A New Deterministic Code Allocation Technique for Data Compression in Wireless Sensor Networks

H. Anwer Basha, S. Arivalagan, P. Sudhakar, R.P. Narmadha

Abstract--- ***Energy utilization is the huge conflict in modeling WSN because it has only inbuilt limited battery. To diminish the sum of data requires to be communicated, an efficient technique data compression is used prior to transmission. Since, there is lot of textual data that are available in WSN dataset, lossless compression method is desirable where the data loss is not preferable. This paper introduces a new deterministic code allocation (DCA) technique for effectively compresses the data. The DCA technique is a dictionary based, single character encoding scheme which uses a static dictionary for codeword allocation. The unique feature of DCA technique is the use of fixed 4-bit codewords for every character in the input sequence. As the nodes are constrained on computational resources, the proposed DCA technique is highly suitable because of its easier implementation. In addition, the DCA technique compresses the data with no loss of quality and it does not requires any extra information to be transmitted along with the compressed file. To verify the robustness and compression performance, an extensive experimental analysis is carried out using real time WSN dataset interms of various performance measures. The experimental results ensured that the DCA method obtains better compression performance with less computation complexity.***

Keywords--- ***WSN, Data compression, Encoding, Energy utilization.***

I. INTRODUCTION

In the domain of information technology and Micro Electro Mechanical System (MEMS), the latest technological advancement allows the growth of low power, low cost, autonomous and compact sensor nodes. An essential element of IoT is Wireless Sensor Networks (WSN); it provides sharing of data over billions of gadgets for enhancing the environmental user control [1]. It comprise of many sensor nodes which is embedded deeply in real time for environmental condition monitoring. It is appropriate for applications which gather data and for tracking; for instance, industrial automation, precision agriculture, structural and seismic monitoring, industrial automation, health monitoring, smart cities, disaster management, environmental monitoring, etc. [2]. In practical, it is employed to sense physical conditions such as vibration, pressure, acoustic signals, temperature, humidity and so on [3]. Transceivers, microcontroller, battery and sensors are the elements that are equipped with each individual sensor node. The sensor node senses the physical conditions and transfers the data sensed subsequent to the node deployment randomly through multihop communication or directly to base station (BS) [4]. Lack of fidelity, energy, memory, packet size, bandwidth and computational capability are the factors that the sensor nodes are constrained. Energy utilization is the huge conflict in modeling WSN because it has only inbuilt limited battery. In sensitive environmental conditions, the sensor node deployment makes it challengeable or impossible to replace or recharge the batteries[5]. When comparing with processing and sensing operations in sensor nodes, many studies exhibits that huge sum of energy is spent over transmitting the data [6]. To save the available energy resourcefully, energy efficient data transmission method is required.

To diminish the sum of data requires to be communicated, an efficient technique data compression is used prior to transmission. Among four thousand (using Chipcon CC2420 transceiver) to two million (using MaxStreamXTend transceiver) cycles of computation, it is worth to spend saving a data byte by employing compression reported by [7]. The data size reduction that is being transferred within the network will benefit in major power saving. For energy-efficient WSNs, it is a main strategy to compress data before transmission. The common criterion under compressing data is the procedure to remove irrelevant or redundant data. In compact form, it presents the data without arbitrating the quality of data to a particular extent. Hence, there is lot of textual data that are available in WSN dataset, lossless compression method is desirable where the data loss is not preferable. For example, the factors such as alphabets in the real time and seismic data, numeral characters in humidity and temperature metrics are highly sensible to the data loss. The requirement of robust and efficient lossless compression method motivated us to do this research over WSN dataset.

To devise an efficient lossless compression technique for WSN, this paper introduces a new deterministic code allocation (DCA) technique for effectively compresses the data. The DCA technique is a dictionary based, single character encoding scheme which uses a static dictionary for codeword allocation. The unique feature of DCA technique is the use of fixed 4-bit codewords for every character in the input sequence.

