Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Survey of Compressive Sensing Techniques Based on Secure Date Aggregation in WSNs

marwa madkour (✉ marwa.madkour@eaea.sci.eg)
Egyptian Atomic Energy Authority
S. E. Soliman (✉ Soliman_1950@yahoo.com)
Egyptian Atomic Energy Authority
M. I. Dessouky (✉ dr_moawad@yahoo.com)
Faculty of Electronic Engineering, Menoufia University
F. E. Abd El-Samie (✉ fathi_sayed@yahoo.com)
Faculty of Electronic Engineering, Menoufia University
A. S. Elsafrawey (✉ Amir.hafez@el-eng.menofia.edu.eg)
Faculty of Electronic Engineering, Menoufia University
M.E. hammad (✉ m_hammad2020@yahoo.com)
Egyptian Atomic Energy Authority

Research Article

Keywords: Wireless sensor networks, Security, Elliptic-Curve-Diffie-Hellman, PMLEACH routing, Elliptic curve cryptography

Posted Date: October 18th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3453941/v1

License: 🅭 🅯 This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations:
Competing interests: The authors declare no competing interests.

Survey of Compressive Sensing Techniques Based on Secure Date Aggregation in WSNs

Marwa Madkour[1], S. E. Soliman[1]†, M. I. Dessouky[2]†, F. E. Abd El-Samie[2]†, A. S. Elsafrawey[2]†and M.E. hammad [3]†*

[1] Nuclear Fuel Technology Department, Hot Labs Center, Egyptian Atomic Energy Authority,
Cairo, Egypt.

[2] Department of Electronics and Electrical Communications, Faculty of Electronic Engineering,
Menoufia University, Egypt. E-mails:

[3] Engineering Department, Nuclear Research Center,
. Egyptian Atomic Energy Authority, Cairo, Egypt.

*Corresponding author. *Marwa Madkour* E-mail: marwa.madkour@eaea.sci.eg;

Contributing authors: Soliman_1950@yahoo.com., dr_moawad@yahoo.com. , fathi_sayed@yahoo.com. , Amir.hafez@el-eng.menofia.edu.eg. , m_hammad2020@yahoo.com;

†These authors contributed equally to this work.

Abstract

A key function of wireless sensor networks (WSN) is data collection. Due to the hot spot issue and the limited energy supply, developing data gathering techniques is complicated. The WSN faces three main challenges: security, data routing, and processing a lot of data. Since compressive sensing can achieve simultaneous sampling and compression, it is widely used in signal processing technique. Due to resource limitations and computational limitations, WSN security solutions are different from those in traditional networks. Compressive sensing (CS) and Elliptical curve Diffie-Hellman key exchange are used to solve these problems. The measurement matrix is configured to be as a public key that is understood by both the sensor node and the base station in order to achieve high safety and efficiency for data gathering in wireless sensor networks. Security and effective data collecting are the main study goals. A prime-numbered address strategy for TPID (tree path identifier) routing and cluster head selection is used. Comparison between seven types of CS algorithms is introduced over different data sparsity levels. The network parameters is being tested are Network life time, throughput, residual node energy and total energy dissipated. The results revels that the compression system can reduce the size of the transmitted data and consequently the energy consumption while still maintains the data security.

Keywords: **Wireless sensor networks, Security, Elliptic-Curve-Diffie-Hellman, PMLEACH routing, Elliptic curve cryptography.**

I. Introduction

There are many WSNs based on Cs applications has been somewhat studied, includes: - imaging, video processing, cognitive radio networks, machine type communications, radar signal processing. In communication systems, physical layer operations include, channel estimation in wireless networks and channel estimation in power line communication [1].

Two basic categories of encryption algorithms exist. One is based on private keys, while the other is based on public keys. Irrespective increased security provided by public-key based encryption techniques (such as ECC, RSA, etc.); they are not favored for usage on resource-constrained WSN devices [2, 3]. The private key-based encryption techniques (such as AES, DES, etc.) don't require a lot of compute or memory, but they do demand that the keys be pre-stored on nodes that are vulnerable to attack when left in unguarded situations [4-6].

II. LITERATURE REVIEW

Encryption mechanisms, including ECC and Diffie-Hellman, have been proposed for effective key management and distribution in WSNs [7]. Key pre-distribution techniques [8] conserve energy and offer scalable storage capacity and efficiency. Approaches employing symmetric keys, such as Lightweight security scheme [9] and hierarchical clustered WSNs [10] enhance security, scalability, and energy consumption. However, identical CS matrices generated by the BS and sensor nodes in each round may be vulnerable to Known Plain-text Attacks [11]. Although these methods have shown success in ensuring data privacy and security, their high computational complexity makes them less suitable for low-power and limited-storage sensor nodes.

A lot of work has been proposed by utilizing CS as a security scheme regardless of its security degree to achieve data privacy, security, energy and efficiency [12-16].

A CS-based security approach for data collecting (SeDC) is proposed in [17], where the authors used compressive sensing across limited fields to lower the cost of data gathering and public key encryption to address the issue of key distribution. The similarity between CS and lattices is investigated. The network lifetime is reduced, though; if additional calculations, such as encryption and compression, are performed at each node (they are computationally demanding operations.). Cipher-text attack and plain-text attack are analyzed under two different scenarios. Authors in [18] proposed with the objective to improve the performance of CDG by utilizing Even-Rodeh codes and ElGamal algorithm to compress and secure the data respectively. The compression ratio and space savings are the variables that are being investigated. Data security is provided via El-Gamal algorithm; however the size of data from the El-Gamal encryption process is amplified and the difficulty of solving the discrete algorithm is introduced.

