the random multiplier required to update the LFSR1 states. The algorithm takes initial state of LFSR1, k_1; feedback polynomial, $f(x)$; and initial key, $\gamma_1 \in F_q$ of LFSR2 as inputs. LFSR2-UPDATE STATE denotes the initial state updation, γ_1 of LFSR2, which produces a random symbol. $RS_i(k_1)$ indicates that the initial state, k_1 is right shifted by i bits. The key length of intentional error vector is decided by the number of initial states of LFSR1, feedback polynomials and multiplier γ_1. The key length of intentional error vector depends upon the number of initial states of LFSR1 and the feedback polynomial, which can be expressed as $KL_e = 2log_2(N.log_2(q))$.

The generator matrix of non-binary polar code is made secure by embedding secrecy in the size of information bit u_A and frozen bit size u_{A^c}, which inturn makes the generator matrices $G_N(A)$ and $G_N(A^c)$ secure. For a (N, K) polar code with rate R, the number of different generator matrices possible is given by $\binom{NR}{K}$.

The design of scrambler matrix can be simplified by taking it as a circulant matrix of size $K \times K$, in which case only the first row need to be stored. At the same time, to ensure the invertibility of the matrix, the first row polynomial, $fr(x)$ should satisfy the condition that gcd($fr(x), x^{K-1}$) should be a zero degree polynomial. Hence, the key length of scrambler matrix is $KL_S = Klog_2(q)$.

The permutation matrix of size $N \times N$ is chosen to be a block diagonal matrix of the form shown in (17), where each π_j, $1 \le j \le N/m$ is of size $m \times m$ and 0 represents an all zero matrix. Each submatrix can be made random, by deciding the position of 1's using a m-state LFSR. Thus, the key length of permutation matrix can be denoted as, $KL_P = \frac{N}{m} log_2(m)$.

$$P = \begin{bmatrix} \pi_1 & 0 & 0 & 0 \\ 0 & \pi_2 & 0 & 0 \\ 0 & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \pi_{\frac{N}{m}} \end{bmatrix} \qquad (17)$$

2) Privacy Preserving Data Aggregation

As shown in system model (Fig.1), the encrypted data (energy consumption units) received from each consumer (SM) at different time instants need to be added (aggregated) by the aggregator before sending to the smartgrid controller. For this, initially, each smart meter will be embedded with a distinct master key, which is the concatenation of different keys and the keys used by different SMs will be known to smart grid operator. The SM can derive the keys required for scrambler, generator, permutation matrix and initial state of LFSRs from this master key. Even though the generator matrix key is fixed, the keys of scrambler matrix, permutation matrix and intentional error vector, will be updated during every month, for improved security. The ciphertexts corresponding to the daily energy consumption received from the SM in a month will be summed (aggregated) at the smart grid controller for statistical analysis (bill generation).

IV. SECURITY ANALYSIS

The security of the proposed scheme is analysed in terms of the capability of the proposed scheme to withstand various security attacks. A code-based encryption scheme should mainly resist chosen plaintext attacks such as Rao-Nam attack and majority voting attack.

A. *Brute Force Attack*

In this attack, the attacker performs an exhaustive key search to find the correct secret keys. The resistance to this attack depends on the key space. The work factor for this attack is the total number of possible combinations of all secret keys. For a non-binary polar code of size (256, 128) over F_4, with a rate $R = 0.8$, the total attack complexity is $O(2^{258})$, since the number of secret generator matrices possible is $N_G = 2^{192}$, the number of scrambler matrix and permutation matrix possibilities are 2^8 and 2^{40} respectively and the total number of intentional error vectors are 2^{18}. Similarly for (512, 256) code over F_8, with same rate $R = 0.8$, the total attack complexity is $O(2^{500})$, since $N_G = 2^{387}$, the possible combinations of scrambler and permutation matrices are 2^{11} and 2^{80} respectively and the total number of intentional error vectors are 2^{22}.

B. *Rao-Nam (RN) Attack*

In this kind of chosen plaintext attack [9], attacker tries to extract the scrambled generator matrix, $S_M G$ using several plaintext-ciphertext pairs, u_i, c_i. The difference between two ciphertexts is computed to derive the estimate of each of scrambled generator matrix. i.e., $c_1 - c_2 = (u_1 - u_2)S_M G + (e_1 - e_2)P_M$. Since the intentional error vectors are carefully designed in such a way that the randomness properties will be preserved even after linear combination, the resultant error vector $(e_1 - e_2)$ also retains the Hamming weight of $N/2$. So, it is not possible to independently verify the correctness of the derived generator matrix rows. Hence the complete G matrix need to retrieved and verified. The work factor required for this depends upon the number of permuted error vectors, $N_{eP} = e_i P$, i.e., $2^{(KL_P + KL_e)}$. For a non-binary polar code of size (256, 128) over F_4 and (512, 256) over F_8, with same rate $R = 0.8$, the attack complexity will be around $O(2^{58})$ and $O(2^{102})$ respectively, with permutation submatrix size taken as $m = 32$. Thus, this attack is not feasible due to high attack complexity.