Revised Version Manuscript Received on 22 February, 2019

H. Anwer Basha, Research Scholar, Department of Computer Science, Annamalai University. Chennai, TamilNadu, India(e-mail: hiiamanwerbasha@gmail.com)

S. Arivalagan, Assistant Professor, Department of Computer Science & Engineering, Annamalai University. Chennai, TamilNadu, India (e-mail:arivucseau@gmail.com

P. Sudhakar, Assistant Professor, Department of Computer Science & Engineering, Annamalai University. Chennai, TamilNadu, India

R.P. Narmadha, Assistant Professor (Senior Grade), Department of Computer Science & Engineering, Sri Shakthi Institute of Engineering & Technology. Chennai, TamilNadu, India.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

As the nodes are constrained on computational resources, the proposed DCA technique is highly suitable because of its easier implementation. In addition, the DCA technique compresses the data with no loss of quality and it does not requires any extra information to be transmitted along with the compressed file. To verify the robustness and compression performance, an extensive experimental analysis is carried out using real time WSN dataset interms of various performance measures.

The upcoming study is formulated as follows. Section 2 provides the recently presented compression techniques in WSN. The section 3 introduces the proposed DCA technique with an illustration. An extensive experimental analysis takes place in Section 4 and conclusions are made in Section 5.

II. RELATED WORKS

Because of the intrinsic features of sensing gadgets and lack of ability to replace or recharge the battery, WSNs are limited in energy when compared to other wireless networks. As noted by the researches, the data compression methods can attain efficiency in energy depending upon the constructive tradeoff among transmission energy and computation. Few compression methods that are particularly built for WSNs are described here. In [8], lightweight temporal compression (LTC) method is commenced, which gives few error into each reading, constrained through a controlling knob. More conservation in storage can be attained through compression by increasing the error bound. Subsequently, an enhanced version for run length encoding (RLE) for WSN is projected named as K-RLE with value of range [K-d, K+d] that approximate a string of N measurements as the pair (N,d), wherever d denotes a data item [9] and K denotes the precision level. An improved design of LZW dictionary based coding is a lightweight compression method known as Sensor LZW (S-LZW) which is developed especially for resource limited WSN[7]. The performance of compression has to be improved as there is a problem of growing dictionary. Moreover, it is less autonomous because encoding 528 data bytes into 10 bytes otherwise more intermittent packets. The residual packets cannot decodable if there is a packet loss during transmission. One of the predictive coding method is Lossless entropy compression (LEC)[10] which contains a encoder or a predictor. LEC estimates the difference among partitions and successive sensor measurements which the volume enhances in a massive way. Each group demonstrates the count of bits required to denote the differences in measurement, after then through a bounded compression table, the groups are entropy encoded. When compared to S-LZW, LEC achieves enhanced compression which cannot be suitable to the differences in source data statistics.

To avoid this problem, a robust and efficient compression method known as Sequential LEC (S-LEC) technique is projected [11]. For data compression in WSN, it employs sequential context data over neighborhood residue. To remove the disadvantages of LEC algorithm [12], other method named as adaptive lossless data compression (ALDC) is projected. To improve the performance of compression, ALDC algorithm can adapt dynamically with the changes in source data statistics. Through multiple code choices, it work in one pass adaptively and can be suitable to various data types. The ALDC method requires enhancing the performance of compression, even though it is autonomous. an adaptive linear filter is used for forecasting the M samples upcoming in the dataset and the prediction errors are compressed through entropy encoder [13] in adaptive linear filtering compression (ALFC). To fulfill the constrained computational abilities of the sensor node, a quantization procedure is included. Adaptive prediction procedure allows the machine to adjust with the modifications in the source and removes the requirement of describing the filtering coefficients previously. However, it needs some hardware resources, when compared to previous methods and ALFC method achieves enhanced compression rate. For predictive coding, a two modal transmission (TMT) is projected in [14], it comprises of two modes named as non-compressed and compressed mode. While in the initial compression mode, the compressed bits of error terms come behind the interval [−R, R] are transferred. While in second mode, instead of without compression, the raw data of error terms present outside the interval [−R, R] are transferred. Due to the poor performance of huge error terms prediction, this method solves the problem of compact coding efficiency. For WSNs [15], an efficient and fast lossless adaptive compression algorithm (FELACS) is designed to attain low memory and speed compression algorithm. Due to the practice of Golomb rice coding and it is very speed and robust to packet loss. Power savings and compression efficacy can be enhanced while it is robust and fast.