In [19] a routing protocol for load balancing and QoS enhancement is proposed. WSN face security challenges based on CS encryption scheme such as Brute Force Attack, Fault Injection Attacks, Side-Channel Attacks, and Invalid Curve Attacks. To address these issues and achieve secure data collection, researchers have explored various approaches. Compressive sensing (CS)-based systems combining compression and encryption have been used to lower data gathering costs and improve network performance [20]. The remainder of this paper is structured as follows. The background knowledge is introduced in Section III .The proposed scheme is introduced in Section IV. Section V is an experimental result. Section VI is a conclusion.

III. BACKGROUND

A. CS background

Compressive sensing (CS) avoids traditional transforms for compression. It maps high-dimensional signals to lower-dimensional domains via random sampling. Reconstruction of the original signal is possible from compressed data. Fourier and wavelet transforms, although not the main focus, are relevant for CS. Hence, a brief overview of sparsity and signals is valuable.

If the sparse signal is projected on an appropriate basis, natural signals like music, images, or seismic data can be stored in compressed form. Many projection coefficients are zero or small enough to be neglected when the domain (basis) is properly set. A signal is considered to be h-Sparse if it only contains h non-zero coefficients. Signal is referred to as compressible if a high number of projection coefficients are tiny enough to be neglected. If there are orthogonal bases provided by (ψ_1,ψ_2,ψ_3,……. ψ_N), we can express x as a k sparse signal in ψ as in equation (1).

$$x = \psi k \tag{1}$$

If $x \in R^N$ describes the sparse signal to be detected, In the CS framework, compression of x is performed using the following linear operation:

$$[y]_{M\times 1} = [\Phi]_{M\times N}[x]_{N\times 1} \tag{2}$$

$[y]$: The compressed signal

$[x]$: The signal vector to be compressed

$[\Phi]$: Projection or measurement matrix of

$$y = \Phi x = \Phi\Psi k = \Theta k \quad (3)$$

It is possible to reconstruct x under certain conditions on the measurement matrix Φ if x is sufficiently sparse. The first natural solution is to solve the following optimization problem.

$$\min \| \hat{x} \|_0 \ \text{s.t.}\ y = \Phi\hat{x} \quad (4)$$

Unfortunately, this l_0-norm minimization problem is generally computationally complex. In order to approximately solve (4), several approaches have been proposed. . The Convex Relaxation algorithms, Greedy algorithms, Iterative Thresholding Algorithms ,Combinatorial/ Sublinear Algorithms , Non-convex Optimization category , Bayesian algorithms, and Bregman iterative algorithms [21] .

B. Elliptic Curve Diffie–Hellman background

In cryptography, a key is a sequence of characters utilized in an encryption/decryption algorithm to transform data in such a way that it appears random [22].

The Elliptic Curve Diffie-Hellman (ECDH) Key Exchange is a key agreement scheme that enables two parties, each possessing an elliptic-curve, to establish a shared secret over an insecure channel.

ECDH uses algebraic curves over finite field F_p to generate keys to be used by the parties. Also, both parties need to agree on an elliptic curve beforehand. In mathematics, an elliptic curve can be described as a plane algebraic curve is given by an equation of the form $E_p(a,b)$:

$$y^2 = x^3 + ax + b \mod(p)$$

$4a^3 + 27b^2 \neq 0$, This ensures that the curve's graph is non-singular, and thus a tangent line can be identified at every point, which is important for point duplication. Figure (1) shows what a real elliptic curve looks like$E_p(-2,2)$:

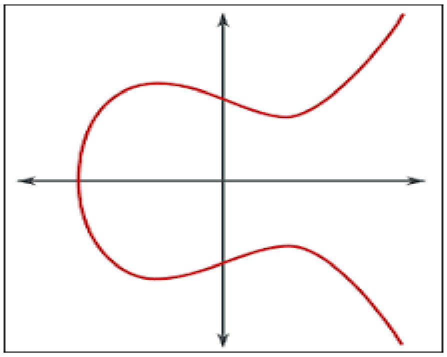

Fig.1. Elliptic Curve

ECC offers a comparable level of cryptography like RSA but with noticeable reduced key size. The discrete logarithm problem (DLP) for an Elliptic curve is defined as for two points Q_1 and Q_2on eliptic curve surface to find a positive integer n ,such that $Q_2 = n * Q_1$.n Represent the private key. It is important to note that certain elliptic curves are considered more secure than others, and therefore the selection should be made with caution and based on established security criteria. Elliptical curve cryptography (ECC) has three uses: digital signature (ECDSA), encryption (ECC) and key exchange (EC-DH).

ECDH is an asymmetric encryption algorithm used for key sharing. The key pairs consisting of a private key and a public key are utilized. The public key is distributed openly, while the private key must remain confidential. ECDH operates by multiplying the private key with the public key of another party to generate a shared key, which is then employed for symmetric encryption.To illustrate how this works, This scenario can be explained as follow:

Ax and Bx must first agree on a certain Elliptic Curve $E_p(a,b)$, a prime p, and point on curve G; ;$G \in E_p(a,b)$∶ .

- Ax selects a secret value an "α", computes $A = \alpha G$, and sends A to Bx.

