

# Data compression report samples

[Technology](#), [Development](#)



## **Data compression algorithm for wireless sensor networks**

### **Abstract**

There is an increased use of wireless networks in organizations. There also the use of Wireless Sensor Network (WSN) that is used for monitoring the environment. The two basic activities that are involved in WSN include data acquisition and data transmission. The problem with the process of transmitting and acquiring tasks is that there is high power consumption. This needs the use of data compression so that there is reduced consumption of power. This paper will analyze algorithms that are used in Wireless Sensor Networks to undertake data compression. In particular, there will be the analysis of compression algorithms that are used in environments of ultra-low power and then propose a data compression algorithm. The proposed data compression algorithm is developed from Run-Length Encoding.

### **Introduction**

There have been developments and enhancements in technology so that there are low-power processing units that are used in communication. These developments have led to the production of communication nodes that are autonomous and are able to sense environmental data. They are able to compute and send the data by use of wireless medium to base station that is referred to as Sink. This data is used for future analysis and processing. This, thus, forms Wireless Sensor Networks(Sayood 2012).

One problem that is evident with Wireless Sensor Networks is that of power management. This power management issue has resulted in researchers of

scheduling sensor states. The technique of scheduling sensor state is a process of determining which sensor is likely to change its state. The different states of sensors include transmit, receive, idle, and sleep. The changes will be effective according to the communication needs(Sadler & Martonosi 2006).

The popular technique that is used for power management and reducing the constraint of power is making use of the sleep mode. This is where most of the parts of the sensor's transceiver are switched off. The radio transceivers that are found on board sensor nodes are the ones which consume the most power(Rissanen 1983). This, therefore, requires that transceivers are kept in switched mode most of the time so that they save energy. However, the use of sleep mode to reduce energy consumption will reduce the rate of receiving/transmitting data. This will mean that the communication in the network will be reduced. The question that comes to mind is how to keep the same rate of data transmission to the base station by reducing the number of transmission. This is the role that this research paper will play(Rodeh et al. 1981). It will focus on the introduction of in-network processing technique to reduce the energy that is consumed. In-network processing techniques are the procedures that enable the reduction of the amount of the data that is being transmitted in a network. This article addresses sensor networks. The popular in-processing technique that is known is data compression or data aggression. Data compression is the process of reducing the amount of data to be transmission. This will, in turn, reduce the time that it takes to transfer data in a sensor network. There are many algorithms that have been developed which helps in the reduction of the amount of data in the

transfer(Salomon 2004). These developments have led to the production of communication nodes that are autonomous and are able to sense environmental data. They are able to compute and send the data by use of wireless medium to base station that is referred to as Sink. This data is used for future analysis and processing. This, thus, forms Wireless Sensor Networks(Burrows & Wheeler 1994).

One of the limitations of the existing nodes of sensor networks and nodes include the limitations that come as a result of RAM and processing capabilities. The nodes of the sensor networks have limited capabilities to undertake data compression capabilities. This has resulted in adoption of data compression algorithms that have been there in existence. There are two types of data compression algorithms. These are the lossless and lossy data compression algorithms. The best algorithm that is of lossless nature for wireless sensor networks is that of S-LWZ. This has been widely used in the compression of data so that the transmission of the data cannot bring constraints to the network(Ziv & Lempel 1977).

The S-LWZ algorithm has been developed from LWZ algorithm which has been used popular for data compression. The algorithm is dictionary-based. All algorithms which are dictionary-based have been known to utilize the RAM extensively. These kinds of algorithms are not popular with Wireless Sensor Networks because of the limitations that come with sensor networks. Sensor networks have limited RAM capabilities and are not optimal when dictionary-based algorithms are used in the process. This has led to the development of generic data compression algorithms that work well and optimally with Wireless Sensor Networks (WSN). There is the development of

the Run Length Encoding (RLE) which is used in ultra-low power microcontroller product, TI MSP430, from Texas Instruments. This instrument has been used for monitoring environmental temperatures(Welch 1984). One of the challenges that come with Run Length Encoding is the fact that the results of the data compression will depend on the sources of data that has been found on the network. There has been the development of RLE-ST which is the application of RLE with a given set of data. This is an improvement of the RLE to enhance the usability of the data compression algorithm.

## **Different data compression algorithms**

There are different types of data compression algorithms which have been developed to enable data compression. The most popular dictionary-based lossless compression algorithm is that of LZW which was developed from LZ78. For Wireless Sensor Networks, the best data compression algorithm is S-LZW which is an enhancement of the popular LZW. This algorithm has been developed and adopted in Wireless Sensor Networks. The transmission of data in a Wireless Sensor Network requires that the process uses minimal energy. There will be analysis and comparison of various data compression techniques and an assessment of the various trade-offs that come with the different data compression algorithms(Nelson 1991).

## **S-LZW**

There are basic parameters that are integrated to the development of this data compression algorithm for Wireless Sensor Network. There is a block size of 528bytes which represents two flash pages. The algorithm divides the

block of data that has not been compressed into blocks of data so that these blocks of data can be compressed separately. There is also a 512 entry data dictionary. The algorithm starts the data compression process by initializing the dictionary so that it has the standard characters of the alphabet which have a representation of 256 entries of the data dictionary. The dictionary is re-initialized for every block that is being used in the compression process. There is the creation of a new entry in the dictionary by the input string. This is the reason as to why data that is to be compressed have limited size. There are various strategies that have been developed to eradicate the problem of full dictionaries. One of the options to solve this problem is to freeze the dictionary and to use it as-is and to compress the rest of the data that is found in the block. The other option is to reset the block and start the initialization from scratch. This problem is not common when the data block is small and thus the dictionary is not full.

There is also a mini-cache of 32 entries. This is used for getting the advantages of the repetitive nature of sensor data. For this to be achieved there is the addition of mini-cache to the S-LZW. The mini-cache is an index which is hashed and of size  $N$ . in this case,  $N$  is in the power of 2. This mini-cache is used to store dictionaries which have been created recently. The problem with this algorithm is that there is a need to have extensive RAM size which is why it is not sufficient for sensor networks where there is the use of RAM and processing capabilities.

## **Run Length Encoding algorithm**

This is another algorithm which is basic and was developed for general compression procedures. The idea that was used to develop this algorithm is

that if there is the occurrence of  $d$  data item  $n$  consecutive times in the input stream, this will mean that the  $n$  occurrence will be replaced with  $nd$  single pair.

The diagram is a graphical representation of the RLE algorithm which has applied on the temperature algorithms. Because of the fact that RLE algorithm depends on consecutive input streams of data, the results that come out of the algorithm depends on the source of data. For better results which depend on different data source statistics, there is the introduction of a new algorithm which is a development of RLE. This has been developed basing on RLE but with the addition of K-Precision. This algorithm is referred to as K-RLE.

## **K-Run-Length Encoding**

The development of this algorithm is based on the theorem where,  $K$ , being a number, so that if a data item  $d$ ,  $d+K$  or  $d-K$  will occur  $n$  consecutive times, seen in the input stream, there will be the replacement of  $n$  occurrences with a single pair which is  $nd$ .

In this new algorithm, there will be the introduction of  $K$  which is a precision.

The definition of  $K$  is as follows:

$$K = \frac{\delta}{\alpha}$$

In this case,  $\delta$  is the degree while  $\alpha$  is the ecartype where it represents some form of dataset. In this case if  $\delta$  is 0 degrees, then K-RLE will be equivalent to RLE.

The problem with the introduction of K-precision is that there is data loss in the process. This is because of the fact that K-RLE is a lossy algorithm while RLE is a lossless algorithm. The algorithm is lossless when assessed at the

user level because the user chooses  $\delta$  and having considerations that there is no major difference between  $d$ ,  $d+K$  or  $d-K$  which is in regard to the application. The diagram below shows the difference between the two algorithms.

## **Features of the K-RLE data compression algorithm**

Temperature plays a crucial role in algorithm analysis that is used for data compression. From research, it has been found out that as the latitude increases, there will be increased change of temperature. This means that different locations can be chosen and the behaviors of the algorithms that are used for data compression are studied. This shows that the temperature of the input streams of data that are used in the algorithm have different behaviors.

There is a difference in the use of different algorithms even if the variation of data compression is the same. There will still be different compression ratios in the data that will be compressed. From the various researches, it is clear that S-LZW is found to be better than RLE. The ratio becomes worse as the location of the experiment is done further from the equator. K-RLE can achieve higher ratios of compression at the cost of the precision. When the precision is increased, there will be increased ratios of compression. The data precision is increased with increased ratios of data compression. Better and higher compression ratios can be increased bit at the expense of the original data and the data which has been decompressed.

One notable feature of lossy algorithm is that is that as data is compressed and decompressed, there are higher possibilities of retrieving data that is different from the original data that had been fed to the algorithm. Although



this is the case, it is evident that the data resembles and they are close to the original. This is the reason why precision is chosen by the user basing on the application that is being used in the process. The precision that is used and commonly applied to K-RLE is 2. This means that most K-RLE algorithms use 2-RLE. With this precision chosen, it is evident that half of the original data have been modified when K is taken to be 2.

## **Energy consumption**

There is a need to analyze the consumption of energy of various data compression algorithm in comparison with K-RLE. While there is a better and preferred compression ration with the use of 2-RLE, it is found to consume a lot of energy. 2-RLE consumes 0, 42 mj while that of RLE consumes 0, 0053 mj. On the other hand, S-LZW consumes 0, 0224mj. In this comparison, it is evident that RLE consumes lesser power than the rest of the data compression algorithms. This is a trade-off between energy compression ratio and a good compression ratio. It is also important to note that whereas RLE and 2-RLE have constant energy consumption rates, there is increased consumption of energy for S-LZW as more changes on data is realized.

Although 2-RLE uses a lot of energy in compressing data, it uses very little energy in the decompression process. This is so with its use of 0, 0011mj in the process of decompressing data. S-LZW uses 0, 015mj on average. RLE, on the other hand, uses 0, 00165mj on average to decompress data. Like in the process of compressing data, S-LZW uses more data in the process of decompressing data when there are more changes realized on data.

Data compression is the process of reducing the amount of data to be transmission. This will, in turn, reduce the time that it takes to transfer data

in a sensor network. There are many algorithms that have been developed which helps in the reduction of the amount of data in the transfer. These developments have led to the production of communication nodes that are autonomous and are able to sense environmental data

## **Conclusion**

This paper has focused on the various data compression algorithms. The analysis has been based on undertaking them experiments on ultra-low power microcontrollers from Texas Instruments. The comparison has been based on the temperature changes. There has been comparison of the various algorithms. The major algorithm that has been the basis for comparison is that of dictionary-based data compression algorithm, S-LZW. The comparison has been done with RLE and K-RLE. The need to have a different algorithm has been based on the fact that S-LZW has issues when it comes to RAM and energy consumption of the sensor platform. There are constraints of processing power of the RAM. This has led to the creation of a new algorithm which makes use of lower energy and lower processing power of the RAM. This has been undertaken so that there is data compression with minimal energy consumption and processing power. A new algorithm has been introduced and proposed which is based on RLE. This new algorithm, K-RLE has been based on the increase of the precision that will determine the ratio in which the data compression will be undertaken. The user changes the compression precision basing on the data that is being compressed. The increase in K-value in the algorithm increases the ratio of the data compression algorithm. This is unlike RLE and S-LZW algorithms which do not have precision adjustments and provisions. With the use of 2-RLE, there

is the introduction of 40% data compression ratio. With this, RLE will introduce 50% data loss in the process. This is because the two algorithms are lossy in nature which means that they will lose data in the process. The S-LWZ algorithm has been developed from LWZ algorithm which has been used popular for data compression. The algorithm is dictionary-based. All algorithms which are dictionary-based have been known to utilize the RAM extensively. These kinds of algorithms are not popular with Wireless Sensor Networks because of the limitations that come with sensor networks. Sensor networks have limited RAM capabilities and are not optimal when dictionary-based algorithms are used in the process. The comparison has been based on the temperature changes. There has been comparison of the various algorithms. The major algorithm that has been the basis for comparison is that of dictionary-based data compression algorithm, S-LZW. The comparison has been done with RLE and K-RLE. The need to have a different algorithm has been based on the fact that S-LZW has issues when it comes to RAM and energy consumption of the sensor platform. This is the reason why care should be taken when undertaking the process of data compression. While 2-RLE algorithm is good when it comes to data compression ratio, it consumes a lot of energy in the process of data compression. There is the trade-off between energy consumption and data compression efficiency.

## **References**

- Burrows, M. & Wheeler, D., 1994. A block-sorting lossless data compression algorithm.
- Nelson, M., 1991. Data compression book.
- Rissanen, J., 1983. A universal data compression system. Information Theory, <https://assignbuster.com/data-compression-report-samples/>

IEEE Transactions on.

Rodeh, M., Pratt, V. & Even, S., 1981. Linear algorithm for data compression via string matching. Journal of the ACM (JACM).

Sadler, C. & Martonosi, M., 2006. Data compression algorithms for energy-constrained devices in delay tolerant networks. of the 4th international conference on .

Salomon, D., 2004. Data Compression.: The Complete Reference.,

Sayood, K., 2012. Introduction to data compression,

Welch, T., 1984. A technique for high-performance data compression.

Computer.

Ziv, J. & Lempel, A., 1977. A universal algorithm for sequential data compression. Information Theory, IEEE Transactions on.