To diminish the sum of data transmission, a data-centric method is projected in [16]. For resource-constrained networks, while the application is autonomous over data accuracy otherwise the network works in exception condition, data reduction method finds helpful. For sensors [17], an adaptive sampling method is projected that calculate the optimal sampling frequencies. For snow-monitoring applications, a case study of a sensor is assumed. In terms of traditional fixed-rate method, the experimental results demonstrated that the adaptive algorithm diminished the count of samples acquired upto 79%. In remote and enlarged aquatic conditions, a new adaptive sampling method for power management in robotic monitoring of the water quality is initiated. A data-driven adaptive sampling algorithm (DDASA) is modeled to enhance the energy efficiency when entrusting the sampled data accuracy [18]. After three months of subsequent monitoring of water quality, it is clear that 30.66% of the battery energy can be conserved is demonstrated from the experimental results. To improve the sampling process, a multivariate sampling (MuSA) algorithm [19] by employing component analysis method in WSN is projected which is used to rank the data and retrieve the most delegate information after the ranking procedure. The experimental outcomes depicts that MuSA decreases the energy utilization and delay to data delivery over the network.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

When to allow the one step decoding to rebuild the actual data, [20] analyze the influence that the compressed sensing cause over network coding. Hence, it illustrates the modeling of coding and reconstruction algorithms which enable reconstruction accurately. To achieve robustness and energy efficiency in data gathering, subsequently, sparsest random sampling method for cluster-based compressive data gathering (SRS-CCDG) in WSNs is projected [21], where the sparest random sampling method is merged to clustered WSNs. Moreover, analytical model are used to model the relationship among energy cost and cluster size while employing various inter-cluster and intra-cluster transmission methods. The experimental results clears out that the SRS-CCDG majorly improves the system autonomous to node failure and diminishes the energy cost.

III. THE COMPRESSION ALGORITHM

3.1. Overview

The presented DCA technique is a bit oriented, dictionary based single character encoding technique which makes use of a deterministic code allocation dictionary (DCAD) to allocate codewords to the input sequence. The overall process of the DCA technique is shown in Fig. 1. It assigns a fixed length codewords to every character in the input sequence by the exploitation of codewords present in the DCAD. The unique feature of DCA technique is the use of fixed 4-bit codewords for every character in the input sequence. For an input sequence of length N, the DCA technique needs a minimum of $C_{bits}$to save the compressed file and is determined in Eq. (1):

$$C_{bits} = \sum_{i=1}^{N} N_{DCA}(i) \quad (1)$$

Where N_{DCA} indicates the number of bits in a codeword. Mostly, the number of bits needed to store a character is exactly 4. Next, the number of bits, on average, needed to store every character in DCA technique is determined Eq. (2).

$$DCA_{ch\_av} = \frac{C_{bits}}{N} \approx 4 \quad (2)$$

From Eq. (2), the maximum number of bits needed to store a character and it leads to better compression performance.

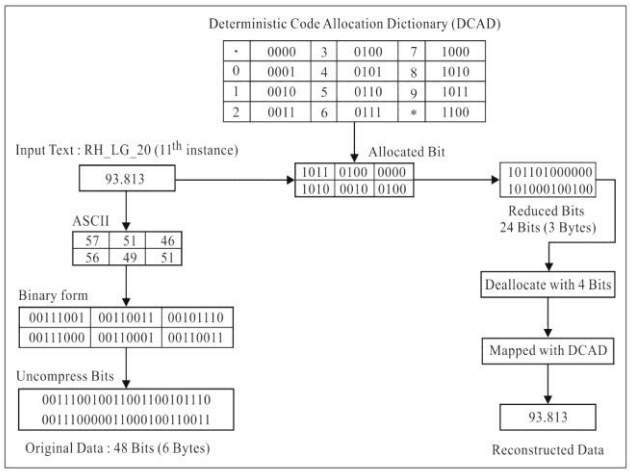

Fig. 1: Overall process of DCA technique

3.2. Working principle of DCA algorithm

The overall process of the DCA compression and decompression process is illustrated in Fig. 1 and the optimal DCAD is given in Table 1. At the beginning, the DCA technique maintains a DCAD which holds the codewords of 12 characters (0-9 numerals, ',' and '*'). Since the DCA technique is solely developed for WSN data which involves only numeric characters and dot(.) characters, codewords are maintained for those characters only. It mainly helps to reduce the complexity level of the algorithm. In addition, astreik (*) is used as a delimiter, used to identify the end of every instance. The DCAD contains a fixed 4-bit codewords for the 12 characters and thereby eliminates the need of delimiter for every individual character. As the DCAD is predefined, the compression and decompression side holds the DCAD apriori. Once the input sequence is received, the DCA technique uses DCAD and allocates codeword to them. After the codeword allocation is completed, the codewords are merged to form a compressed file. Finally, all the resultant optimal codewords of encoded characters are concatenated to generate the compressed file which holds only 50% of original size and is transmitted to the receiver.

