Energy Efficient NK-RLE Data Compression Scheme

Narinder Singh, Harmaninderjit Singh

***Abstract:** Wireless sensor network vigorous research area now a day .In WSN sensor nodes are outfitted with tiny batteries having low power where power conservation is main concern to increase the network life time. In order to conserve energy by reduce the data to be send from sense nodes to sink node. Data compression is effective and well known technique, to get better compression results with varied data from data sources it is essential to introduce competent processing technique. In this paper we proposed improved data compression schemes motivated from RLE and K RLE. The proposed algorithm enhances the compression ratio in comparison of K-RLE and RLE. The result shows that our compression algorithms efficiently enhance the data compression ratio and hence improving energy efficiency in WSNs.*

***Keywords: -** Data compression, RLE, C-RLE, K-RLE, Energy Efficiency*

I. INTRODUCTION

Wireless Sensor Networks are becoming trendy because of their huge range of application areas. WSN based nodes are equipped with a processing and communicating unit organized in such a way to perform specific task [1]. Sensor nodes are usually powered by batteries processor, wireless transceiver, and memory. As batteries have low power it is important to design an effective deployment layout of WSN. As the nodes are deployed in inconsiderate site. Due to unattended deployment and inability of recharging, the power utilization of the nodes should be best possible.

Conservation of energy is major concern in WSN.The common solution to conserve the energy is to take the advantage random energy by shuffling the sensor nodes. Making some nodes active while putting another to sleep.[2] Another way is to decrease the amount of sensing data to be send by the sensor nodes . To do so DataCompression is best option to overcome the conservation of energy issue in WSN. [3]

II. LITERATURE SURVEY

C-RLE algorithm - the authors have proposed and evaluate a new compression approach, called C-RLE. It is based on the principle of the K-RLE algorithm. Proposed C-RLE solves the problem of transaction among energy utilization and compression rate effectiveness in K-RLE.

Revised Manuscript Received on October 30, 2019.
* Correspondence Author
Narinder Singh*, Research Scholar Department of Computer Science, Desh Bhagat University, Mandi Gobindgarh (Punjab), India. ns_kalra@yahoo.co.in
Harminderjit Singh, Assistant professor in Computer Science, Desh Bhagat University, Mandi Gobindgarh(Punjab), India, jeetsinder@gmail.com

On the basis of experimentation the authors proves that the proposed C-RLE approach keeps the same K-RKE's performance in term of compression ratio while the energy consumed can decrease up to 27.03% et 16.67% compared to the K-RLE and RLE algorithms, respectively. [4]

S. Jancy et.al [5] proposed a Packet level compression scheme. With this scheme a WSN can get improved compression ratio in comparison of traditional scheme. They also talk about the number of standard compression schemes like Arithmetic coding, RLE, Huffman coding and Delta encoding. Rawat, P., et.al. [6] Have presented an another algorithm to compress the data, is also motivated from K-Precision RLE algorithm k which is known as K-RLE algorithm. With this algorithm compression ratio is increased in comparison of RLE algorithm. So to get better compression outcomes with various statistics of data resources. They introduced in-network processing scheme to save the energy. R. Hou, have presented a new energy efficient algorithm for IEEE standard 802.15.4. It's implemented on physical layer to arrange the BFSK with the chance of amount of all data strings become visible in data output tributary. They simulate this scheme and show the better results of compression with various numbers of values of the dictionary size in comparison of traditional LZW algorithm and S-LZW algorithm [7]. This paper discusses the RLE compression algorithm to compress the data. This scheme gives better compression ratio in comparison of traditional schemes. They took a RLE data compression algorithm for WSN using real temperature datasets. There is a major limiting constraint in RLE, for RLE to achieve good compression ratio, the input data must contain long sequences of repeated characters, and this rarely occurs in the data generated from sensors. The energy consumption using RLE scheme gives better ratio. It consumes less energy as compression ratio is increased. They have presented in their work trade off among power utilization and compression effectiveness. [8][9]