C. *Majority Voting(MV) Attack*

MV attack [21] tries to retrieve the secret G matrix of size $K \times N$ in a much more efficient way compared to RN attack. Here the attacker first generates all possible ciphertexts corresponding to all zero information vector. All these permuted error vectors e_iP will be placed as a matrix. Then the attacker generates all possible ciphertexts corresponding to an arbitrary information vector, say u_1 and arrange it as another matrix, $u_1S_MG + e_iP$. Now, difference of these two matrices is computed to generate the estimate of u_1S_MG, by performing majority voting on each column of this matrix. To find the generator matrix, this process will be continued with K linearly independent plaintexts; and finally premultiply the estimate matrix, u_iS_MG with matrix comprising of K linearly independent plaintexts. The attack complexity is $O(KN.N_{eP})$, since $K \times N_{eP}$ majority votes are required for N columns. For a non-binary polar code of size (256, 128) over F_4 and (512, 256) over F_8, with same rate $R = 0.8$, the attack complexity will be around $O(2^{75})$ and $O(2^{122})$ respectively. Thus, the proposed scheme withstands this attack also,due to high attack complexity.

V. PERFORMANCE ANALYSIS

The performance of the proposed system is evaluated in terms of error performance, computational complexity and communication overhead.

A. *Error Performance*

The bit error rate (BER) performance of non-binary polar codes using SCL decoding for various list sizes over additive white Gaussian channel (AWGN) channel is shown in Fig. 2. Simulation results are performed using MATLAB software, for (256, 128) polar code over F4. It is clear from the graph that for a given signal to noise ratio (Eb/N0), BER reduces with increase in list size of SCL decoding. Since in the proposed encryption scheme, the security is embedded in the generator matrix, the attacker will not be able to extract any meaningful information from the ciphertext and the error probability of attacker will be 0.5.

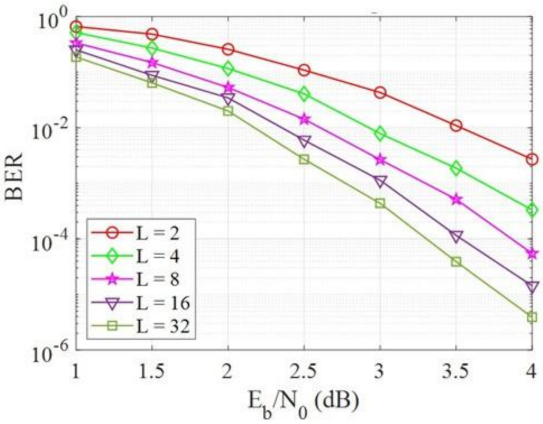

Fig. 2 BER Performance

B. *Computational Complexity*

The computational complexity of the proposed cryptosystem is analyzed in terms of the number of multiplications and additions. The ciphertext in this scheme is generated as $(u_iS_M.G + e_i)P_M = (u_iG_S + e_i)P_M$, where G_S is the scrambled generator matrix. Multiplication of information vector with G_S matrix involves KN modular multiplications and N modular additions, whereas addition of intentional error vector contributes to another N modular additions. Since multiplication with permutation matrix implies simple rearrangement, that step will not intro- duce any multiplication or addition. Thus the encryption complexity of the proposed scheme can be expressed as $KN(log_2q)^2$ binary multiplications (BM) and $2N(log_2q)$ binary additions (BA), since the complexity of modular multiplication with respect to log_2q is $O((log_2q)^2)$ and that of modular addition is $O(log_2q)$.

During decryption, the received ciphertext is post multiplied with the inverse of permutation matrix, which does not involve any computational complexity. Subtraction of e_i involves N modular additions. This step is then followed by the SCL decoding to retrieve scrambled information vector, u_iS. In order to extract the plaintext, the decoded vector is post-multiplied with the inverse of scrambler matrix. The SCL decoding complexity of non-binary polar codes with list size, L' is $O(rqL'Nlog_2N/2)$, since the FFT computation for finding transition probabilities over F_q has a complexity of $O(rq)$ per r bits. In order to extract the plaintext, the decoded vector is post-multiplied with the inverse of scrambler matrix and this includes K^2 modular multiplications and K modular additions. Hence the overall decryption complexity is $K^2(log_2q)^2$ binary multiplication and $(N + K)log_2q$ binary addition, in addition to the decoding complexity.