Table 1: Optimal code words in DCAD

Character	Optimal Codeword	No. of bits required
.	0000	4
0	0001	4
1	0010	4
2	0011	4
3	0100	4
4	0101	4
5	0110	4
6	0111	4
7	1000	4
8	1010	4
9	1011	4
*	1100	4

The proposed DCA algorithm follows symmetrical compression where the decompression process is the inverse function of the compression process. As the DCA algorithm comprised of the same DCAD as the encoder, there is no need to transmit any additional information with the compressed file for the reconstruction process. Initially, the DCA algorithm reads the compressed file that comprises of the codewords in the binary form. The compressed data is split into 4 bits each and then the DCAD is used for mapping the codewords to the corresponding characters. Once all the codewords are identified with the corresponding characters, the decoded characters are merged to reconstruct the original data

3.3. Illustration

For better understanding of the DCA algorithm, an illustration is given here. In Fig. 1, the overall process is depicted for an instance from WSN dataset.

A sample is taken from the 11th instance of the 'RH_LG_20' dataset with the actual value of 93.813. Let us consider an input sequence (inp_{seq}) '93.813' of

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

length$N$ ($N = 6$), which contains numeric characters and '.' character. Without any compression, the inp_{seq} will be stored as follows: initially, it is converted to the ASCII characters and then it is transformed to the respective binary values. Every character in the inp_{seq} requires 8 bits to store it and the given inp_{seq} requires a maximum of 48 bits, i.e. 6 bytes to store it. By applying the DCA technique, the inp_{seq} will be read and the mapping of every character with the DCAD takes place. Now, the codewords will be allocated to every character in the inp_{seq}. i.e. the codewords for the character '9' is 1011, '3' is 0100, '.' is 0000, '8' is 1010, '1' is 0010 and '3' is 0100. Then, these codewords are concatenated together and generates the compressed file of only 24 bits, i.e. 3 bytes. Since the original data transmission requires 6 bytes to store the inp_{seq} and the DCA technique required only half the amount of the original data.

At the decoder, the DCA technique follows the reverse process where the compressed file will be read and it is split into 4 bits of each block. Then, the mapping of 4 bit binary sequence will be mapped with the DCAD and identifies the respective character, i.e. 1011 is matched to '9', 0100is matched to '3', 0000 is matched to '.', 1010 is matched to '8', 0010 is matched to '1' and 0100 is matched to '3'. In line with, every binary sequence is mapped with the DCAD and the reconstruction of the original data will be takes place. The given inp_{seq} achieves C_{bits} of 24 bits and $DCA_{ch_{av}}$of 4 bits instead of 48 bits. Since, the DCAD is stored in both the encoder and decoder, there is no extra processing or communication delay.

IV. PERFORMANCE VALIDATION

To examine the efficiency of DCA algorithm, it is tested over a collection of extensive WSN dataset like temperature and humidity dataset. With lately projected methods like ALDC, LEC, SLZW, FELACS algorithms, the DCA's performance is compared by means of power savings and compression performance. The projected algorithm is also compared with the conventional compression methods named as gzip [22], bzip2 [23],Huffman [24] andarithmetic coding [25].

4.1. Dataset used

SensorScope [26], the real-time environmental monitoring WSN dataset is used. Initially, temperature and humidity metrics from three SensorScope are tested: HES-SO Fish-Net Deployment, Le Gènèpi Deployment and LUCE deployment. The dataset size varies from 12,652 samples to 64,913 samples. The dataset description is provided in Table 2.