- Bx also selects a secret value ''β'', calculates $B = \beta G$, and sends B to Ax.
- Ax then calculates the public key $= \alpha B$ to determine the Diffie-Hellman Key, and Bx does the same βA

Ax's key is $= \alpha B = \alpha(\beta G) = \beta(\alpha G) = \beta A$=Bx's key. both Ax and Bx end up with the same key .all the above calculation is calculated in mod(p).

ECC is the encryption algorithm used at each cluster head.First compute the public key:

$E_{pu} = E_{pr}G$.Second the message encryption using E_{pu} . Let M be the message aggregated by CH. This message needs to be represented on the curve. Then, select q randomly from $0 < q < p - 1$. So the encrypted message is

$$Y = [c_1,c_2] = [q * G, M + (q * E_{pu})] \tag{5}$$

Third and finally message decryption

$$M = [c_2 - E_{pr}c_1] \tag{6}$$

C. WSN structure:

A WSN is made up of spatially dispersed sensors and one or more base station (also known as sink node). Sensors monitor physical factors such as temperature, vibration, or motion in real time and generate sensory data. A sensor node can act as both a data source and a data router. A sink, on the other hand, gathers information from sensors. In an event monitoring application, for example, sensors must communicate data to the base station (s) when they identify the occurrence of events of interest. WSN deployment can be mesh type or star type or tree types are mainly adopted. Tree structure has many advantages like solving hot spot problem. Figure (2) depicts tree WSN architecture.

Wireless data transmission through insecure channels makes it relatively simple for an attacker to listen in on the conversation. As a result, the primary issues for WSNs are security and energy conservation. Security can be improved by encrypting the data sent between sensor nodes and the BS. The CS compression and reconstruction processes can be paralleled to encryption and decryption, respectively; as a result, data compression and encryption are carried out concurrently, which increases energy efficiency.

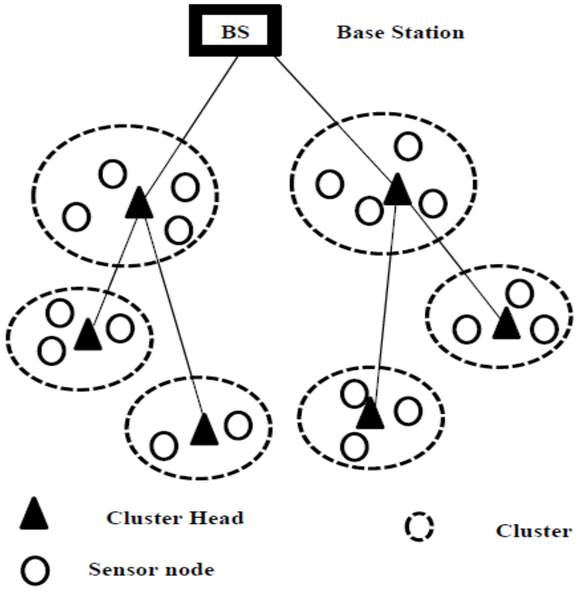

Fig.2. WSN Structure

IV. SYSTEM MODEL

A. Two challenge scenarios

- If each node generates two keys, namely the public key and the private key, the node can share its public key with others for communication purposes, while the private key is used for decryption. However, this process is computationally intensive and requires a significant amount of energy. Therefore, it is not a suitable solution to be applied on low-powered WSN devices.

- If the Base Station (BS) generates both the public and private keys and transmits the public key to the entire network, each node can use it to encrypt its data. However, when it comes to data aggregation at the Cluster Head (CH), the CH needs to decrypt the data, perform aggregation, and encrypt it again for transmission to the BS. This process results in increased energy consumption for the CHs and requires the CHs to possess the private key. This poses security concerns as the use of a public-key algorithm in this manner becomes insecure.

B. The challenges are solutions

- The integration of Compressed Sensing (CS) method, which combines encryption and compression, with Elliptic-curve cryptography (a public key algorithm) allows CS to address the aggregation issue without the need for the private key at the Cluster Head (CH) side.

- An EC-DH key sharing method is introduced to solve the CS-Encryption key distribution problem, enabling secure exchange of a pseudo-random key between the Base Station (BS) and nodes in a straightforward manner.

- A novel method is introduced to enhance the security of the CS scheme by safeguarding it against potential security attacks.

C. *The proposed scheme stages*

The proposed scheme encompasses three main stages: Setup, Secure data gathering and Data reconstruction stage. In the Setup stage, efficient clustering and routing trees are established by leveraging the properties of prime numbers to minimize power consumption during data transmission. The Secure data gathering stage utilizes a Compressive Sensing (CS) based method for data compression and encryption. Lastly, Data reconstruction stage introduces a proficient reconstruction algorithm that effectively recovers the original sensor data from the compressed samples, thereby enhancing the data reconstruction process.

1) Setup

The proposed scheme leverages the principles of prime numbers theory to establish a routing tree that enables multi-hop routing from Cluster Heads (CHs) to the Base Station (BS), thereby enhancing the power efficiency of Wireless Sensor Networks (WSNs). The process of CH selection and cluster creation in the proposed scheme is as follows:

Each node in the network is assigned a prime-numbered address, and the paths in the routing tree are identified using Tree Path Identifiers (TPIDs). A TPID is obtained by multiplying the node addresses along the path, resulting in a unique ID for each path based on the prime addresses of the nodes. By decomposing the TPID, each node along the path can be identified, and each node can determine its own path based on its TPID. This approach ensures that each path is uniquely identified and facilitates efficient routing within the network.