C. Communication Overhead

The communication overhead of the proposed system depends upon the size of encrypted data, since it is being transmitted. The overhead incurred is the ratio of ciphertext size to plaintext size and is the inverse of code rate, R for our scheme. Reducing the overhead by increasing the code rate may decrease the error performance of the underlying code. So suitable code rate needs to be selected, by taking into account the trade-off between communication overhead and data reliability.

VI. Comparison and Discussions

In this section, the proposed homomorphic encryption scheme based on polar codes is compared with two widely used additive homomorphic encryption schemes used for smart grid data aggregation, elliptic curve ElGamal (EC-EG) and Paillier scheme.

A. EC-EG Scheme

In order to provide 80-bit security, an elliptic curve over finite field F_p is to be chosen, where p should be a prime number of at least 160 bits length [5]. Here, the information to be send is mapped to an elliptic curve point and thus the ciphertext is an elliptic curve point. Since one point in elliptic curve consists of x and y - co-ordinates, the size of ciphertext is double that of plaintext and communication overhead is 2 [22].

In EC-EG scheme p is of 163 bits. Encryption operation consists of two scalar multiplications and 1 point addition. Double and add algorithm is used for scalar multiplication, which involves p doublings and $p/2$ additions on 163-bit elliptic curve. Approximately, 5 modular multiplications are required per point addition and doubling. This is equivalent to $[2(p + p/2)] \times 5$ p-bit modular multiplications. Therefore, the complexity of encryption is $O([[2(p+p/2)] \times 5].log_2 p^2)$ BM and $(log_2 p)$ BA. Decryption of EC-EG scheme requires one scalar multiplication and so the complexity in decryption is $O([(p + p/2) \times 5].log_2 p^2)$ BM. This step tries to reverse the mapping operation and is similar to solving the discrete logarithm problem (DLP) on elliptic curve. The difficulty in solving DLP increases with message size and messages of large size cannot be retrieved due to this issue.

B. Paillier Scheme

The security of this scheme relies on the difficulty in integer factorization [23]. The information to be encrypted are considered as elements of Z_c, the set of integers modulo c, where c is the product of two large prime numbers. The communication overhead of Paillier scheme is 2 since the ciphertexts are denoted as integer modulo c^2. To provide 80-bit security, c should be at least 1024 bits. The encryption and decryption process involves one exponentiation and one modular multiplication operation. Exponentiation operation with respect to exponent w, where $w = 1024$ can be considered equivalent to $2log_2w$ modular multiplications. Thus the encryption and decryption complexity is $(2log_2w) + 1$ modular multiplications or $[(2log_2w) + 1].(log_2c)^2$.

C. Comparison

Table 1 shows the performance comparison of the proposed homomorphic encryption scheme with EC-EG and Paillier scheme for fixed 80-bit security. The $(512, 256)$ polar code over F_8 with rate R = 0.8. is considered for comparison as it provides 80-bit security. The computational complexity of sending K messages is described for the proposed scheme, whereas the complexity mentioned for the other schemes in Section *A* and *B* corresponds to the transmission of single message. Thus, the encryption and decryption complexity mentioned in table corresponds to that of sending *K* messages for the other two schemes. From the table, it is clear that the computational complexity and communication overhead is very less for the proposed system. Moreover, it offers error correction whereas the other schemes do not provide error correction.

Table 1: Comparison of the proposed system with EC-EG and Paillier scheme for 80-bit security level

Parameters	**Elliptic curve ElGamal**	**Paillier**	**Proposed**
Computational Complexity	Enc – 2^{35} BM + 2^{16} BA	Enc – 2^{40} BM	Enc – 2^{20} BM + 2^{13} BA
	Dec – 2^{36}BM	Dec – 2^{40} BM	Dec – 2^{22} BM + 2^{13} BA
Communication Overhead	2	2	1.25
Data Reliability	No	No	Yes

VII. Conclusion

A private key additive homomorphic cryptosystem based on non-binary polar codes is proposed for supporting secure smart grid data processing. The main advantage of this scheme compared to other related schemes is that it simultaneously provides data confidentiality and data reliability. This is due to the fact that the design of proposed scheme is done by preserving the error correction capability of the non-binary polar codes, and without sacrificing security. The security is achieved through decoding problem by incorporating secrecy in generator matrix and randomizing the codeword through the addition of random error vectors. The error vectors are designed based on LFSR, to preserve its randomness properties during encrypted domain processing. Through mathematical cryptanalysis, it is shown that the proposed cryptosystem resists all known attacks. Moreover, the developed scheme is compared with elliptic curve ElGamal and Paillier which possess additive homomorphism, in terms of communication overhead and computational complexity. The results shows that this system outperforms the existing methods and is suitable for privacy preserving linear operations.

References

[1] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, G. P. Hancke, Smart grid technologies: Communication technologies and standards, IEEE transactions on Industrial informatics 7 (4) (2011) 529–539.

[2] X. Li, X. Liang, R. Lu, X. Shen, X. Lin, H. Zhu, Securing smart grid: cyber attacks, countermeasures, and challenges, IEEE Communications Magazine 50 (8) (2012) 38–45.

[3] K. Xue, Q. Yang, S. Li, D. S. Wei, M. Peng, I. Memon, P. Hong, Ppso: A privacy-preserving service outsourcing scheme for real-time pricing demand response in smart grid, IEEE Internet of Things Journal 6 (2) (2018) 2486–2496.

[4] L. Chen, R. Lu, Z. Cao, Pdaft: A privacy-preserving data aggregation scheme with fault tolerance for smart grid communications, Peer- to-Peer networking and applications 8 (2015) 1122–1132.

[5] O. R. M. Boudia, S. M. Senouci, M. Feham, Elliptic curve-based secure multidimensional aggregation for smart grid communications, IEEE Sensors Journal 17 (23) (2017) 7750–7757.

[6] E. Vahedi, M. Bayat, M. R. Pakravan, M. R. Aref, A secure ecc-based privacy preserving data aggregation scheme for smart grids, Computer Networks 129 (2017) 28–36.

[7] R. J. McEliece, A public-key cryptosystem based on algebraic, Coding Thv 4244 (1978) 114–116.

[8] T. R. Rao, K.-H. Nam, Private-key algebraic-coded cryptosystems, in: Conference on the Theory and Application of Cryptographic Techniques, Springer, 1986, pp. 35–48.

[9] T. R. Rao, K.-H. Nam, Private-key algebraic-code encryptions, IEEE Transactions on Information Theory 35 (4) (1989) 829–833.

[10] J. van Tilburg, Security-analysis of a class of cryptosystems based on linear error-correcting codes.

[11] R. Hooshmand, T. Eghlidos, M. R. Aref, Improving the rao-nam secret key cryptosystem using regular edf-qc-ldpc codes., ISeCure 4 (1).

[12] M. Esmaeili, T. A. Gulliver, Joint channel coding-cryptography based on random insertions and deletions in quasi-cyclic-

low-density parity check codes, IET communications 9 (12) (2015) 1555–1560.

[13] J. Liu, Y. Wang, Z. Yi, Z. Lin, polarrlce: a new code-based cryptosystem using polar codes, Security and Communication Networks 2019 (2019) 1–10.

[14] R. Hooshmand, M. R. Aref, Polar code-based secure channel coding scheme with small key size, IET Communications 11 (15) (2017) 2357–2361.

[15] E. Arikan, Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels, IEEE Transactions on information Theory 55 (7) (2009) 3051–3073.

[16] I. Tal, A. Vardy, List decoding of polar codes, IEEE transactions on information theory 61 (5) (2015) 2213–2226.

[17] W. Park, A. Barg, Polar codes for q-ary channels, $q = 2^r$, IEEE Transactions on Information Theory 59 (2) (2012) 955–969.

[18] S. Caycı, O. Arıkan, E. Arıkan, Polar code construction for non-binary source alphabets, in: 2012 20th Signal Processing and Communications Applications Conference (SIU), IEEE, 2012, pp. 1–4.

[19] S. Cayci, T. Koike-Akino, Y. Wang, Nonbinary polar coding for multilevel modulation, in: 2019 Optical Fiber Communications Conference and Exhibition (OFC), IEEE, 2019, pp. 1–3.

[20] V. Lakshmi, S. Deepthi, P. Deepthi, Collusion resistant secret sharing scheme for secure data storage and processing over cloud, Journal of Information Security and Applications 60 (2021) 102869.

[21] R. Struik, J. van Tilburg, The rao-nam scheme is insecure against a chosen-plaintext attack, in: Conference on the Theory and Application of Cryptographic Techniques, Springer, 1987, pp. 445–457.

[22] L. C. Washington, Elliptic curves: number theory and cryptography, Chapman and Hall/CRC, 2008.

[23] P. Paillier, Public-key cryptosystems based on composite degree residuosity classes, in: International conference on the theory and applications of cryptographic techniques, Springer, 1999, pp. 223–238.