Table 2: Dataset description

Deployment name	Node ID	Symbolic name	No. of samples	Time Interval	
				From day	To day
LUCE	84	LU_84	64,913	23-November-2006	17-December-2006
HES-SO FishNet	101	FN_101	12,652	09-August-2007	31-August-2007
Le Gènèpi	20	LG_20	21,523	04-September2007	03-October-2007

4.2. Compression performance

Usually, through the compression ratio (CR) and compression factor (CF), the performance of the compression method is estimated and is expressed as

$$CR = \left(\frac{No.of bits required after compression}{No.of bits required before compression}\right) \quad (3)$$

$$CF = \left(\frac{No.of bits required before compression}{No.of bits required after compression}\right) \quad (4)$$

The uncompressed instances of humidity and temperature are denoted through 16 bit unsigned integers and are byte aligned. Hence, the actual size of the dataset given can be simply estimated from table 2. In table 3, for each dataset, the volume of compressed data and uncompressed data are given. It is visible from the table that the projected method achieves enhanced compression performance when compared to other methods over the dataset applied. To attain power saving [30], it is considered with reduced count of messages that each instance will be send to the BS. While each packet comprises 29 bytes of payload [31], required packet count to transfer uncompressed and compressed data can be simply estimated.

Table 3: Compressed size of DCA on various WSN dataset

Deployment name	Temperature	Original size(bits)	Compressed size (bits)	Relative Humidity	Original size(bits)	Compressed size (bits)
LUCE	LU_84 Temp	3135824	723264	LU_84 RH	4096168	1513120
HES-SO FishNet	FN_101 Temp	680376	191216	FN_101 RH	696720	217400
Le Gènèpi	LG_20 Temp	1043032	263704	LG_20 RH	1358872	563688

Table 4 provides the attained values by the DCA method interms of CR and CF on the applied dataset. From the table, it is given that the CR and CF for the LU_84 Temp is 0.230645597 and 4.335656137 respectively. Similarly, for FN_101 Temp, the attained CR and CF are 0.281044599 and 3.558154129 respectively. In line with, the obtained CR and CF by the DCA method for LG_20 Temp are 0.252824458 and 3.955313533 respectively. Furthermore, for the LU_84 RH dataset, the attained CR and CF are 0.369398911 and 2.70710056 respectively. In addition, the CR and CF of the FN_101 RH dataset are 0.312033529 and 3.204783809.

Table 4: Results of the DCA technique interms of CR and CF on various WSN dataset

Deploymentname	Temperature	CR	CF	Relative Humidity	CR	CF
LUCE	LU_84 Temp	0.2306455	4.3356561	LU_84 RH	0.3693989	2.7071005
HES-SO FishNet	FN_101 Temp	0.2810445	3.5581541	FN_101 RH	0.3120335	3.2047838
Le Gènèpi	LG_20 Temp	0.2528244	3.9553135	LG_20 RH	0.4148205	2.4106810

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

By means of bit rate and Space saving, to know the projected algorithm efficiency, its performance during compression is compared with recent methods. Table 5 and Table 6 demonstrate the gained outcomes. From the values, when comparing with the other methods, it is absolute that the projected method attained good compression over the given dataset on the whole. Over the conventional methods, S-LZW algorithm depicts worst performance and is less autonomous when compared to others. Especially for LG_20 Temp dataset, the actual instances are denoted through 12 bits and S-LZW gives the rate of compression as 12.4770 bits/sample. The results denote that this algorithm fails to set when dynamic modifications occur and tends to negative compression. It suffers from the problem of increasing dictionary while DCA method does not have any dictionary problem for its processing. As FELCAS and ALDC, it is denoted that LEC method depicts nearby performance. But it fails to create efficient outcomes when compared to DCA method, because it is static and employs one coding table. Subsequently, for storing coding dictionaries, ALDC needs a hardware memory and it exhibits slow performance, because it requires to access memory for each instance to retrieve the needed code. To repeatedly access the memory, it has more energy utilization. While comparing with the entire methods, DCA technique attains enhanced performance over compression for DCA technique. FELACS manage to generate enhanced outcomes over other schemes for seismic dataset and also achieves nearby compression performance as DCA method. The projected method creates mainly enhanced compression outcomes than FELACS in the applied dataset.