III. DATA COMPRESSION

The data compression is used to reduce the quantity of data to be send without losing its originality. [10] Data Compression plays a significant job in area of communication and storage. The main merit of data compression is better utilization of recourses due to its compression ratio. Higher is compression better is efficiency. Lossy and Lossless are two classification of data compression, As the name indicates in lossy technique some data get loss during the process of data compression where as in case of lossless compression technique there is no loss of data or compressed data is same as uncompressed data. [11][12]

IV. SHORTCOMINGS IN EXISTING DATA COMPRESSION SCHEMES.

The various existing data compression techniques are efficient in natural recourses but in case of WSN, compression techniques should be optimal enough to make maximum use of constraint based equipment of WSN. Previous approach based of data precision k i.e. K-RLE compress the data on value of k, if value of k increases compression ratio is also high and resulting data loss from the process. As a comparison with RLE the simulation results shows that K-RLE achieves 40% more than that of the RLE scheme, and data loss is 50%. When the value of k =2, Hence, in K-RLE we can see that compression ratios fall down as the precision requirements are high. Hence it shows that there is great need of better compression ratio with minimal loss [13] [14] [15]

V. PROPOSED ALGORITHM

The proposed algorithm is enhanced from RLE and KRLE. AS RLE is lossless compression technique in which If a sensor node transfer data I happen n successive times in input data we change the n amount with single in pair. Where as in KRLE, K is supposed as a precision, If a data i or data among i+K with i-K occur n successive times in input stream, we change n amount with single nd pair [16]. In this k introduces as parameter Modification in k precision make the difference i.e. k =1, k=2 moreover it consider no difference among the data i, i-k, i+k. But in our proposed NK-RLE Nodes are homogeneously distributed and instead of sending sense data by node every time we send the difference between the previous sense data and new sense data rather than sending the original new data

The central idea behind the algorithm is based on K-RLE if the data i lies between i+k and i-k occur many times successive times replace it with ni K be a number, which is a precision parameter. Here K is denoted as: $\Delta=k/\sigma$; i. our proposed scheme is a lossy compression scheme, except at the level of user according to the application. In the proposed algorithm for sending nd1,nd2,nd3 repeating value n is send once i.e. nd1,d2,d3 which further enhanced of KRLE,which conserve the energy and life time of the network .Hence transition data will get reduce and it will take less data bytes and ultimately will increase the efficiency of sensor network and further as compare to K-RLE instead of sending d , d+k ,d-k .we are sending difference between the sensed data and average value of previous data which reduces bytes of data i.

The detail working of algorithm is shown in flowchart. We assume that in beginning at each node level previous value will be N prev= 0 and in each round val_i =N(i)sense- N prev

Where vali is the result value to be send by node and N prev is the average of the previous saved value and N (i) sense is the new value.

Now to check the repetitive value in list of data we will check the element if it lies between new val-k<store<newval+k if yes it will initialize the counter Rj=Rj+1 and it will store the value j in store variable now to calculate nprev divide nprev by c + nprev and initialize count =1 m =1 and temp r1 =0 and intilize compressed array to count repetitive value and m pointer type variable will check until end of list and val of temp r1 will be compared with r1+1 if both are equal it will increase the counter m and stored the value in compressed array and go on .which will gives us count of repetitive value with value i.e. 3d1,3d2,3d3 will be send as 3 d1,d2,d3.

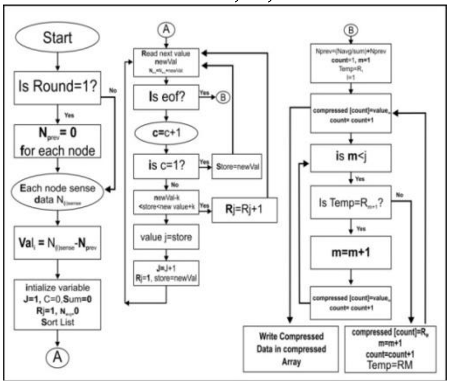


Fig. 1. Flow Chart of proposed algorithm

Detail working of algorithm is shown in flowchart. we assume that in beginning at each node level previous value will be N prev= 0 and in each round val_i =N(i)sense- N prev where vali is the result value to be send by node and N prev is the average of the previous saved value and N (i) sense is the new value . Now to check the repetitive

In our algorithm, firstly each node computes data as given in algorithm compute node _data.