Prime number based Modified Leach algorithm (called PMLEACH) is used for CH selection and cluster creation. PMLEACH is an improved algorithm with the aim to achieve equitable distribution of expended energy in the WSN [23]. The CH election and cluster creation process of PMLEACH is as explained below:

CH Election and Cluster Formation

In the initial round, nodes randomly generate numbers compared to a threshold T. Nodes with numbers below T become members, while those above T become cluster head (CH) nodes. In subsequent rounds, an energy model considers factors such as distance to the base station (BS) and residual energy parameters. It is important to differentiate nodes based on distance and energy for energy expenditure and network lifetime. A new cost function is implemented to manage these factors effectively.

Cluster formation begins as follows: Every normal (non-CH) node s join the CH (CH_i) that satisfies the two conditions: (1) the distance between node s and CH_i is less than the distance between node s and BS (2) the distance between CH_i and BS is less than the distance node s and BS. If the two conditions for the node s do not hold with any of the selected CHs, it selects the nearest CH from the selected CHs.

2) *Secure Data Gathering stage*

The first security stage is the CS-Encryption at each non-CH node is implemented by introducing EC-DH key sharing method that enables secure exchange of sensing matrix Φ seed between the Base Station (BS) and the non-CH node. The security of the CS based encryption technique is provided by the fact that an attacker cannot access the sensing matrix, which contains pseudo-random values generated by the exchange of CS keys (seed) between the each sensor node and the BS.

The CS-based encryption method achieves the objectives of minimizing the amount of transmitted data and safeguarding the transmitted data against potential adversaries.

Second stage of security is between the CH and the base station, where each CH use the public key to encrypt all the received data from its node members.

The primary objective of this phase is to ensure the security of the sensor data transmitted among the cluster (CH), and the Base Station (BS). To accomplish this, the integration of ECC and the Elliptic Curve Diffie-Hellman (ECDH) public key algorithm is employed.

At the Base Station (BS), two types of keys are generated: (i) CS-Matrix (seed) in the first round only and (ii) ECDH private and public keys for Cluster Heads (CHs). Firstly, for the CS-Key, the BS selects the most suitable choice, such as a Bernoulli or Gaussian distribution matrix. Pseudo-random number generation techniques require a starting point known as a seed, which can be initialized as g_0 at the BS and transmitted using EC-DH. By using identical seed values between a node and the BS, an identical random matrix Φ is generated for data encryption and decryption. However, the main drawback is that if an adversary successfully guesses the seed, they can produce the same matrix. The proposed technique aims to generate the seed reliably and make it difficult for attackers to guess. Subsequently, in further rounds, the seed generation at each sensor node and the BS is performed using the following equation.

$$c_{n+1} = b_d * c_n * (1 - c_n) \quad ;b_d \neq 0 \tag{7}$$

In further rounds The BS generates public and private keys E_{pu} , E_{pr} as in ECDH and send the share key to each CH.

- *The algorithm steps*

Node Side:

- Each node i receive the seed value g_{0i}
- Each node i use the seed value to generates its Φ_i matrix and perform compression operation $y_i = x_i * \Phi_i$ by equation (3)
- Each node i uses the seed to generate the secret value $s_i = g_{0i}^{-1}$.
- Each node i calculates the secure $\grave{y}_i = y_i * s_i$ and then sends $\grave{y}_i$ to its CH_i.

CH Side:

- Each CH i aggregates its member nodes data $M_i = \grave{y}_1 + \grave{y}_2 + \grave{y}_3 + \ldots\ldots \grave{y}_r$, where r equals the nodes count in the cluster i.
- Each CH uses the public key E_{pui} to encrypt the aggregated data M_i and output Y_i by equation (5)

BS Side:

- The BS uses the private key E_{pri} to decrypt the received data of each CH_i Y_i and output $M_i$by equation (6)
- BS calculates the secret value $s_i = g_{0i}^{-1}$and computes each node data individually.
- The BS uses the generated seed g_{0i} and generates the associated CS matrix Φ_i.
- Finally, the BS recovers the actual data with the help of the agreed reconstruction algorithm by equation (4)

3) *Data reconstruction stage*

This section compares seven sparse recovery algorithms. The Basis Pursuit algorithm represents Convex Relaxation, while Compressive Sampling Matching Pursuits, Subspace Pursuits, and Orthogonal Matching Pursuits represent greedy algorithms. Approximate Message Passing represents the iterative thresholding category. Bayesian via Relevance Vector Machine represents the Bayesian category. The Split Bregman iterative algorithm represents the Bregman iterative algorithms. Heavy Hitters on Steroids (HHS) and Iterative Re-weighted Least Squares were excluded from the simulation results due to their longer reconstruction process time, making them unsuitable for real-time WSN applications.

V. PERFORMANCE EVALUATION

A WSN network of 100 sensor nodes is used for testing the performance of the proposed algorithms. The network nodes are randomly distributed through a network area of A (200 m×200 m). For fair comparison, the nodes locations are stored in order to apply all the proposed algorithms on the same network environment. Also the BS is located at a fixed location at the middle of the network area. During performance evaluation, the following network parameters are considered:

- The network nodes are immobile and based on batteries with initial energy of 2 joules per node.
- Every node is assigned a prime-numbered address [23] and can be identified uniquely.
- Transmissions are assumed under ideal channel circumstances, and collision-free transmissions. The number of simulated rounds is 2000.
- The nodes gather a sparse signal (Signal Length=512) contains [0, spikes of 1] and deliver the gathered data to their relevant CH nodes.