Table 5: Comparative results of the DCA technique interms of Bit rate

Dataset	Bit rate				
	LEC	S-LZW	ALDC	FELACS	DCA
LU_84 Temp	4.6732	8.1628	4.1896	4.1153	1.8452
FN_101 Temp	5.5423	11.1472	5.2432	5.1940	2.2484
LG_20 Temp	7.3981	12.4770	6.8960	6.8151	2.0226
LU_84 RH	6.1123	11.0012	5.5136	5.4199	2.9552
FN_101 RH	6.0645	10.1956	5.3872	5.2579	2.4963
LG_20 RH	8.3034	12.4915	7.5408	7.3832	3.3186

Table 6: Results of the DCA technique interms of SS

Dataset	SS (%)				
	LEC	S-LZW	ALDC	FELACS	DCA
LU_84 Temp	70.81	48.99	73.94	74.00	76.94
FN_101 Temp	65.39	30.35	67.48	67.53	71.90
LG_20 Temp	53.83	22.02	56.90	57.41	74.72
LU_84 RH	62.86	31.24	65.54	66.12	63.06
FN_101 RH	62.95	36.27	66.33	67.20	68.80
LG_20 RH	48.67	21.93	52.87	53.85	58.52

Comparison with traditional compression algorithms

With few popular methods named as arithmetic coding, bzip2, rar, Huffman and gzip, this section make comparison over the outcomes with the compression of DCA method. By means of SS, the comparison outcomes are shown in Table 7. It is absolute that the arithmetic coding and Huffman produces worst performance when compared to other techniques. Except DCA, rar, gzip, bzip2 somewhat manages to outperform the other methods well. On the entire given dataset, DCA technique shows enhanced results because of the simpler DCAD based code allocation with the input data. The DCA technique's better performance is due to the subsequent causes. DCA technique fortunately needs least count of bits to denote every character within the input sequence. It is demonstrated that the projected method takes a maximum of 4 bits for every possible character appears in the WSN dataset.

Table 7: Comparative analysis of various traditional compression algorithms in terms of Space savings

Dataset	SS (%)					
	Gzip	Bzip2	Rar	Huffman	Arithmetic	DCA
LU_84 Temp	48.87	69.24	69.16	23.98	26.62	76.94
FN_101 Temp	34.76	55.20	63.59	21.59	22.06	71.90
LG_20 Temp	31.38	46.84	51.56	22.34	22.65	74.72
LU_84 RH	37.86	57.82	59.03	23.79	24.06	63.06
FN_101 RH	41.29	56.22	59.12	23.19	23.34	68.80
LG_20 RH	27.61	42.56	45.26	18.85	19.30	58.52

The needed count of packets to transfer uncompressed and compressed data is shown in Table 8. In Table 9, Packet Compression Ratio (PCR) denotes the packet delivery ratio and is demonstrated as

$$PCR = 100 * \left(1 - \frac{Required\ No.of packetstotransmitcompresseddata}{Required\ No.of packetstotransmituncompresseddata}\right) \quad (5)$$

Table 8: Compressed packet size of DCA on various WSN dataset

Deployment name	Temperature	Original packet size(bits)	Compressed packet size (bits)	Relative Humidity	Original packet size(bits)	Compressed packet size (bits)
LUCE	LU_84 Temp	13516.48276	3117.517241	LU_84 RH	17655.89655	6522.068966
HES-SO FishNet Le Gènèpi	FN_101 Temp	2932.655172	824.2068966	FN_101 RH	3003.103448	937.0689655
	LG_20 Temp	4495.827586	1136.655172	LG_20 RH	5857.206897	2429.689655

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Table 9: Packet Compression Ratio (PCR) of DCA on various WSN dataset

Deployment name	Temperature	PCR (%)	Relative Humidity	PCR (%)
LUCE	LU_84 Temp	76.9354 40	LU_84 RH	63.0601 09
HES-SO FishNet	FN_101 Temp	71.8955 40	FN_101 RH	68.7966 47
Le Gènèpi	LG_20 Temp	74.7175 54	LG_20 RH	58.5179 47

It is clear from the table that the process of compression majorly diminishes the count of packets to be transferred and hence decreases the power utilization.