In this, each node will sense the data N_{sense}. If round is one, it means the network is running for first time. So, there will be no previous value. So data will be sensed data

$N_{data}= S_{ense}$

But for other rounds, there will be some previous stored value. So, data will be difference between sensed data and previous stored data.

$N_{data}= N_{sense}-N_{prev}$

Algorithm compute_node_data (node)

This procedure computes the data of each node that will be sent

Node will sense the data N_{sense}
If (round! = 1)
$N_{data}= N_{sense}-N_{prev}$
Nprev is the average of previous sensed data
Else
$N_{data}= N_{sense}$
endif
Return N_{data}

Procedure sort (data [])

This will sort the data in ascending order

```
Cou= count (data [])
For (i=1; i<=cou-1; i++)
{
min-i
For (j=i; J<=Cou; j++)
{
If (data [min]> data[j])
Min=j
}
Exchange data [min] with
data[i]
}
Return data []
```

which will gives us count of repetitive value with value i.e. 3d1, 3d2, 3d3 will be send as 3 d1,d2,d3.

Procedure count (data [])

The procedure "count" will count the number of element in data array. Firstly, cou variable is declared to zero. While loop will continue till the data array is not empty. In each iteration, cou variable is increased by one.

```
set cou=0
While (data [cou] in not empty)
Cou++
End while
Return cou
```

Procedure computes repetition (k)

This procedure will compress the data of group such that rather than sending compute data, if sense value is repeated a number of times or in range value –k to +k, we will send value and number of times values.

```
    Set data [] =sort (group_data [])
    Set x=count (data [])
    Set i=1
    Set j=1
    Set cou[j] =0
    Set value=data[i]
    Set data1 [j] =value
    While (i<=x)
If (value-k=data[i] <=value + k)
    Set cou[j] =cou[j] +1
    Else
    j=j+1
    Value=data[i]
    data1 [j] =value
    cou[j] =0
    endif
    Set i=1
    End while loop
    return data1 [] and cou []
```

Value in list of data we will check the element if it lies between new val-k<store<newval+k if yes it will initialize the counter Rj=Rj+1 and it will store the value j in store variable now to calculate nprev divide nprev by c + nprev and initialize count =1 m =1 and temp r1 =0 and intilize compressed array to count repetitive value and m pointer type variable will check until end of list and val of temp r1 will be compared with r1+1 if both are equal it will increase the counter m and stored the value in compressed array and go on .

Algorithm computation final group

This algorithm will compress the data of node of group. It will return data that will send to sink node.

Set $data_1$[] =compute repetition (k)
Set x=count ($data_1$[])
Set j=1
Set i=1
Set compress[j] =data[i]
Set m=1
While (i<x)
If (cou_1[i]==cou_1[i+1])
Set i=i+1
Set m=m+1
else
Set i=i+1
Set j=j+1
Set compress[j]=m
Set j=j+1
Set compress[j]=data[i]
endif
end while
return compress[]

VI. SIMULATION RESULTS

Parameter settings

To simulate the proposed Data compression approach in wireless security network and to evaluate its performance. Table 1 show the network parameters, with their respective values which are used in the simulation.

Table- I: network parameters

Size of Network	hundred ×hundred m^2
Sensor node used	Hundred
Data aggregation energy	5nJ/bit/signal
Energy consumption for free space	$10pJ/bit/m^2$
Emp	$0.0013pJ/bit/m^4$

Table 1: Simulation parameters, and their respective values

A. Stability period

Stability period or First Node Dead In this, we measure the number of round in which first node of network gets unstable or dead at various energy levels

Table- II shows: First node death with various initial energy in our NK-RLE algorithm

INITIAL ENERGY	NK-RLE
0.0001	17
0.0002	30
0.0003	43
0.0004	60
0.0005	73
0.0006	86
0.0007	101
0.0008	101
0.0009	112
0.001	137

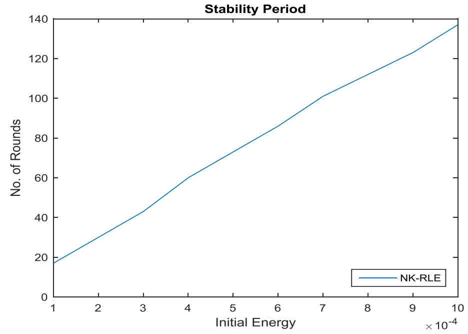


Fig.2. Stability period

Simulation results for stability period with respect to different energy levels are calculated. The above table shows the no of round of various node get alive in network or no of rounds the first node become unstable at which round .The results shows that as initial energy is increasing stability period of node is also increasing .i.e. at 0.0001 energy first node is getting unstable at only 17^{th} round where as at it is becoming unstable at 123th round at 0.0009 energy.

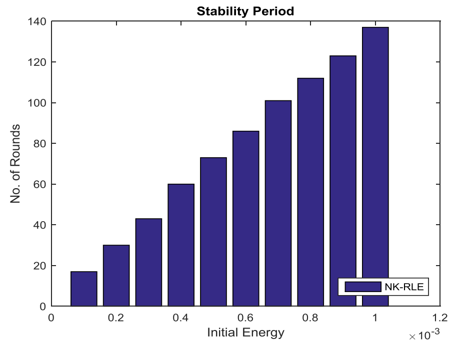


Fig.3. shows the analytical graph of simulation values of Stability Period of NK-RLE Data compression algorithm

B. Middle Node DeaD

In MND we measure the round and time gap from initial stage of network to death of 50 % nodes [17].

Table- III: Middle Node Dead (MND) with various initial energy in our NK-RLE algorithm

INITIAL ENERGY	NK-RLE
0.0001	481
0.0002	962
0.0003	1443
0.0004	1924
0.0005	2404
0.0006	2885
0.0007	3366
0.0008	3847
0.0009	4327
0.001	4808

Of network take place. Simulation results for MND with respect to different initial energy for Improved KRLE (NKRLE) Data compression algorithm as shown in table 2 shows that at energy level 0.0001 the half of nodes get dead at 481 rounds and when energy level is 0.0009 half of nodes are getting dead at 4808 rounds

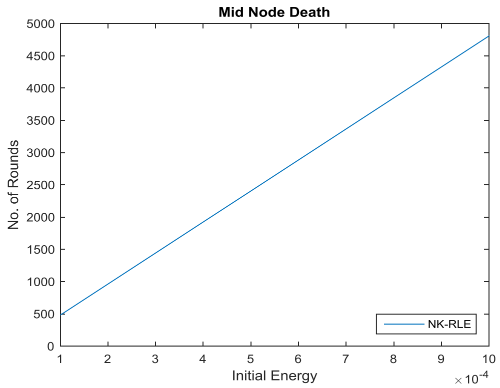


Fig.4.Middle node death

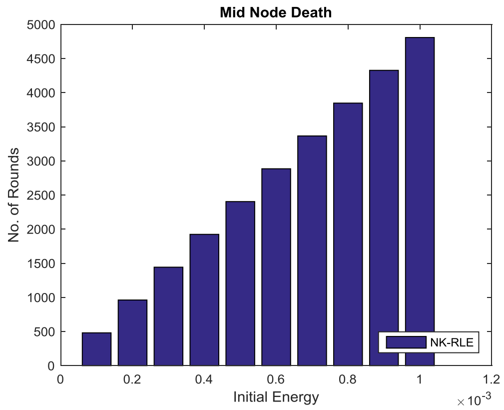


Fig.5. shows the analytical graph of simulation values of MND of Improved KRLE (NKRLE) Data compression algorithm

C. Network Life time

In Network lifetime or last node death (LND) we measures time gap from starting of the network operation to the death of last sensor node alive Simulation results for LND with respect to different initial energy for Improved KRLE Data compression algorithm as shown in table 3 shows that at energy 0.0001 last node is getting dead at 500th round and at energy level 0.0009 last node is getting dead at 4507th round

Table- IV: Last Node Dead (LND) with various initial energy in our NK-RLE algorithm

INITIAL ENERGY	NK-RLE
0.0001	500
0.0002	1001
0.0003	1448
0.0004	2005
0.0005	2508
0.0006	3005
0.0007	3524
0.0008	4102
0.0009	4507
0.001	5000