PMLEACH routing algorithm [23] is applied for organizing the network nodes into clusters, where each cluster has one CH and some of CMs. During each simulation round, the CMs of each cluster transmit their sensed data to their respective CH¸ which in turns aggregate the received data and then transmit the aggregated data to BS. OMP algorithm that used for compressive sensing in [24] is also applied to the organized network and their response is compared with the proposed algorithms. The response of OMP algorithm and the different proposed algorithms is observed for different cases of sparsity lengths. The Compression performance metric is used for comparison of different algorithms behavior.

- Compression ratio = output data size/input data size
- Compression performance =100*(1-Compression ratio)

Figure (3) illustrates the estimated MSE of the reconstructed data for different compression ratios of the applied CS based algorithms when the sparsity equals 50. The SBI based algorithm achieves the worst behavior compared to other algorithms as the reconstruction error increased largely for compression performance higher than 20 %. The BP algorithm enhances the performance greatly as it achieves the lowest MSE at compression performance of 62.5%. SP, AMP and CoSaMP algorithms achieve also performance close to BP one. The traditional OMP algorithm as shown achieve an acceptable MSE up to a compression performance of only 57.8 %.

The performance of the applied algorithms is evidenced as shown in Fig. (4) For a sparsity of 100 bit with some noticed enhancement in the response of SBI algorithm that able to achieve an acceptable MSE for a slightly higher compression ratio of 33.7 %. For 100 bit sparsity, the behavior of OMP and BCS-RVM get worse as they can operate well up to only 37.9 %, 35.2% respectively. While the behavior of both AMP and BP achieves the best results as they can operate up to 44.5 % with an acceptable MSE value. SP and CoSaMP algorithms achieve almost equal compression performance 45.5%, 43.6% respectively .SP and CoSaMP algorithms as indicated by Fig. 3, 4 achieve an approximately the same behavior in low and moderate sparsity as they belong to the same category (Greedy algorithms) and have almost the same steps except in they are in different order. They apply a pruning process in different way.

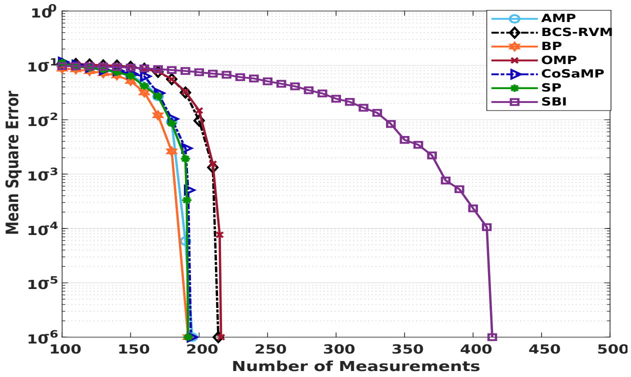

Fig. 3 Mean Square error behavior for Sparsity equals 50 for different CS based algorithms.

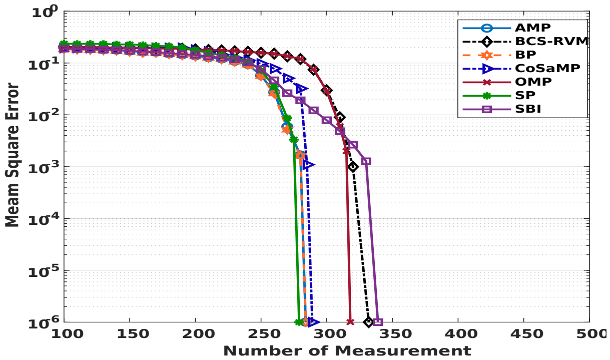

Fig. 4 Mean Square error behavior for Sparsity equals 100 for different CS based algorithms.

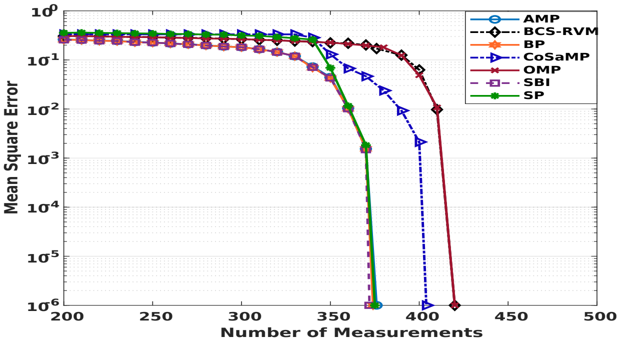

Fig.5 Mean Square error behavior for Sparsity equals 170 for different CS based algorithms.

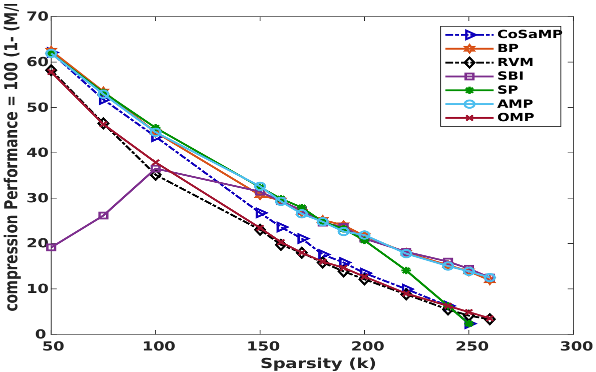

Fig.6 Compression performance and Sparsity for different CS based algorithms.