4.2. Power saving

In WSN, for processing, communication and sensing purposes, the sensor nodes spend energy. Usually, for data transmission, the energy utilization is majorly higher than processing or sensing the data [10]. This is because of the communication module residing in the sensor node and it is a highly forceful energy consumer. But, WSN works under battery which is inbuilt and they shows a study relation with network lifespan. Therefore, it is important for each sensor node to manage energy dissipation to enhance the network lifespan. The energy utilization decreases with the decrease in amount of data transmission. Before transmission, the data compression methods decrease the data size that tends to reduced sum of power utilization. This conservation cause owing to the compression directly affects the maximum lifetime expectancy of the sensor nodes [32]. Proficient compression tends to least sum of data transmission that tends to least power utilization out of the communication unit. The DCA method attains enhanced compression by the implication of very less computational resources and it tends to get high power conservation for each sensor node. Using common operations such mapping, the execution of DCA algorithm is done. In a fast manner, the encoding procedure is done and it compresses the data quickly. While compared to the consumed power for data transmission, the consumed computational power and computational time is very less and negligible. Subsequently, through DCA algorithm, the power saving percentage is estimated and is expressed in Eq. (6):

$$Powersaving = \frac{P_{s\_comp}}{P_{s\_uncomp}} \quad (6)$$

where P_{s_comp} denotes the sum of power conserved with the use of compression and P_{s_uncomp} denotes the consumed power after compression. While comparing to the power utilization of data transmission, it is considered that the utilization of computational power is low for DCA. Approximately, the power saving of the sensor node is more or less equal to the power conservation of the communication unit. Eq. (7):

$$Powersaving \approx \frac{N_{s\_comp}}{N_{s\_uncomp}} \quad (7)$$

N_{s_uncomp} is the bit count for each instance of the actual source data (ADC resolution) and N_{s_comp} is the average count of bits saved for each instance. Eq. (7) becomes,

$$Powersaving \approx \left(1 - \frac{compressionrate}{N}\right) \times 100 \quad (8)$$

Table 10: Performance comparison between various compression algorithms in terms of Power saving

Dataset	Power saving (%)				
	LEC	S-LZW	ALDC	FELACS	DCA
LU_84 Temp	66.62	41.69	70.07	70.61	86.82
FN_101 Temp	60.41	20.38	62.55	62.90	83.94
LG_20 Temp	47.16	10.88	50.74	51.32	85.55
LU_84 RH	49.06	8.323	54.05	54.83	75.37
FN_101 RH	49.46	15.04	55.11	56.27	79.20
LG_20 RH	30.81	12.57	37.16	38.47	72.35

Depending on the source data statistics, the power saving of each sensor node is depended. Eq. (8) is employed to estimate the power savings in percentage achieved through DCA algorithm while it is used to four test dataset. The comparative outcome in terms of power saving attained through recent and proposed methods are demonstrated in Table 10. It is absolute from table 10, remarkable power saving is achieved through DCA algorithm and in a significant manner; the network lifetime is also enhanced. It is clear that the projected method is 50% more efficient in terms of energy when compared to FELACS. As a result, the enhanced compression achieved through DCA method results to minimal network load which tends to least retransmissions and collisions.

V. CONCLUSION

This paper has devised a new DCA technique for effectively compresses the data. The DCA technique is a dictionary based, single character encoding scheme which uses a static dictionary for codeword allocation. The unique feature of DCA technique is the use of fixed 4-bit codewords for every character in the input sequence. As the nodes are constrained on computational resources, the proposed DCA technique is highly suitable because of its easier implementation. In addition, the DCA technique compresses the data with no loss of quality and it does not requires any extra information to be transmitted along with the compressed file. To verify the robustness and compression performance, an extensive experimental analysis is carried out using real time WSN dataset interms of various performance measures. The experimental results ensured that the DCA method obtains better compression performance with less computation complexity. As a future scope, the proposed DCA algorithm can be embedded in the real WSN hardware.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