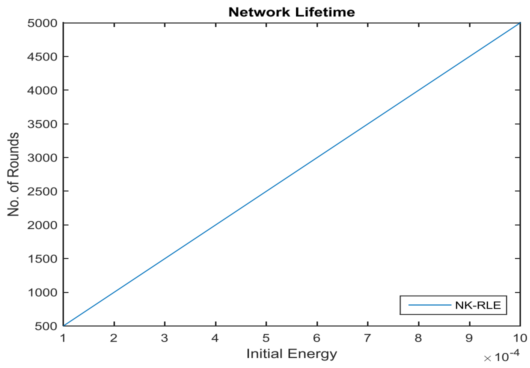


Fig.6. Shows the analytical graph of comparison of simulation values of LND of Improved KRLE (NK-RLE) Data compression algorithm

D. Comparison Ratio

In this we have calculated the compression ratio of new improved KRLE (NKRLE) Algorithm by changing energy level in simulation software .Simulation results for Compression Ratio with respect to different initial energy for Improved KRLE (NKRLE) Data compression algorithm as shown in table no 4 as shows that compression ratio is 0.721 at initial energy0.0001 and 0.682 at 0.0009 due to lossless compression process and it far better than previous algorithms which is proved in the latter part of this paper .

Table- V: Compression Ratio with various initial energy in our NK-RLE algorithm

INITIAL ENERGY	NK-RLE
0.0001	0.721
0.0002	0.6994
0.0003	0.701
0.0004	0.681875
0.0005	0.6868
0.0006	0.682325
0.0007	0.684486
0.0008	0.672519
0.0009	0.682511
0.001	0.67808

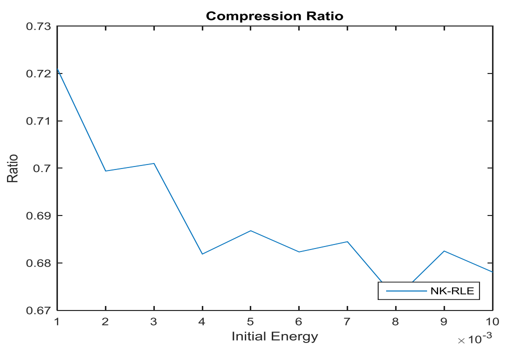
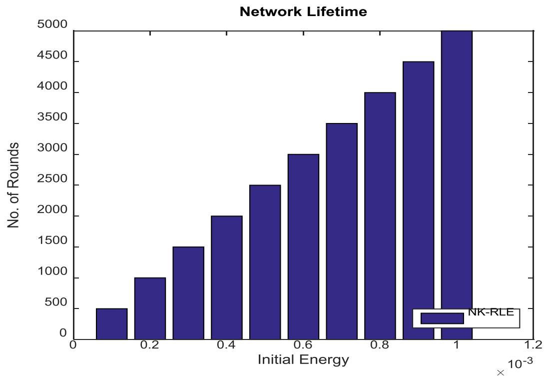


Fig.7.Compression ratio of NK-RLE



VII. COMPARISON OF K-RLE AND NK-RLE AND DISCUSSION

A. Stability period

The compartative study of KRLE AND NKRLE with respect to stabilty period shows that first node in KRLE and NKRLE is geeting dead at 11th round and 17th round respectively at energy level 0.001 and at energy level 0.0009 it is getting dead at 78th and 137th round . which proves that stability period of NKRLE is far better than KRLE.

Table- VI: Comparison table of K-RLE and NK-RLE for stability period with various initial energy in our NK-RLE algorithm

INITIAL ENERGY	K-RLE	NK-RLE
0.0001	11	17
0.0002	17	30
0.0003	27	43
0.0004	34	60
0.0005	43	73
0.0006	51	86
0.0007	59	101
0.0008	66	112
0.0009	70	123
0.001	78	137

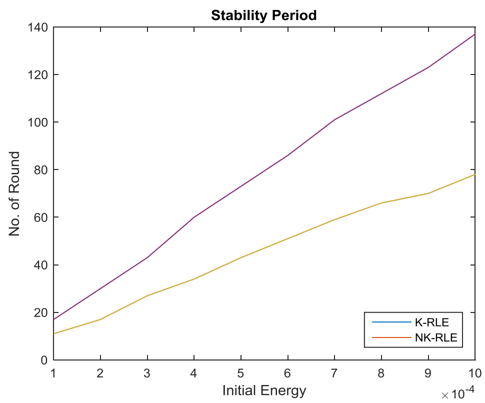


Fig.8. Shows the analytical graph of comparison of simulation values of Stability Period of K-RLE with N-KRLE.

B. Middle Node Dead (MND)

50% or half of nodes in network in KRLE are getting dead 241 round and where as in NKRLE are getting dead at 481 round at 0.0001 energy and at 2164th round in KRLE and 4327th in NKRLE at 0.0009 energy level. Compartive study of result prove thatNKRLE is showing energy efficency . as nodes are getting alive for longer period of rounds and hence making the algorithm more reliable .

Table- VII: Comparison table for middle node death of K-RLE and NK-RLE with various initial energy in our NK-RLE algorithm

INITIAL ENERGY	K-RLE	NK-RLE
0.0001	241	481
0.0002	481	962
0.0003	722	1443
0.0004	962	1924
0.0005	1202	2404
0.0006	1443	2885
0.0007	1683	3366
0.0008	1924	3847
0.0009	2164	4327
0.001	2404	4808

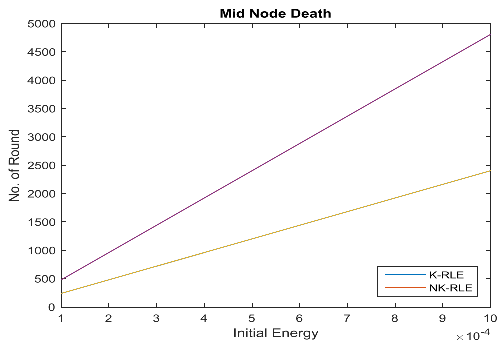


Fig. 9. Shows the analytical graph of comparison of simulation values of Middle node dead of K-RLE with N-KRLE.

C. Network Lifetime

The simulation result of both the algorithim are shown in table .As result shows that in our proposed algorithim nodes are getting alive in approxmitaly double round at almost each energy level and efficieny of network is double . we have taken various values by varing energy level starting from 0.0001 to 0.00001 in kRLE last node dead at 2500 round and in case of NKRLE at 5000 round .

Retrieval NumberF9263088619/2019©BEIESP
DOI: 10.35940/ijeat.F9296.088619
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

OPEN ACCESS
International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August, 2019

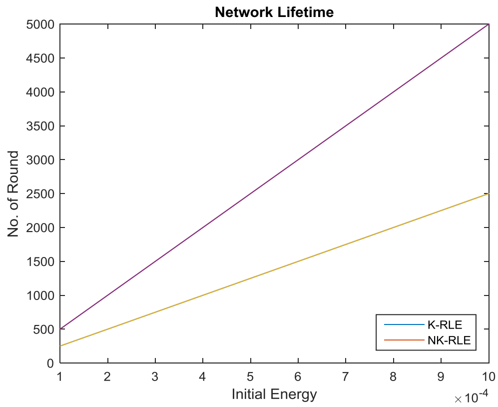


Fig.10. Network life time

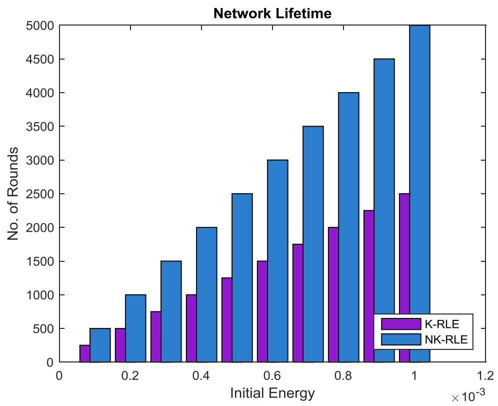


Fig.11. Shows the analytical graph of comparison of simulation values of Network life time of K-RLE with N-KRLE.

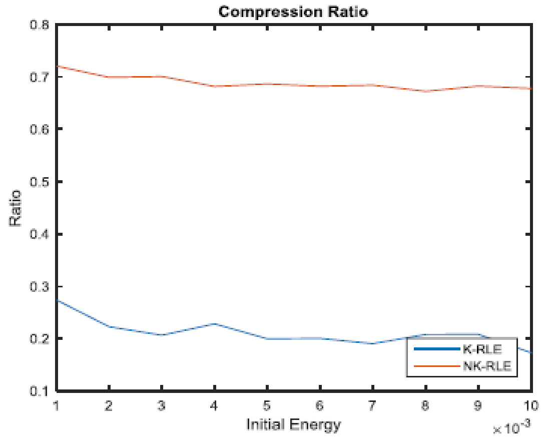
D . Compression ratio

The compression ratio of KRLE is based on value of k (k=2) and each node is sending the data to sense node .and In case of NKRLE we are sending the avg value which is calculated at node level by calculating difference between current and previous value and sening only difference not the value that why the result shows huge diffierence in compression ratio of both algorithim . but due to constraint resources in wirless sensor network higer compression ratio conserve the energy.

Table- VIII: Comparison table for network life time of K-RLE and NK-RLE with various initial energy in our NK-RLE algorithm

INITIAL ENERGY	K-RLE	NK-RLE
0.0001	0.2738	0.721
0.0002	0.2226	0.6994
0.0003	0.2064	0.701
0.0004	0.22815	0.681875
0.0005	0.19952	0.6868
0.0006	0.2002	0.682325
0.0007	0.190114	0.684486
0.0008	0.207775	0.672519
0.0009	0.208044	0.682511
0.001	0.17268	0.67808

Fig.12. Compression ratio of K-RLE with NK-RLE



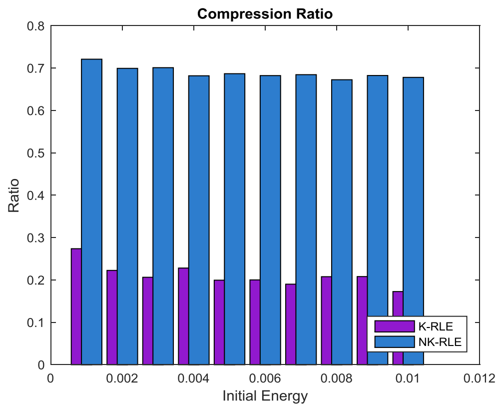


Fig. 13. Shows the analytical graph of comparison ratio of simulation values of K-RLE with N-KRLE.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we have evaluated numerous data compression algorithms We have compared a KRLE-based data compression algorithm for WSN, we have introduced a new algorithm inspired from RLE and K-RLE named NK-RLE ,which increases the ratio compression compared to RLE and K-RLE The compression ratio of KRLE is based on value of k (k=2) and each node is sending the data to sense node .and In case of NKRLE we are sending the Average value which is calculated at node level by calculating difference between current and previous value and sening only difference not the value that why the result shows huge diffierence in compression ratio of both algorithim . The simulation results shows that proposed algorithim is better than the previous existing algorithims .Future work will focus on the scalability issues of the proposed enhancements

REFRENCES

1. Carlos-Mancilla, M., López-Mellado, E., & Siller, M. (2016). Wireless Sensor Networks Formation: Approaches and Techniques. Journal of Sensors, 2016, 1–18.doi:10.1155/2016/2081902.
2. Ayinde, B. O., & Barnawi, A. Y. "Energy Conservation in Wireless Sensor Networks using Partly-Informed Sparse Autoencoder". IEEE Access, 1–1, doi:10.1109/access.2019.2917 322, 2019.

3. Reinhardt, A., Christin, D., Hollick, M., & Steinmetz, R. (2009). "On the energy efficiency of lossless data compression in wireless sensor networks". IEEE 34th Conference on Local Computer Networks.doi:10.1109/lcn.2009.5355014, 2009.
4. Marwen Roukhami, Younes Lahbib, Abdelkader Mam, "C-RLE: Energy efficient compression approach for Wireless Sensor Networks ", IJCSNS International Journal of Computer Science and Network Security, VOL.18 No.6, June 2018 pg. 49-48.
5. S. Jancy, 2 Dr. C. Jaya Kumar " Sequential Coded Data Compression Techniques for Wireless Sensor Networks" International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 9, September- 2014, ISSN: 2278-0181
6. Rawat, P., Singh, K. D., Chaouchi, H., & Bonnin, J. M. "Wireless sensor networks: a survey on recent developments and potential synergies". The Journal of Supercomputing, 68(1), 2013 pg.1–48. Doi: 10.1007/s11227-013-1021-9.
7. R. Hou, K. Lui, F. Baker, and J. Li, "Hop-by-Hop Routing in Wireless Mesh Networks with Bandwidth Guarantees," IEEE Transactions on Mobile Computing, vol. 11, no. 2, pp. 264- 277, February 2012.
8. Reshma B. Bhosale, Rupali R. Jagtap "Data Compression Algorithm for Wireless Sensor Network" international research journal of multidisciplinary studies & sppp's, Vol. 2, Special Issue 1, March, 2016 ISSN (Online): 2454-8499.
9. Hamidreza Asgarizadeh and Jamshid Abouei "An Energy-Efficient SD-based LZW Algorithm in Dynamic Wireless Sensor Networks", IEEE, Doi- 978-1-4673-5634-3/13, 2013, pg. 1-6.
10. Farooq sunar mahammad, V. madhu Viswanatham. "Performance analysis of data compression algorithms for heterogeneous architecture through parallel approach", The Journal of Supercomputing, 2018
11. Dinesh Suresh Bhadane, Shailaja Y. Kanawade. "Comparative study of RLE & K-RLE compression and decompression in WSN", 3rd International Conference on Advanced Computing and Communication Systems (ICACCS), 2016.
12. Ruchi Gupta, Mukesh Kumar, Rohit Bathla "Data Compression - Lossless and Lossy Techniques" International Journal of Application or Innovation in Engineering & Management (IJAIEM) Volume 5, Issue 7, July 2016 ISSN 2319 - 4847
13. J. Uthayakumar, T. Vengattaraman, P. DhavachelvanJ. Uthayakumar et al.," A survey on data compression techniques: From the perspective of data quality, coding schemes, data type and applications", Journal of King Saud University – Computer and Information Sciences, 2018.
14. M.VidyaSagar, J.S. Rose Victor "Modified Run Length Encoding Scheme for High Data Compression Rate" International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 12, December 2013, ISSN: 2278 – 1323.
15. Amandeep Singh Sidhu, Er. Meenakshi Garg "Research Paper on Text Data Compression Algorithm using Hybrid Approach" International Journal of Computer Science and Mobile Computing, IJCSMC, Vol. 3, Issue. 12, December 2014, pg.01 – 10, ISSN 2320–088X
16. S. Sarika, S. Srilali S. Sarika et al "Improved Run Length Encoding Scheme for Efficient Compression Data Rate" International Journal of Engineering Research and Applications www.ijera.com ISSN: 2248-9622, Vol. 3, Issue 6, Nov-Dec 2013, pp.2017-2020
17. Deepali, Padmavati. "Improved energy efficiency semi static routing algorithm using sink mobility for WSNs", Recent Advances in Engineering and Computational Sciences (RAECS), 2014.

AUTHOR PROFILE

Mr. Narinder Singh had done Master of computer application from Maharishi dayanand university, Rohtak in 2004 and M.phil from choudhary Devi Lal University in 2008. He is currently pursuing Ph.D. and currently working as Assistant Professor in Department of Computer Sciences, Guru Nanak College, Budhlada since 2007. He is a member of computer society of India since 2010. He has published more than 10 research papers in national and international reputed journals and published three books. His main research work focuses on Data compression wireless sensor network, Network Security, Cloud Security and Privacy. He has 11 years of teaching experience and 5 years of Research Experience.

Dr. Harmaninderjit singh is an Assistant Professor and has more than 12 years of academic experience. He has done MCA degree from Punjabi University Patiala, Punjab and received the PhD Degree in Computer Science from Desh Bhagat University, Mandi gobindgarh, Punjab. He has published more than 17 papers in different Journals and conference proceedings. His research interest is in the areas of Open Source Software and Algorithm Design and is presently working on it.