Another scenario is indicated in Fig. (5) For sparsity of 170 bit, the response of OMP algorithms behaves like BCS-RVM with the variation of the compression ratio. These algorithms achieve the worst response as they result in acceptable MSE value up to only a rate of compression equal 18 %. The other SP, SBI, BP, and AMP algorithms achieve an enhanced response as they increase the rate of compression to 26.7, 27.3, 26.95, and 26.56 %, respectively. CoSaMP algorithm achieves savings of 21.1% that is in the middle.

Figure (6) illustrates the optimum compression rate for the different applied algorithms that achieve the acceptable value of MSE for different sparsity values. The optimum compression rate is defined as the maximum rate that achieve the minimum required MSE value which is selected as said to be $1 \times e^{-6}$. This threshold MSE value that ensures a correct recovery of the received signal may vary from one application to other. It is clear from Fig. 6 that there is inversely proportional relationship between the sparsity and the achievable compression performance, whereas the sparsity value increases, the number of measurements required increase and therefor the compression performance decrease.

The relation of Mean Square Error and compression performance for different CS based algorithms is shown in fig. (7). as the sparsity increase the compression performance decrease because the number of measurements required increase to achieve a certain value of mean square error.

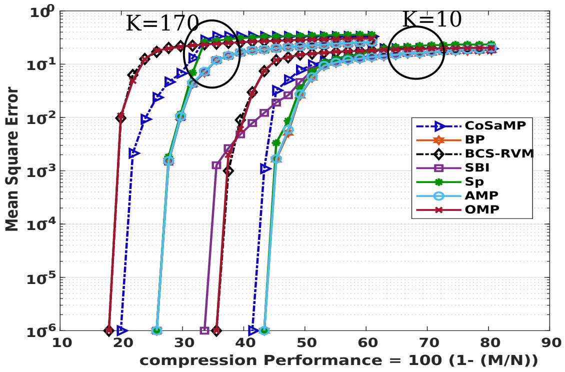

Fig.7. Mean Square Error and compression performance for different CS based algorithms

The following section is devoted for studying the effect of the achieved compression through the different proposed algorithms on the network metrics like; network life time, First Dead Node, and network residual energy. The length of the compressed data for the different applied algorithms is determined from Fig. 5 for a sparsity case of 170 at MSE of $1 \times e^{-6}$

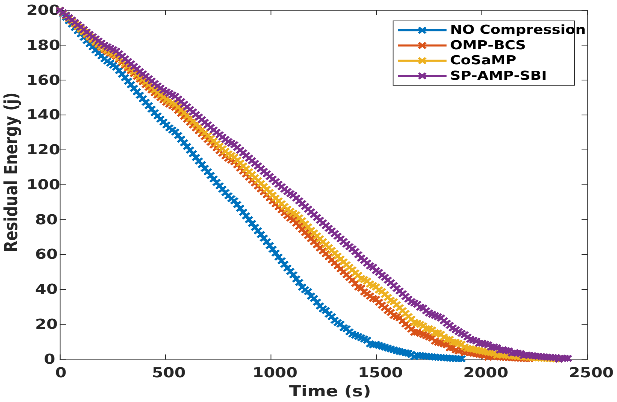

Fig. 8Network residual energy per round for different CS

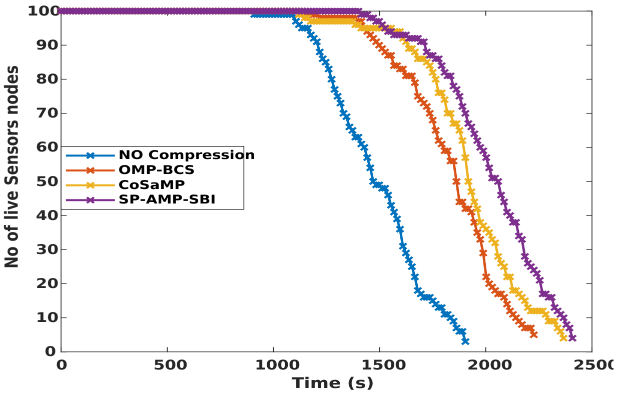

Fig.9 Network life time and FDN for different CS based algorithms.

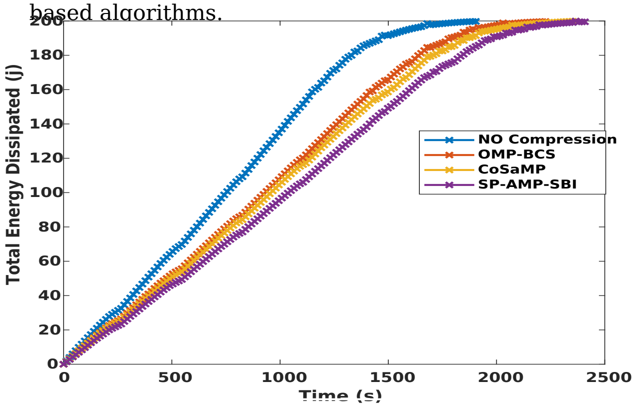

Fig. 10 Total energy dissipated per round for different CS

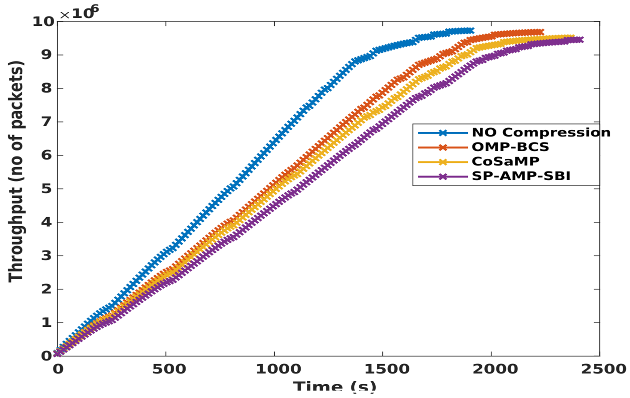

Fig. 11 Network throughput per round for different CS

The first impact of the achieved compression is the network residual energy per round that illustrated in Fig. 8 It is clear from this figure. That transmitting sensor data without compression deplete rapidly the network energy, while applying the traditional CS that based on OMP-BCS algorithm enhances the network energy. A further enhancement in the network residual energy is noticed when applying the different proposed CS techniques, especially SP, AMP, and SBI algorithms as they dissipate less energy per each round of the network operation.

The next impact is the network life time and First Dead Node (FDN) that illustrated in Fig. 9 This network parameter reveals how the network can operate with all of its nodes for long time without losing one of them. The energy saving due to data compression is reflected as shown on the network life time, the network life time of no CS case is minimum followed by OMP and BCS compressive sensing case, where the CS based on SP, AMP, and SBI algorithms not only enhances the network life time, but also begin losing the first node at higher rounds. CoSaMP algorithm network life time was moderate.

The energy efficiency is presented in Fig. 10 where the total energy dissipated in no CS case is the largest one followed by OMP and BCS case. The energy dissipated by SP, AMP, and SBI algorithms was minimum, and energy dissipated in CoSaMP algorithm was moderate.

Finally, a measure of network throughput that related to the amount of data transmitted is also illustrated in Fig. 11. The CS based on SP, AMP, and SBI algorithms has the ability for sending the same network information, but with small data size compared to the traditional algorithm and compared to the case of sending data without compression.

VI. Conclusion

Based on the previous findings, it can be concluded that the proposed algorithm combines CS-based encryption methods and public key algorithms to achieve a high level of security with minimal communication costs. This is achieved through the encryption and compression of sensor data using the CS scheme, as well as the inclusion of data compression, CS-based encryption, and key sharing stages. To enhance the security of this process, the suggested approach incorporates an effective key sharing method and employs a secret value technique to protect the CS method against various attack models. Additionally, the proposed technique utilizes a public-key mechanism for encrypting cluster data, thereby mitigating CS attacks during the data aggregation and EC-DH encryption phases. As a result, the suggested scheme outperforms other CS systems in terms of security and the operational life span of WSN. ECC depends on DLP however any cryptographic system is vulnerable to various types of attacks. Brute Force Attack where this type of attack attempts to find the private key by trying all possible combination, ECC solve this problem by using large key size and Standardized Curves. Side-Channel Attack, the attacker finds the private key by monitoring the side channel leaked information like power consumption, timing, and electromagnetic radiation during the execution of the encryption algorithm this problem is solved by using secure hardware. Fault Injection Attacker, in this case the attacker introduces faults in the ECC calculation process, if successful the attacker reveals a part of the private key. ECC solve fault injection attacker and Invalid Curve Attacker by using trusted ECC and well tested algorithms and libraries that are resistant to various types of attacks. Key Reuse and Management Issues; if a single private key is used for multiple transmissions the attacker can exploit the poor key management practices to gain unauthorized access. Quantum Attackers, ECC is considered secure against classical computers, but it may become vulnerable to quantum computers in the future. ECC must stay informed with the latest update. Finally Implementation Flow; where error in the implementation of ECC algorithm can be exploited by attacker for example poor random generator, insufficient entropy or programming mistakes.so proper and robust random generator is necessary. In the future work we will study the effect of the number of keys in details.

References

[1] Thakshila Wimalajeewa, Senior Member, IEEE and Pramod K Varshney, “ Application of Compressive Sensing Techniques in Distributed Sensor Networks: A Survey,”2019

[2] Gulen, A. Alkhodary and S. Baktir, ”Implementing RSA for Wireless Sensor Nodes,” Sensors, 19(13), 2864, 2019

[3] E. T. Oladipupo, O. C. Abikoye, A. L. Imoize, J. B. Awotunde,Ting-Yi Chang, C.-Chi Lee, and D.-Thuan Do, “An Efficient Authenticated Elliptic Curve Cryptography Scheme for Multicore Wireless Sensor Networks, ”Received 6 December 2022, accepted 22 December 2022, date of publication 2 January 2023, date of current version 5 January 2023. Digital Object Identifier 10.1109/ACCESS.2022.3233632

[4] B. Abood, A. N. Faisal, and Q.A. Hamed, “Data Transmitted Encryption forClustering protocol in Hetrogeneous Wireless Sensor Networks , “Indonesian J. Elect. Comput. Sci. ,vol.25,no.1,pp.347-357,2022,doi:10.11591/IJEECS.V25I1.PP347-357.

[5] Osamy W., El-Sawy A. A. and Khedr A. M. Effective TDMA scheduling for data collection in tree based wireless sensor networks. Peer-to-Peer Netw., 13:796–815, 2020. https://doi.org/10.1007/s12083- 019-00818-z

[6] P. Zhang, S. Wang, K. Guo, and J. Wang, “A secure data collection scheme based on compressive sensing in wireless sensor networks,” AdHoc Networks, vol. 70, pp. 73–84, 2018.

[7] D. Mausam, W. Zenghui ED25519: A New Secure Compatible Elliptic Curve for Mobile Wireless Network Security, ” J, “ordanian Journal of Computers and Information Technology, Vol. 8, Issue 1 (31 Mar. 2022), pp.57-71.

[8] S. Itoo,A.Ali Khan, M. Ahmed, and M.J. Idrisi, . ” A secure and Privacy-Preserving lightweigh Authentication and Key ExchangeAlgorithm for smart AgricultureMonitoring System, ” IEEE Access,VOL(11) 5 june 2023..DOI:101109/ACCESS.2023.3280542.

[9] Aziz A. and Singh K. ” Lightweight security scheme for internet of things, ”Wireless Personal Communications, 104:101–120, 2019. https://doi.org/10.1007/s11277-018-6035-4.

[10] R. Priyadarshi, B.Gupta, and A. Anurag, ”Deployment Techniques in Wireless Sensor Networks:Asurvey, classifications,and challenges and future reserch issues , ”I.Supercomput., vol.76, pp.7333-7373, jan.2020,doi:10.1007/s1127-020-03166-5.

[11] D. Omar, A. M. Khedr, ” Prolonging Stability Period of Wireless Sensor Networks Using Compressive Sensing, International Journal of Communication Networks and Information Security (IJCNIS) Vol. 11, No. 1,April 2019.

[12] P. Zhang, S. Wang, K. Guo, and J. Wang, “A secure data collection scheme based on compressive sensing in wireless sensor networks,” Ad Hoc Networks, vol. 70, pp. 73–84, 2018.

[13] S. Ifzarne, I. Hafidi and N. Idrissi. ”Compressive Sensing Based on Homomorphic Encryption and Attack Classification using Machine Learning Algorithm in WSN Security.” In Proceedings of the 3rd International Conference on Networking, Information Systems and Security, pp. 1-6. 2020.

[14] S. Ifzarne, I. Hafidi and N. Idrissi. Secure data collection for wireless sensor network. In Emerging Trends in ICT for Sustainable Development, pages 241–248, 2021.

[15] Yang P. Wang Q., Lin D. and Zhang Z. An energy-efficient compressive sensing-based clustering routing protocol for wsns. IEEE Sensors Journal, 19:3950–3960, 2019. https://doi.org/10.1109/JSEN. 2019.2893912.

[16] Liu Z., Han Y.-L and Yang X.-Y. A compressive sensing–based adaptable secure data collection scheme for distributed wireless sensor networks. International Journal of Distributed Sensor Networks, 15:1150–1185, 2019. https://doi.org/10.1177/1550147719856516

[17] M. Budiman, E. M. Zamzami and C. L. Ginting. (2020). A Crypto Compression System Using ElGamal Public Key Encryption Algorithm and Even-Rodeh Codes. Journal of Physics: Conference Series. 1566. 012071. 10.1088/1742-6596/1566/1/012071.

[18] Ngabo C. I. and El Beqqali O. Implementation of homomorphic encryption for wireless sensor networks integrated with cloud infrastructure. Journal of Computer Science, 15:235–248, 2019. https://doi.org/ 10.3844/jcssp.2019.235.248

[19] Benelhouri, H. Idrissi-Saba, and J. Antari, “ An evolutionary routing protocol for load balancing and QoS enhancement in IoT enabled heterogeneous WSNs, Simulation Modelling Practice and Theory, ” Volume 124, 2023, 102729, ISSN 1569-190X, https://doi.org/10.1016/j.simpat.2023.102729.

[20] Li S. Liu Y. Cui D. Li Q., Xu B.. Reconstruction of measurements in state estimation strategy against deception attacks for cyber physical systems. Control Theory and Technology, 16:1–13, 2018. https:// doi.org/10.1007/s11768-018-7080-y.

[21] Z. Gao, L. Dai, S. Han, I. Chih-Lin, Z. Wang, and L. Hanzo, “Compressive sensing techniques for next-generation wireless communications,” IEEE Wireless Commun., vol. 25, no. 3, pp. 144–153, 2018.

[22] E. Taiwo; O. C. Abikoye,A.L. Imoize, J. B. Awotunde, T. Chang, C. C. Lee, and D. T. Do, “An Efficient Authenticated Elliptic Curve Cryptography Scheme for Multicore Wireless Sensor Networks,”IEEE Access,(2023),doi:10.1109/ACCESS.2022.3233632.

[23] A. Salima , W. Osamy, A. M. Khedr , A. Azizd, and M. Abdel-Mageed., “ A Secure Data Gathering Scheme based on Properties of Primes and Compressive Sensing for IoT based WSNs, ” JOURNAL OF Wi max CLASS FILES, VOL. 14, NO. 8, AUGUST 2015

[24] Salim A, Ismail A, Osamy W, M. Khedr A (2021) Compressive sensing based secure data aggregation scheme for IoT based WSN applications. PLoS ONE 16(12): e0260634. https://doi.org/10.1371/journal.pone.0260634.