REFERENCES

1. Sariga, P. Sujatha, A survey on unequal clustering protocols in wireless sensor networks, J. King Saud Univ. (2017).
2. K. Sohraby, D. Minoli, T. Znati, Wireless Sensor Networks: Technology, Protocols, and Applications, John Wiley & Sons, Hoboken, New Jersey, 2007.
3. Estrin, J. Heidemann, S. Kumar, M. Rey, Next century challenges: scalable coordination in sensor networks, in: Proceedings of the 5th annual ACM, 1999, pp. 263–270.
4. G. Anastasi, M. Conti, M. Di Francesco, A. Passarella, Energy conservation in wireless sensor networks: a survey, Ad Hoc Networks 7 (2009) (2009) 537–568.
5. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, Comput. Networks 38 (4) (2002) 393–422.
6. C.S. Raghavendra, K.M. Sivalingam, T. Znati, Wireless Sensor Networks, 1st ed., Springer US, US, 2004.
7. Sadler, Martonosi, Data compression algorithms for energy-constrained devices in delay tolerant networks, in: In: Proceedings of the 4th international conference on embedded networked sensor systems – SenSys ’06, 2006, pp. 265–278.
8. T. Schoellhammer, B. Greenstein, E. Osterweil, M. Wimbrow, D. Estrin, Lightweight temporal compression of microclimate datasets, UCLA Cent. Embed. Netw. Sens., 2004.
9. E.P. Capo-Chichi, H. Guyennet, J.M. Friedt, K-RLE: a new data compression algorithm for wireless sensor network, in: Proceedings of the 3rd International Conference on Sensor Technologies and Applications, (SENSORCOMM’09), 2009, pp. 502–507.
10. M. Francesco, M. Vecchio, An efficient lossless compression algorithm for tiny nodes of monitoring wireless sensor networks, Comput. J. 52 (8) (2009) 969–987.
11. Y. Liang, Y. Li, An efficient and robust data compression algorithm in wireless sensor networks, IEEE Commun. Lett. 18 (3) (2014) 439–442.
12. J.G. Kolo, S.A. Shanmugam, D.W.G. Lim, L.-M. Ang, K.P. Seng, An adaptive lossless data compression scheme for wireless sensor networks, J. Sensors (2012).
13. Kiely, M. Xu, W.-Z. Song, R Huang, B. Shirazi, Adaptive networks, linear filtering compression on realtime sensor networks, Comput. J. 53 (10) (2010) 1606–1620.
14. Y. Liang, W. Peng, Minimizing energy consumptions in wireless sensor networks via two-modal transmission, Comput. Commun. Rev. 40 (1) (2010) 13–18.
15. J. Gana, S.A. Shanmugam, D. Wee, G. Lim, L. Ang, Fast and efficient lossless adaptive compression scheme for wireless sensor networks q, Comput. Electr. Eng. 41 (2015) 275–287.
16. A.L.L. Aquino, E.F. Nakamura, Data centric sensor stream reduction for real-time applications in wireless sensor networks, Sensors 9 (12) (2009) 9666–9688.
17. Alippi, G. Anastasi, An adaptive sampling algorithm for effective energy management in wireless sensor networks with energy-hungry sensors, IEEE Trans. 59 (2) (2010) 335–344.
18. [18] T. Shu, M. Xia, J. Chen, C. de Silva, An energy efficient adaptive sampling algorithm in a sensor network for automated water quality monitoring, Sensors 17 (11) (2017) 2551.
19. A.L.L. Aquino, O.S. Junior, A.C. Frery, E.L. Albuquerque, R.A.F. Mini, MuSA: multivariate sampling algorithm for wireless sensor networks, IEEE Trans. Comput. 63 (4) (2014) 968–978.
20. M. Taghout, A.K. Chorppath, T. Waurick, F.H.P. Fitzek, On the design of a joint compressed sensing and network coding framework, in: European Wireless 2018; 24th European Wireless Conference, 2018, pp. 113–119.
21. P. Sun, L. Wu, Z. Wang, M. Xiao, Z. Wang, Sparsest random sampling for cluster-based compressive data gathering in wireless sensor networks, IEEE Access 6 (c) (2018) 36383–36394.
22. “Gzip Home Page,” 2018. [Online]. Available: http://www.gzip.org/.
23. “Bzip2 Home Page,” 2018.
24. D.A. Huffman, in: A method for the construction of minimum-redundancu codes, 1952, pp. 1098–1102.
25. I.H. Witten, R.M. Neal, J.G. Cleary, Arithmetic coding for data compression, Commun. ACM 30 (6) (1987) 520–540.
26. “Sensor Scope deployments homepage,” 2011. [Online]. Available: https://lcav. epfl.ch/page-145180-en.html.
27. T. Schoellhammer, B. Greenstein, E. Osterweil, M. Wimbrow, D. Estrin, Lightweight temporal compression of microclimate datasets, UCLA Cent. Embed. Netw. Sens., 2004

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication