



Compression Techniques for Real-Time Control and Non-Time-Critical Big Data in Smart Grids: A Review

Kamil Prokop * , Andrzej Bień 🕩 and Szymon Barczentewicz 🕩

Department of Power Electronics and Energy Control Systems, Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering, AGH University of Science and Technology, 30-059 Krakow, Poland; abien@agh.edu.pl (A.B.); barczent@agh.edu.pl (S.B.)

* Correspondence: kprokop@agh.edu.pl

Abstract: Significant amounts of data need to be transferred in order to optimize the operation of power grids. The development of advanced metering and control infrastructure ensures a growth in the amount of data transferred within smart grids. Data compression is a strategy to reduce the burden. This paper presents current challenges in the field of time-series data compression. This paper's novel contribution is the division of data in smart grids to real-time data used for control purposes and big data sets used for non-time-critical analysis of the system. Both of these two applications have different requirements for effective compression. Currently used algorithms are listed and described with their advantages and drawbacks for both of these applications. Details needed for the implementation of an algorithm were also provided. Comprehensive analysis and comparison are intended to facilitate the design of a data compression method tailored for a particular application. An important contribution is the discription of the influence of data compression methods on cybersecurity, which is one of the major concerns in modern power grids. Future work includes the development of adaptive compression methods based on artificial intelligence, especially machine learning and quantum computing. This review will offer a solid foundation for the research and design of data compression methods.

Keywords: data compression; smart grid; communication in power grids; real-time operating systems; signal processing; information theory; big data; cybersecurity; Internet of Things

1. Introduction

Efficient management of the power grid requires the operator to possess accurate information about the state of the system in as many points as possible, in order to have a precise input to control the system. A way to carry this out is to develop measurement infrastructure, add new measurement points, increase time and amplitude resolution, increase communication bandwidth, and use modern computation systems or modern actuators with faster reaction time and less disturbance introduced to the action of the system [1]. Implementing any of these actions requires incurring costs. At the same time, consumers expect energy to have the lowest possible price, especially in modern and developed markets, which allow energy trade by allowing many entities to trade in it. According to control theory, the role of a measurement system is to gather data about the state of a controlled system and supply it to the controller, which analyzes the data, extracts information from them and commands the actuator to act accordingly. The typical data acquisition and control system is presented in Figure 1.

One of the main challenges in the development of modern power grids is ensuring a sufficient supply of measurement data [2]. Given the inherent periodicity exhibited by signals within power grids, the process of compressing signal data emerges as a favorable approach to increase the throughput of the communication system and efficiently manage the required disk space. Data compression methods are very popular in domains such as



Citation: Prokop, K.; Bień, A.; Barczentewicz, S. Compression Techniques for Real-Time Control and Non-Time-Critical Big Data in Smart Grids: A Review. *Energies* 2023, *16*, 8077. https://doi.org/10.3390/ en16248077

Academic Editor: Ahmed Abu-Siada

Received: 19 November 2023 Revised: 11 December 2023 Accepted: 13 December 2023 Published: 15 December 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). image (JPEG, GIF, PNG, WebP), video (H.264, H.265, VP9) or audio (MP3, AAC, Opus) processing, as well as general information-to-data ratio optimization (ZIP, Gzip, LZ77, LZ78). This review article presents popular techniques and methods used to compress time series in power grids. The goal of this article is to provide information on popular components of data compression methods, to be a guide for the selection of appropriate methods and a resource for gathering knowledge about the topic. This article also presents an overview of recent research on data compression in smart grids. A novel contribution of this work is defining two areas (Table 1) of application of data compression techniques in the smart grid:

- 1. Static—compressing data at rest, or data that can be transferred without crucial time constraints. An example can be the storage of 24 h of measurements in order to perform disturbance trend analysis, consumption analysis, etc.
- Dynamic—compressing data in transit that have to be transferred under exigent timing constraints. An example of this application can be data that needs to be registered, transmitted, and processed within a few milliseconds in systems like short-circuit, over voltage, over current, or islanding protection.



Figure 1. Architecture of distributed measurement and control systems in power grids. Components marked with green are typical parts of big data transmission systems. Big data acquisition systems typically use large data buffers (gigabytes, even terabytes) on the sensor side. Components marked with blue are typical parts of real-time control systems, which typically use smaller buffers or operate in pure streaming mode. Dotted line symbolizes optional data flow in the system. D_s —data size, C_r —compression ratio, RD_s —raw data size.

The IEEE Standard for Synchrophasor Measurements for Power Systems (IEEE Std. C37.118.1-2018) [3] and the IEC/IEEE Communication Networks and Systems for Power Utility Automation: Precision time protocol profile for utility automation (IEC/IEEE 61850-9-3) [4] are two normative documents that provide precise timing constraints for real-time communication in smart grids [5,6].

1. Sub-microsecond level synchronization: According to IEEE Std C37.118.1-2018, the synchronization protocol should provide clock synchronization on the order of a microsecond. This means that the communication system should be able to maintain synchronization to a level of precision that is a few microseconds.

- 2. Timeliness: IEEE Std C37.118.1-2018 also specifies that the communication system should be able to deliver data and control signals within a few milliseconds to ensure the safety and reliability of the smart grid.
- Precision time protocol profile for utility automation: IEC/IEEE 61850-9-3 specifies the precision time protocol (PTP) profile for utility automation. PTP is a network protocol for clock synchronization, which can be used to synchronize the clocks of devices in the smart grid to a few microseconds.

Table 1. Data types in distributed measurement systems in smart grids with crucial requirements for data compression.

Data Type	Properties	Example of Application	Compression Requirements
Non-time critical power quality data	Large datasets, aggregated data, data content defined by norms and legislation	Electrical grid data collected by power quality meters, energy consumption data used for billing purposes, Wide Area Measurement System, Large Scale Phasor Measurement	Maximizing compression ratio, low data degradation, preserving data crucial for the application, scalability for large data sets
Real time control systems is Smart Grid	Constant stream of data, transients reporting, high resolution	Islanding protection, overvoltage and overcurrent protection, transient detection, short-circuit protection	low-latency, preservation of data crucial for application, low computational complexity, operation on blocks of data

Prioritization of data is a crucial issue in mission-critical systems like power grid control. Real-time data, which include information on current energy consumption, grid status, and equipment performance, may need to be prioritized over historical data for immediate decision-making. Historical data, such as long-term consumption patterns and equipment maintenance records, can be stored and analyzed separately. Critical data, such as information related to system faults, security breaches, or emergency situations, should be given the highest priority. Noncritical data, like routine monitoring information, can be processed with lower priority. Data related to load forecasting and prediction models may require special attention for accurate energy planning. Operational data, such as real-time grid conditions and equipment status, are essential for efficient management [7]. Segregation of measurement data is a crucial matter in data-intensive environments. In power grids, data are segregated using rule-based segregation or time-series analysis. Rule-based systems define criteria to categorize data into different priority levels. Rules can be based on factors such as data source, urgency, and criticality. Time-series analysis techniques distinguish between real-time data and historical trends. Algorithms such as ARIMA (AutoRegressive Integrated Moving Average) can help in forecasting and segregating data. More recently, edge computing architecture and machine learning techniques were introduced in order to improve the quality and efficiency of data segregation. Machine learning algorithms automatically categorize and prioritize data based on historical patterns and real-time conditions. Clustering algorithms can group similar data for efficient processing. Edge computing uses data processing locally at the source, reducing the need to transmit large volumes of raw data. Edge devices can perform initial data segregation and send only relevant information to central systems [8].

2. Data in Power Systems

The popularity of data compression methods grew along with the popularization of distributed computing and data storage systems since it reduced the ratio between the cost and benefit of such systems. In many fields, data compression is a fundamental part of their development. Complex and data-demanding information, such as audio or video files, has been a subject of data compression study for a long time. Real-time applications, such as bilateral voice transmission, also exist in cellular technology. In the domain of power systems, data compression was a niche topic most of the time, but the recent rise in the complexity of modern power systems, along with the need for detailed measurements, calls for optimization of data transfer [9]. The types of information that must be known about the system to be managed precisely are presented in Figure 2.



Figure 2. Data measured in power systems in time and frequency domain. Stripe on top presents the main reasons for measurement of power systems properties in listed time domains.

Using such a large amount of data makes data compression methods an attractive choice to increase system efficiency.

Data utilized for real-time control usually need to be compressed with methods that allow low-latency streaming and quick access to the data [10]. However, in most cases, the aggregated data that describe power quality in the system is not time critical and, therefore, can be compressed with methods that do not meet demanding time constraints but offer a higher compression ratio or lower computational efficiency instead [11].

Based on the principle of operation, data compression methods can be divided into two categories—lossy and lossless.

2.1. Lossless

Lossless compression algorithms are designed to preserve all the information contained in the original data. This means that when the data are compressed using these methods, they can be later decompressed to obtain an exact replica of the original data without loss or distortion [12]. Lossless compression methods often rely on the concept of entropy, which is a measure of the information content in the data. The basic idea is to identify and exploit patterns, redundancy, and statistical properties within the data to represent them more efficiently [13]. The most common techniques that are used to compress data without any losses are based on statistical encoding in order to reduce the entropy in the signal [14], thus reducing the quantity of the data or on dictionary methods, which replace repetitive data with shorter codes [15].

2.2. Lossy

Lossy data compression methods are techniques used to reduce the size of digital data by selectively removing some information that is considered less essential or perceptually less significant [16]. These methods are commonly employed in applications such as image and audio compression [17,18]. Due to that fact, most of the methods used in industry are based on deep analysis of human senses like sight and hearing and designed in a way to primarily lose the information that is less important to the receiver (like color-related information in JPEG, due to the fact that the human eye has more rod cells than cone cells, which makes information related to shape more crucial than information related to color to human receivers or frequencies bordering on the audible range in MP3 since they are less notable by humans) [19,20]. In lossy compression, the primary objective is to achieve high compression ratios by eliminating redundant or less important details. This involves quantization, where the data values are rounded or mapped to a smaller set of values. The discarded information is irretrievable during decompression, making lossy compression unsuitable for applications where data fidelity is critical, such as undertaking high-impact decisions about the state of the power grid. The key challenge in lossy compression is to achieve a balance between achieving significant compression while minimizing perceptual degradation [21]. While lossy compression offers substantial size reduction, it is crucial to use it judiciously in scenarios where slight data loss is acceptable. Historically, most lossy data compression methods were developed for the information received by humans, like video, audio, or image. In most cases, the information from the power grid does not need to be directly understood by humans, rather than by other IoT devices [22]. Direct transfer of methods from different domains should be carried out with caution. Lossy data compression methods are also specific to their application, which creates a field of data compression methods that should be tailored to be received and processed by machines, not by humans. The development of a lossy data compression method requires a deep knowledge of the system and the information contained in the data, which makes lossy compression algorithms harder to implement, but may offer a more optimized smart grid supervision system in the end [23].

3. Algorithms Used in Smart Grid Data Compression

Within the domain of information theory and data compression, two fundamental concepts, source coding [24] and entropy coding [25,26], play a pivotal role in optimizing data representation and transmission (Table 2). Source coding, often termed data compression or signal compression, serves as the foundation for reducing the size of datasets while preserving their essential information [27]. This process involves eliminating redundant details and superfluous information. For example, the ubiquitous Huffman coding technique assigns variable-length codes to characters in a text document based on their frequency of occurrence [28]. The most frequently used values receive shorter codes, resulting in efficient compression of text data.

Entropy coding, on the other hand, represents a specialized subset of source coding that delves into the statistical properties and probabilities inherent in the data source. It takes advantage of principles from information theory and probability theory to optimize compression efficiency. For example, arithmetic coding is an entropy coding method that encodes data based on its cumulative probability distribution [29]. By assigning shorter codes to more likely symbols, compression ratios are achieved that approach the entropy of the data source [30]. This is particularly valuable in lossless compression applications, such as event-related data compression, where exact reconstruction is essential.

Source coding methods such as Run-Length Encoding (RLE) can efficiently reduce the size of binary images by encoding consecutive runs of identical bytes as a single value [31]. Meanwhile, entropy coding techniques, including Huffman coding or arithmetic coding, further compress the data by exploiting the statistical properties of sample values. In essence, source coding and entropy coding represent essential tools in the data compression toolkit, each offering unique strategies to optimize the representation and transmission of information, with applications spanning from text and images to audio and video data [27,32].

3.1. Run-Length Encoding (RLE)

Run-Length Encoding (RLE) is a lossless compression algorithm used in power grid signals analysis to efficiently represent time-series data from voltage and current sensors [33]. During the encoding phase, consecutive identical values in the signal are compressed into

6 of 26

a single value and a count, reducing storage and transmission requirements (Algorithm 1). The compressed data, accompanied by relevant metadata, is stored or transmitted [34]. In the decoding phase, the original signal is reconstructed for analysis, allowing identification of voltage variations, transient events, and patterns of interest in the power grid. RLE is a simple algorithm; thus, it can be implemented at a low resource cost [35].

Algorithm 1 Run-Length Encoding (RLE)

$compressed_output \leftarrow ""$	▷ Initialize the compresse	ed output string
$current_character \leftarrow input[0]$	⊳ Initialize the c	urrent character
$count \leftarrow 1$	⊳ Initializ	the count to 1
for $i \leftarrow 1$ to length $(input) - 1$ do		
if $input[i] = current_character then$		
$count \leftarrow count + 1$		
else		
$compressed_output \leftarrow compressed_output$	$tput + current_character + c$	count
$current_character \leftarrow input[i]$		
$count \leftarrow 1$		
end if		
end for		
$compressed_output \leftarrow compressed_output + comp$	current_character + count	▷ Append the
character with counter		**
return compressed output		

Sliding-Window Run-Length Encoding (SW-RLE) is an extended version of the classic Run-Length Encoding (RLE) algorithm, specifically designed for scenarios where consecutive identical values are encountered within a sliding window of data in a time series sequence, as often occurs in the context of power grids signals analysis (Algorithm 2). SW-RLE operates in two distinct phases [36].

During the encoding phase, a fixed-size sliding window is used to traverse the timeseries data. Within this window, a count variable is initialized to 1, and a result buffer is used to accumulate the encoded data. At each position of the sliding window, a comparison is made between the values within the window. If all values within the window are identical, the count is incremented to represent the consecutive run of identical values. In cases where variations exist within the window, the value within the window and its count are appended to the result buffer. This process continues as the sliding window moves forward, covering all data in the time series. The result buffer eventually holds the compressed data, effectively representing consecutive identical values within the sliding window.

The decoding phase, on the other hand, starts at the beginning of the encoded data. An output buffer is initialized to store the decoded signal. The encoded data are then iterated through, processing pairs of values (element, count). For each pair, the element is appended to the output buffer count times, effectively reconstructing the original time-series data. This decoding process continues until the end of the encoded data is reached, producing the final output buffer with the uncompressed time-series data.

In the domain of analysis of power grid signals, SW-RLE is particularly valuable when analyzing sliding windows of continuous signals, where localized patterns and anomalies are of interest [37]. The adaptability of SW-RLE allows the adjustment of sliding window sizes to capture patterns of different durations, making it versatile for various analysis requirements. Importantly, SW-RLE preserves the integrity of the original data within the sliding windows, ensuring that no localized information is lost during the compression and decoding processes. This makes it a powerful tool for focused and localized analysis within the broader context of power grid signals.

Algorithm 2 Sliding-Window Run-Length Encoding (SWRLE)

 $compressed_output \leftarrow ""$ Initialize the compressed output string windowStart $\leftarrow 0$ Initialize the start of the sliding window while *windowStart* < length(*input*) do $windowEnd \leftarrow min(windowStart + windowSize, length(input))$ ▷ Window end $subsequence \leftarrow input[windowStart : windowEnd]$ ▷ Get the subsequence $runLength \leftarrow 1$ for $i \leftarrow 1$ to length(subsequence) - 1 do if subsequence[i] = subsequence[i-1] then $runLength \leftarrow runLength + 1$ else $compressed_output \leftarrow compressed_output + subsequence[i - 1] + runLength$ *runLength* $\leftarrow 1$ end if end for $compressed_output \leftarrow compressed_output + subsequence[length(subsequence) - 1] +$ runLength $windowStart \leftarrow windowStart + length(subsequence)$ ▷ Move the sliding window end while return compressed_output

3.2. Huffman Coding

Huffman coding, a widely used data compression technique, also finds application in power grid signal compression [38]. It begins by collecting time-series data from power grid sensors, and then, based on the data's frequency distribution, constructs a Huffman tree that assigns shorter binary codes to common signal values and longer codes to less frequent ones (Figure 3). This tailored encoding significantly reduces data size. The compressed data, along with information about the Huffman tree structure, are stored or transmitted efficiently. During decoding, the Huffman tree is utilized to reconstruct the original signal values, preserving data accuracy for critical power grid applications, including grid monitoring and fault detection.



Figure 3. Structure of binary tree created during encoding. Red circles are nodes, arrows are labeled with consecutive code signs, blue circles with letters contain encoded symbols.

Huffman coding can significantly reduce repetitive data size, which is useful in the domain of power grids [39]. The algorithm works efficiently with large datasets, which makes it a good choice for compressing large datasets. For streaming time-critical data, the choice of dictionary (binary tree) is crucial, since frequent changes in the dictionary increase the load on the communication channel and might result in low or even negative compression ratio [40].

3.3. Lempel-Ziv-Welch

The LZW algorithm (Algorithm 3) works by reading a sequence of symbols, grouping them into strings, and converting the strings into codes. The codes take less space than the strings they replace, achieving compression [41]. The algorithm uses a code table, with 4096 as a common choice for the number of table entries. Codes 0–255 in the code table are always assigned to represent single bytes of the input file. As the encoding continues, LZW identifies repeated sequences in the data and adds them to the code table [42].

Algorithm 3 Lempel–Ziv–Welch algorithm steps

```
P = \text{first input character}
while not end of input stream do
C = \text{next input character}
if P + C \in \text{string table then}
P = P + C
else
output the code for P
add P + C to the string table
P = C
end if
end while
output code for P
```

LZW takes advantage of short patterns. Due to the fact that power grids function in a periodic mode, this can be a useful way to compress data from a power grid [39]. This algorithm can also facilitate the observation of the power grid with periodic disturbances, because a sequence of samples may follow the same model if they register a distortion caused by the same reason [43].

3.4. Discrete Cosine Transform (DCT)

The discrete cosine transform (DCT) is extensively used in power systems data compression. Data from power systems, such as electrical quantities such as voltage, current, and power, often have a high-frequency component [44]. DCT can be used to compress these data by transforming them into the frequency domain, where the high-frequency components can be discarded if they do not contribute significantly to the overall data [45]. Due to its nature, data processed by DCT have more applications than sole compression:

Data compression: DCT can be used to compress the data of the power system before storage or transmission. For example, the DCT can be used to compress the voltage profile data, which is a sequence of voltage values at different points in time. By

- discarding the high-frequency components of the DCT, the data can be significantly compressed without significant loss of information.
 Signal processing: DCT can be used in signal processing in power systems. For example, it can be used in the analysis of electrical disturbances, where DCT can be used to transform disturbances into the frequency domain, making it easier to analyze
- used to transform disturbances into the frequency domain, making it easier to analyze and diagnose disturbances [46].
 Predictive analytics: DCT can be used in predictive analytics in power systems. For example, it can be used to predict future power system conditions based on historical
 - data. By transforming historical data into the frequency domain using DCT, the

predictive model can focus on the low-frequency components of the data, which are more likely to be relevant to predict future conditions [47].

• Control systems: DCT can be used in control systems in power systems. For example, it can be used in the control of power distribution systems, where DCT can be used to transform control signals into the frequency domain, making it easier to design and implement the control system.

In all these applications, DCT is used to transform the data into the frequency domain, where the high-frequency components can be discarded, thereby reducing the size of the data and improving the efficiency of data compression [48], signal processing, predictive analytics, and control systems in power systems [49]. The signal is represented as a weighted sum of subsignals (Figure 4), which is lossless, but initially offers little or no compression. The data obtained are quantized as a weighted sum of the DCT matrix, where the weights are lower for higher frequency components. This approach increases the further compression using techniques such as RLE. DCT can be used to amplify the compression ratio received from other algorithms such as wavelet transform [50].



Figure 4. Basic components of discrete cosine transform [49].

3.5. Wavelet Transform

Wavelet transform is a powerful tool in data compression, especially in real-time systems. The process of wavelet transform involves separating the signal into different frequencies, which can then be compressed independently [51]. This is particularly useful in power-system-related data, where signals often exhibit low-frequency and high-frequency components [52].

The fast continuous wavelet transform (fCWT) is an open-source algorithm that separates scale-independent and scale-dependent operations, which is beneficial for real-time, high-quality, noise-resistant time-frequency analysis of nonstationary noisy signals [53].

In power systems, a wavelet-based data compression method can be used to compress the recorded data of oscillations [54]. This method selects the optimal wavelet function and decomposition scale according to the criterion of the minimum compression distortion composite index (CDCI). The most popular wavelet functions are the Haar, Daubechies, and Coiflet families (Figure 5). This balances compression performance and reconstruction accuracy [55].



Figure 5. Wavelets from Haar, Daubechies and Coiflets families. For each wavelet, the scaling function phi and the wavelet function psi are presented [56].

The wavelet function and decomposition scale can be selected directly according to the oscillation frequency, which is the most significant characteristic of oscillations. The amount of calculation in this method is much lower than that in methods that require compressing and reconstructing signals with all candidates of wavelets and scales. This makes the wavelet-based data compression method for oscillations in power systems particularly efficient [57]. In the wavelet-based data compression method, the window length of data to be processed at a time deserves careful consideration. The length of a window defines the compromise between compression ratios and distortion rates [58].

The compression ratio depends on the oscillation frequency and can almost reach the compression ratio limit of scale i. The distortion rate is on the order of 10^{-4} in general and is always no more than 2×10^{-3} . The computational burden is not great for compression in storage.

Wavelet compression in general is a better solution for handling transients [59], which makes it more effective for unstable grids. It can be used as a component of the hybrid method, together with other algorithms, that work better for predictable periodic data, which is a common type of signal in stable power grids [21].

3.6. Differential Encoding

Differential encoding is a data compression technique that works by encoding the difference between consecutive data points rather than the data points themselves (Algorithm 4). This technique is particularly useful for time-series data where the values are often similar to those of their neighbors [60]. By encoding the differences, we can significantly reduce the amount of data that need to be stored, leading to more efficient storage and faster data retrieval [61].

The time complexity of this algorithm is O(n), where n is the number of data points. This is because we are performing a constant amount of work for each data point: calculating the difference and appending it to the list.

The space complexity of the algorithm is also O(n), since we are storing a new piece of data for each data point. However, the space required to store the encoded data is generally much less than the space required to store the original data, especially for time-series data with many similar consecutive values [62].

Algorithm 4 Differential encoding algorithm
Require: Time-series data D
Ensure: Encoded data E
$E \leftarrow \emptyset$
$E \leftarrow E \cup D[1]$
for $i \leftarrow 2$ to n do
$E \leftarrow E \cup (D[i] - D[i - 1])$
end for

Differential encoding is frequently used in compressing data in stable systems, where consecutive values do not change much. The compression ratio may deteriorate in systems with a high disturbance ratio. Differential encoding is perfect for reducing the size of data in stable periods of operation, but it is unlikely to be an optimal solution to compress transients [40]. This algorithm, in general, can be used to improve the performance of real-time systems during stable periods of operation and to significantly improve the compression of non-time-critical data in the processing of large power-quality datasets.

3.7. Burrows–Wheeler Transform (BWT)

The Burrows–Wheeler transform (BWT) is a block-sorting data transformation algorithm that is used in data compression (Algorithm 5). It is not a standalone data compression method, however; it is being used as a component with different solutions to increase the performance of other data compression algorithms [63]. BWT rearranges the input data in a way that similar data elements are grouped together. This property of BWT is exploited in the Burrows–Wheeler transform compression (BWT-C) algorithm, which is a data compression algorithm that is particularly effective for data with long repetitive sequences [64].

Algorithm 5 Burrows–Wheeler transform (BWT)
Require: string
Ensure: BWT_string, last_column_index
$rotations \leftarrow generate_rotations(string)$
sorted_rotations \leftarrow sort(rotations)
$BWT_string \leftarrow concatenate(sorted_rotations)$
$last_column_index \leftarrow find_last_column_index(string, BWT_string)$
return BWT_string, last_column_index

In the context of power-system-related data, BWT-C could be particularly useful. Data from power systems often contain long sequences of similar values, such as multiple readings of the same power consumption or generation value. Using BWT, these similar values are grouped together, which can significantly reduce the size of the data. BWT can also be used for methods that compress and encrypt data sent in power grids [65].

The BWT-C algorithm works in several stages. First, the input data are transformed using the Burrows–Wheeler transform, which rearranges the data in such a way that similar data elements are grouped together [66]. This is carried out by sorting the input data according to the characters that follow each character in the data.

The transformed data are then compressed using a lossless compression algorithm, such as Run-Length Encoding (RLE) or move-to-front transform (MTF). These algorithms take advantage of the fact that similar values are grouped together in the transformed data to further compress the data [67].

Finally, the compressed data are encoded into a format that can be easily stored or transmitted [40]. This can be carried out using any suitable encoding method, such as Huffman coding or arithmetic coding [68].

The use of BWT-C in real-time systems is particularly interesting. In real-time systems, data are continuously generated, and it is necessary to compress these data in real time to avoid storage overflow or transmission delay. Using BWT-C, it is possible to compress large amounts of data from power systems in real time, making it suitable for use in real-time systems [69].

In conclusion, the Burrows–Wheeler transform can be used to compress data related to power systems by rearranging the data in such a way that similar values are grouped together [70]. This can significantly reduce the size of the data, making it suitable for use in real-time systems. The use of the Burrows–Wheeler transform in data compression is a complex process that involves several stages, but it can be highly effective when used correctly.

3.8. Move-to-Front (MTF) Encoding

Move-to-front transform (MTF) is a data encoding technique designed to enhance the performance of entropy encoding techniques of compression [71]. The MTF transform works by maintaining an ordered list of legal symbols (for example, a to z in case of English text) [72]. The process involves searching for the input character in the list, printing the position at which the character appears in the list, and then moving that character to the front of the list (Algorithm 6). This process is repeated until indexes for all input characters are obtained [73].

Algorithm 6 Move-to-front encoding (MTF)

```
Require: input_text, len_text, list
Ensure: output_arr
for i = 0 to len_text do
    output_arr[i] ← search(input_text[i], list)
    moveToFront(output_arr[i], list)
end for
return output_arr
```

The MTF transform is particularly effective in reducing the entropy of a message by exploiting the local correlation of frequencies [74]. This means that letters recently used stay at the front of the list, resulting in a large number of small numbers such as "0" and "1" in the output if the use of letters exhibits local correlations [74].

However, not all data exhibit this type of local correlation. For some messages, the MTF transform may actually increase the entropy. However, an important use of the MTF transform is in Burrows–Wheeler transform-based compression. The Burrows–Wheeler transform is very good at producing a sequence that exhibits local frequency correlation from text and certain other special classes of data [63]. Compression benefits greatly from following up the Burrows–Wheeler transform with an MTF transform before the final step of entropy encoding [73].

One problem with the basic MTF transform is that it makes the same changes for any character, regardless of frequency, which can result in decreased compression, as characters that occur rarely may push frequent characters to higher values [75]. Due to the fact that there is a risk of negative compression, especially in grids with high distortion, proper

algorithm calibration is needed. Various alterations and alternatives have been developed for this reason. One common change is to make it so that characters above a certain point can only be moved to a certain threshold [76]. Another is to make an algorithm that runs a count of each character's local frequency and uses these values to choose the characters' order at any point. Many of these transforms still reserve zero for repeat characters since these are often the most common in data after the Burrows–Wheeler transform. In the context of real-time systems operating in the domain of power grids, the MTF transform can be particularly useful in systems where data are frequently updated or where there is a high degree of local correlation in the data [77].

3.9. Real Time Compression—Prequisities

In general, selecting the right data compression method for a system is based on balancing compression ratio, data distortion, and computational complexity [78]. However, in the domain of power systems, data that are being used for non-time-critical purposes mostly focus on achieving a high compression ratio with low data distortion. Algorithm complexity is rarely a case for such an application, since long computing times are acceptable, for the gain of reduced communication medium usage. Real-time systems give much more constraints on the selected method [10,79].

- Understanding of the application domain: In power grid applications, a thorough
 understanding of data types, characteristics, and permissible loss in data fidelity is
 crucial for effective data compression [80]. Knowledge of the properties of the power
 system and the purpose of compressed data is crucial. The systems used to protect the
 grid shall not lose transient data that may lead to false negatives. Metrology devices
 that are used to monitor systems with constant distortion should be carefully analyzed
 before using methods that result in negative compression.
- Timing constraint analysis: Real-time power grid systems have stringent timing requirements. It is essential to analyze task scheduling, response time, and deadline constraints to select appropriate compression techniques. This is a complex parameter to analyze since it requires deep knowledge of the data being compressed and computational capabilities of metrological system [81].
- Compression algorithm selection: The choice of compression algorithms must consider the trade-off between compression ratio and computational overhead. Customized real-time compression algorithms may be necessary to optimize performance. Stream algorithms in most cases offer lower latency, but also a lower compression ratio. In the case of block algorithms, block size should be carefully considered in order not to violate real-time constraints due to the long time needed to fill the buffer [82].
- Hardware and software considerations: Hardware and software platforms play a vital role in compression [83]. Understanding their capabilities and limitations is crucial to selecting and implementing compression techniques. Many modern embedded systems offer hardware acceleration for selected data compression methods [84], which significantly improves their performance [83,85]. Some software platforms also implement data compression algorithms on the lower layers of their architecture, which makes their use easier and more efficient. Field-programmable gate arrays (FPGAs) are a frequent choice for low-level acceleration of data compression algorithms in power grids [86–88].
- Error detection and correction: Robust error detection and correction mechanisms are essential to ensure data integrity, especially in noisy power grid environments prone to transmission errors [89]. The exact meaning of error varies between applications, but in general, should be considered a violation of the purpose of the system. For systems reacting to transients reporting false detection or omitting the actual event, this can be considered an error from the functional point of view. For monitoring systems used for power quality analysis, a difference in aggregated samples from real values is the most critical measure of an error [90].

- Validation and testing: Rigorous validation and testing procedures, including worstcase execution time analysis and simulation, are essential to confirm that the compression scheme meets real-time requirements in power grid applications [91]. Due to the network nature of power grids, every device used shall be compliant with legislation. Validation of such systems is a complex, long, and costly process; hence, special effort should be carried out during the development phase in order to avoid test failure.
- Efficiency and reliability: Meeting these prerequisites forms a strong foundation for enhancing the efficiency and reliability of data compression in real-time power grid systems [92]. Many power systems are used in remote locations with limited ability to monitor their operations. The design of such systems should also take into account the long period of their usage, and compliance with other systems that can be used in the future in case of expansion of modernization of adjacent systems [93].

3.10. Error Resilience

In the context of power grids, error resilience in data compression algorithms is crucial for real-time systems. This is due to the high volume of data generated by power grids, which can lead to data congestion if not properly managed [94]. The data generated include information about power generation, transmission, and distribution, as well as information about potential faults and disturbances in the power grid.

One approach to managing these data is the use of real-time energy data compression strategies. These strategies aim to reduce data traffic by compressing meter data efficiently. This is particularly important in smart grid systems, which feature an advanced metering infrastructure (AMI) that automatically collects meter data from widely distributed sensors [95].

The proposed algorithm for real-time energy data compression combines several existing compression algorithms and operates from 2 to 10% more efficiently than previously published algorithms. This efficiency is crucial in real-time systems, where data must be processed and transmitted quickly to avoid delays and potential system failures.

In addition to data compression, the resilience of errors in power grid systems also involves the use of machine learning and artificial intelligence techniques for fault detection and diagnosis. These techniques can help identify and diagnose faults in the power grid in real time, allowing quick remediation and minimizing the impact of these faults on power grid operation [96].

For example, hybrid machine learning models can be used to improve the resilience of the power grid through real-time fault detection and remediation [97]. These models combine different machine learning techniques to improve the accuracy and efficiency of fault detection and diagnosis.

In addition, the resilience of a power grid can be measured using a Dynamic Inoperability Input-output Model (DIIM). DIIM reflects the ability of the power grid to deal with disturbances and changes in the equilibrium state through the restoration ability. This model can be used to measure the resilience of the power grid and to verify the effectiveness of the measures taken to improve the resilience of the power grid [98].

Forward error correction (FEC) is a method used in data communication to detect and correct errors that occur during data transmission. FEC works by adding redundant data to the original data before it is transmitted [99]. This redundant data are used to detect and correct errors that occur during transmission. If an error is detected, the redundant data are used to correct the error. This allows the original data to be reconstructed accurately, even if some of it was lost or corrupted during transmission [100,101].

	*	1 0				
Algorithm	Category	Real-Time Application Benefits	Real-Time Application Drawbacks	Non-Time-Critical Application Benefits	Non-Time-Critical Application Drawbacks	References
RLE	Lossless	Low computational complexity	Less efficient for block algorithms	Well performing on large datasets, good compression of datasets with multiple subsequent occurrences of the same symbol	Low compression ratio	[102]
Sliding-window RLE	Lossless	Block operation, low latency for streaming	Lower compression ratio	More predictable output data format, block division may reduce compression ratio	Block division may reduce compression ratio	[103,104]
Huffman Coding	Lossless	Variable length, efficient for compression of smooth time-series	Changes to the dictionary might deteriorate performance	Well performing on large datasets, efficient compression for repetitive patterns	Efficiency relies on the fixed dictionary, complex computation	[40]
LZW	Lossless	Efficient for block operation, availability of hardware acceleration	Requires proper definition of dictionary	Adaptation to the input data, efficient for frequently occurring patterns	The size of the dictionary can grow significantly for large datasets, adds computational complexity	[39]
DCT	Lossy or Lossless	Ability to lose frequency components irrelevant for the application, preprocessing of data	Improper parameters selection may result in the loss of relevant data	Concentration of most of the signal energy in few crucial coefficients	Blocking artifacts as a result of data quantization	[105]
Wavelet Transform	Lossy or Lossless	Preprocessing of transient data, lowering computational effort for the controller	Inefficient for smooth signals, complex computation, possible block artifacts	Efficient for compressing data with diverse patterns, multiresolution analysis, possibility of 2D representation of a large dataset useful for trends observation	Complex computation for large datasets, performance of the algorithm is very sensitive to the choice of parameters	[106,107]

Table 2. Comparison of compression algorithms for time-series databases.

Algorithm	Category	Real-Time Application Benefits	Real-Time Application Drawbacks	Non-Time-Critical Application Benefits	Non-Time-Critical Application Drawbacks	References
Differential Encoding	Lossless, possible lossy	Low computational complexity, optimized for streamed data	Risk of error accumulation, sensitive to data variation	Low complexity and simple implementation, adaptable to trends	Risk of error accumulation, sensitive to data variation	[60,62]
BWT	Lossless	Preprocessing data blocks, enhancement of other algorithms, simple decompression	Added complexity, reduced benefit with distorted data	Efficient for compressing data with local patterns and repetitive sequences	Performance dependent on input data	[77]
MTF	Lossless	Preprocessing data blocks, enhancement of other algorithms	Variable compression ratio depending on the input	Simple implementation	Compression ratios may vary depending on the input data	[108]

Table	2.	Cont.	
-------	----	-------	--

4. Compression and Security

In the past, many data transmission channels in power grids were isolated and thus physically inaccessible to the attacker. In such systems, the security of the data was not considered. At present, many measurement systems work in the IoT network, utilizing mediums and infrastructure available to other entities. This raises concerns about data security. Compression algorithms are not neutral for security. The application of data compression influences the security properties of the systems in various ways.

4.1. Data Transmission Efficiency

Data compression is often used to reduce the size of files or data for more efficient transmission over networks. Smaller data sizes mean faster transmission times and reduced bandwidth usage. In a cybersecurity context, this efficiency is crucial to maintaining a secure and responsive network. Faster transmission can contribute to faster response times in the detection and prevention of security threats.

4.2. Network Security

Compressed data can impact how security devices, such as firewalls and intrusion detection systems, analyze network traffic. Some security tools may struggle to inspect compressed data effectively, potentially allowing malicious content to pass undetected. Attackers may use compression to obfuscate their payloads and make it harder for security systems to identify and block malicious activities.

4.3. Data Integrity

Compression algorithms, if not implemented or configured properly, can introduce vulnerabilities that attackers might exploit. Poorly designed compression algorithms may lead to data corruption or even facilitate certain types of attacks, such as compression-based attacks like the CRIME (Compression Ratio Info-leak Made Easy) attack, which targets the compression used in SSL/TLS protocols.

4.4. Storage Security

In storage environments, compressed data are commonly used to optimize storage space. However, security concerns arise if the compression algorithm is not secure. If an attacker can manipulate compressed data in a way that exploits vulnerabilities in the decompression process, it could lead to security breaches. Properly securing compressed data during storage is essential to maintain the confidentiality and integrity of sensitive information.

4.5. Secure File Transfer

In secure file transfer protocols, compression is often integrated to optimize data transfer times. However, the security of the compression algorithm and its implementation becomes paramount in ensuring the confidentiality and integrity of the transferred data. Encryption techniques are introduced to change the data before transfer in order to preserve the privacy of parties participating in communication [109] or protect the confidentiality of the data from adversaries. In many cases, the transmitted measurement data has the form of a 2D graphic, which is compressed using image codecs [110]. In such cases, it is crucial to compress data with preservation of the quality suitable for further data processing and maintain security properties of encryption [111].

4.6. Attack Scenarios Exploiting Compression Methods

Data compression bring benefit in the form of more efficient communication; however, changing the properties of data may expose the system to exploitation.

• Side-channel security of smart meter data compression techniques. In this scenario, attackers exploit side-channel information to compromise the security of smart me-

ters leveraging data compression techniques. By analyzing patterns such as power consumption, bus load, and temperature of systems processing compressed data, adversaries can gain insight into power consumption patterns, potentially revealing sensitive information about occupants' behavior or activities [112,113]. Unauthorized access to detailed information on power usage could lead to privacy breaches, enabling malicious entities to deduce occupancy patterns and habits or even identify periods of low security on a premise.

- Joint adversarial example and false data injection attacks for state estimation in power systems. Adversaries generate adversarial examples in compressed data sent for state estimation in power systems [114,115]. Additionally, false data injection attacks can involve injecting compressed or manipulated data into the system [116,117]. These attacks can compromise the accuracy of state estimation algorithms, leading to incorrect assessments of the state of the power system [118]. Misleading state estimates can result in improper control actions [119], potentially causing disruptions, overloads, or cascading failures in the power grid. This could lead to severe operational challenges, affecting the reliability and stability of the entire power system [120].
- Adversarial attacks and defenses for deep learning-based unmanned aerial vehicles (UAVs). Smart grids increasingly use UAVs for monitoring and maintenance. Adversarial attacks on the data compression techniques used in UAV communications can involve manipulating compressed data to deceive or disrupt the operation of these vehicles [121]. This might include altering sensor data or compromising communication channels [122]. Adversarial attacks on UAVs can lead to incorrect or delayed information, affecting decision-making processes in power grid management. This could potentially disrupt critical tasks such as infrastructure inspection, maintenance, or emergency response.
- Data integrity attacks on smart grid communication channels. Attackers exploit vulnerabilities in compression algorithms to manipulate or corrupt compressed data during transmission [123]. This can involve injecting false information, altering control signals, or disrupting communication channels between various components of the smart grid. Compromised data integrity can lead to incorrect decision-making in power grid control systems.
- Eavesdropping on compressed communication channels. Adversaries may eavesdrop on compressed communication channels within the smart grid. By intercepting and analyzing compressed data, attackers can gain insight into sensitive information, including control commands, grid configurations, or operational strategies [124]. Unauthorized access to critical information can allow attackers to plan more sophisticated and targeted attacks on the power grid, potentially leading to service disruptions, financial losses, or even physical damage to the infrastructure.

4.7. Influence on Entropy

Both data compression and cryptography focus on the modification of the message entropy; however, their point is the opposite. The goal of cryptography is to increase the entropy of the data, while the goal of compression is to reduce the entropy of the data. Through encryption, cryptography aims to transform data into a more unpredictable and seemingly random form, making it difficult for unauthorized parties to decipher without the appropriate key. Cryptography and compression serve opposite purposes in terms of data entropy. Cryptography seeks to make data more unpredictable and secure, while compression seeks to make data more efficient and space-saving by reducing redundancy and, consequently, lowering entropy. It is important to take security into account during the development of the data compression strategy.

5. Future Work

A future trend in the development of data compression techniques in smart grids could be the integration of machine learning and artificial intelligence (AI) to improve the efficiency and adaptability of these techniques.

Machine learning algorithms can be used to analyze patterns in data, which can help in the development of more efficient compression algorithms. Machine learning algorithms excel at uncovering intricate patterns and relationships within complex datasets. In the realm of data compression for smart grids, these algorithms can be trained to recognize the significance of different data segments. For example, in power systems, machine learning can identify critical aspects of electrical signals that must be preserved for accurate analysis. Subsequently, this insight aids in the development of more sophisticated compression algorithms that selectively retain essential information while efficiently compressing less critical data. This approach helps mitigate computational complexity and storage requirements [125].

Artificial intelligence can also be used to optimize the parameters of compression algorithms in real time, based on the characteristics of the data and the requirements of the system [95,126]. Artificial intelligence, particularly through the use of advanced algorithms, can optimize compression processes dynamically in real time. This adaptability is crucial in smart grid scenarios where the nature of the data and system requirements can vary. AI algorithms can continuously assess factors such as the urgency of data transmission, the available bandwidth in communication channels, and the specific needs of the system. By adjusting compression parameters on the fly, AI ensures that the compression process is aligned with the dynamic demands of the smart grid environment. This adaptability improves the overall efficiency of the data compression system.

Another potential trend is the use of quantum computing for data compression. Quantum computing has the potential to significantly speed up the compression and decompression processes, which could be particularly beneficial for handling large volumes of data in smart grids. However, this would require significant advancements in quantum computing technology [127]. Quantum computing introduces a paradigm shift in data compression by leveraging the principles of quantum mechanics. The inherent parallelism and superposition capabilities of quantum bits (qubits) can significantly speed up compression and decompression processes. In the context of smart grids, this translates to faster and more efficient handling of large volumes of data. Quantum computing has the potential to revolutionize the processing of complex datasets, which makes it particularly advantageous for the high-throughput requirements of smart grids. However, it is essential to note that the practical implementation of quantum computing in this domain necessitates substantial advancements in quantum technology, including error correction and scalability.

Finally, the development of more advanced data compression techniques that can handle different types of data and different levels of compression could also be a future trend. For example, techniques that can handle lossless and lossy compression or that can adapt the level of compression based on the characteristics of the data [128].

6. Summary

The primary objective of this paper was to draw attention to a major problem for automation, measurement, and control of power systems in the imminent future. The focal point of our investigation was the advancements made in the realm of compression techniques applied to electric signals. We delved into the existing developments in this area and shed light on the critical need for robust compression methodologies tailored for smart grid applications.

The most popular algorithms were described with application details. The paper focused on two different ways of handling data in power grids—large datasets, used for non-time-critical analysis, like power quality reports or monthly billing of the electricity consumers, and real-time data transfer, for time-critical purposes like islanding protection or circuit breakers. The advantages and drawbacks of selected data compression algorithms were presented in the application for both of these purposes.

In addition to emphasizing the current state of electric signal compression techniques, the paper underscored the necessity of introducing powerful compression methods to address the specific challenges posed by smart grid applications. As our world becomes increasingly interconnected, the demand for efficient and secure data transmission in smart grids grows substantially. Here, the synergy between cybersecurity and data compression becomes particularly noteworthy.

The relationship between cybersecurity and data compression in the context of smart grids lies at the intersection of efficient data handling and the securing of critical information. Although compression contributes to optimizing data transmission and storage, it introduces considerations for cybersecurity. Ensuring the confidentiality, integrity, and availability of compressed data becomes paramount, given the potential vulnerabilities that can be exploited by malicious actors. Therefore, future research efforts in this domain should focus not only on improving compression techniques but also on fortifying the cybersecurity measures associated with compressed data.

Addressing these research challenges is crucial to pave the way toward the development of novel and cost-effective devices. Reducing the amount of data transferred will facilitate the development of smart sensing, monitoring, metering, diagnosis, and protection systems in the next generation of electric power systems: smart grids. By addressing these challenges at the intersection of compression technologies and cybersecurity, we can propel the advancement of resilient and secure solutions for the evolving landscape of power systems.

The article aims to highlight the pressing challenges in the automation, measurement, and control of power systems, focusing on the need for robust compression techniques tailored for smart grid applications. The paper explores existing advancements in compression methods for electric signals, addressing two main aspects of power grid data handling: large datasets for non-time-critical analysis and real-time data transfer for time-sensitive operations.

The popular compression algorithms, including the discrete cosine transform, wavelet transform, differential encoding, Burrows–Wheeler transform, and move-to-front encoding, are described with their application details. The advantages and drawbacks of these algorithms are discussed in the context of handling both large datasets and real-time applications in power grids.

The article emphasizes the importance of introducing powerful compression methods to meet the specific challenges posed by smart grid applications. As the world becomes more interconnected, the demand for efficient and secure data transmission in smart grids increases substantially. The article highlights the intersection of cybersecurity and data compression, underlining the need to ensure the confidentiality, integrity, and availability of compressed data. Future research efforts are suggested to focus on fortifying cybersecurity measures associated with compressed data, in addition to improving compression techniques.

Addressing these research challenges is crucial for the development of novel and cost-effective devices, facilitating advancements in smart sensing, monitoring, metering, diagnosis, and protection systems in the next generation of electric power systems: smart grids. The article envisions the advancement of resilient and secure solutions to meet the evolving landscape of power systems.

Author Contributions: Conceptualization, K.P. and A.B.; methodology, K.P.; software, K.P.; formal analysis, K.P.; investigation, K.P.; resources, K.P.; data curation, K.P.; writing—original draft preparation, K.P.; writing—review and editing, K.P., S.B. and A.B.; visualization, K.P.; supervision, A.B. and S.B.; project administration, A.B.; funding acquisition, A.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: All source code used to create content for hereby article is available in the public repository: https://github.com/KamilProkop/Data_compression_smart_grid (accessed on 10 December 2023).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

RLE	Run-Length Encoding
SW-RLE	Sliding-Window Run-Length Encoding
LZW	Lempel–Ziv–Welch
DCT	Discrete cosine transform
DWT	Discrete Wavelet Transform
fCWT	Fast Continuous Wavelet Transform
BWT(-C)	Burrows–Wheeler Transform (Compression)
MTF(T)	Move-to-front (Transform)
DIIM	Dynamic Inoperability Input-Output Model
FEC	Forward error correction
AMI	Advanced Metering Infrastructure
FPGA	Field Programmable Gate Array
CDCI	Compression Distortion Composite Index
IoT	Internet of Things
JPEG	Joint Photographic Experts Group
MP3	MPEG Audio Layer III
GIF	Graphics Interchange Format
PNG	Portable Network Graphics
AAC	Advanced Audio Coding
Gzip	GNU Zip
CRIME	Compression Ratio Info-leak Made Easy
SSL	Secure Sockets Layer
TLS	Transport Layer Security
AI	Artificial Intelligence

References

- 1. Kamwa, I. Dynamic Wide Area Situational Awareness: Propelling Future Decentralized, Decarbonized, Digitized, and Democratized Electricity Grids. *IEEE Power Energy Mag.* 2023, 21, 44–58. [CrossRef]
- 2. Dileep, G. A survey on smart grid technologies and applications. *Renew. Energy* **2020**, *146*, 2589–2625. [CrossRef]
- IEC/IEEE 60255-118-1:2018; IEEE / IEC International Standard—Measuring Relays and Protection Equipment—Part 118-1: Synchrophasor for Power Systems—Measurements. IEEE: Piscataway, NJ, USA, 2018; pp. 1–78. [CrossRef]
- IEC/IEEE 61850-9-3 Edition 1.0 2016-05; IEC/IEEE International Standard—Communication Networks and Systems for Power Utility Automation—Part 9-3: Precision Time Protocol Profile for Power Utility Automation. IEEE: Piscataway, NJ, USA, 2016; pp. 1–18. [CrossRef]
- Hasan, M.K.; Ahmed, M.M.; Hashim, A.H.A.; Razzaque, A.; Islam, S.; Pandey, B. A Novel Artificial Intelligence Based Timing Synchronization Scheme for Smart Grid Applications. *Wirel. Pers. Commun.* 2020, 114, 1067–1084. [CrossRef]
- 6. Cai, Z.; Yu, M.; Steurer, M.; Li, H.; Dong, Y. A network model for the real-time communications of a smart grid prototype. *J. Netw. Comput. Appl.* **2016**, *59*, 264–273. [CrossRef]
- Ponnusamy, V.K.; Kasinathan, P.; Madurai Elavarasan, R.; Ramanathan, V.; Anandan, R.K.; Subramaniam, U.; Ghosh, A.; Hossain, E. A Comprehensive Review on Sustainable Aspects of Big Data Analytics for the Smart Grid. Sustainability 2021, 13, 13322. [CrossRef]
- 8. Zhang, Y.; Huang, T.; Bompard, E.F. Big data analytics in smart grids: A review. *Energy Inform.* 2018, 1, 8. [CrossRef]
- Song, D.; Wen, B.; Yang, X.; Gu, Y.; Wei, S.; Ma, S. A New Method for Processing and Application of Wide Area Measurement Big Data in a Power System. In Proceedings of the 2015 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, Xi'an, China, 17–19 September 2015; pp. 137–140. [CrossRef]
- Huang, W.; Wang, W.; Xu, H. A Lossless Data Compression Algorithm for Real-time Database. In Proceedings of the 2006 6th World Congress on Intelligent Control and Automation, Dalian, China, 21–23 June 2006; Volume 2, pp. 6645–6648. [CrossRef]
- Liang, X.; Di, S.; Tao, D.; Li, S.; Li, S.; Guo, H.; Chen, Z.; Cappello, F. Error-Controlled Lossy Compression Optimized for High Compression Ratios of Scientific Datasets. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; pp. 438–447. [CrossRef]

- 12. Jeong, H.; Seo, G.; Hwang, E. Lossless Data Compression with Bit-back Coding on Massive Smart Meter Data. In Proceedings of the 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, 17–20 December 2022; pp. 6667–6669. [CrossRef]
- Wang, Y.; Chen, Q.; Kang, C.; Xia, Q.; Luo, M. Sparse and Redundant Representation-Based Smart Meter Data Compression and Pattern Extraction. *IEEE Trans. Power Syst.* 2017, 32, 2142–2151. [CrossRef]
- Ichihara, H.; Kinoshita, K.; Pomeranz, I.; Reddy, S. Test transformation to improve compaction by statistical encoding. In Proceedings of the VLSI Design 2000, Wireless and Digital Imaging in the Millennium, 13th International Conference on VLSI Design, Calcutta, India, 3–7 January 2000; pp. 294–299. [CrossRef]
- Kanda, S.; Morita, K.; Fuketa, M. Practical String Dictionary Compression Using String Dictionary Encoding. In Proceedings of the 2017 International Conference on Big Data Innovations and Applications (Innovate-Data), Prague, Czech Republic, 21–23 August 2017; pp. 1–8. [CrossRef]
- Chandak, S.; Tatwawadi, K.; Wen, C.; Wang, L.; Aparicio Ojea, J.; Weissman, T. LFZip: Lossy Compression of Multivariate Floating-Point Time Series Data via Improved Prediction. In Proceedings of the 2020 Data Compression Conference (DCC), Snowbird, UT, USA, 24–27 March 2020; pp. 342–351. [CrossRef]
- Christopoulos, C.; Skodras, A.; Ebrahimi, T. The JPEG2000 still image coding system: An overview. *IEEE Trans. Consum. Electron.* 2000, 46, 1103–1127. [CrossRef]
- 18. Girod, B.; Aaron, A.; Rane, S.; Rebollo-Monedero, D. Distributed Video Coding. Proc. IEEE 2005, 93, 71–83. [CrossRef]
- 19. Wallace, G. The JPEG still picture compression standard. *IEEE Trans. Consum. Electron.* 1992, 38, xviii–xxxiv. [CrossRef]
- Luo, D.; Luo, W.; Yang, R.; Huang, J. Compression history identification for digital audio signal. In Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 25–30 March 2012; pp. 1733–1736. [CrossRef]
- Jose, K.M.; Morsi, W.G. Smart Grid Data Compression of Power Quality Events using Wavelet Transform. In Proceedings of the 2022 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), Halifax, NS, Canada, 18–20 September 2022; pp. 159–164. [CrossRef]
- Hsu, C.C.; Fang, Y.T.; Yu, F. Content-Sensitive Data Compression for IoT Streaming Services. In Proceedings of the 2017 IEEE International Congress on Internet of Things (ICIOT), Honolulu, HI, USA, 25–30 June 2017; pp. 147–150. [CrossRef]
- Song, J.; Lee, Y.; Hwang, E. Adaptive Compressive Sensing based Data Compression on Energy Consumption in Smart Grid. In Proceedings of the 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 16–18 October 2019; pp. 568–570. [CrossRef]
- 24. Wyner, A.; Ziv, J. The rate-distortion function for source coding with side information at the decoder. *IEEE Trans. Inf. Theory* **1976**, 22, 1–10. [CrossRef]
- Yan, J.; Wang, L. The Novel Improving Algorithms on DRA Audio Entropy Coding. In Proceedings of the 2020 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Macau, China, 21–24 August 2020; pp. 1–4. [CrossRef]
- Marpe, D.; Schwarz, H.; Wiegand, T. Entropy coding in video compression using probability interval partitioning. In Proceedings of the 28th Picture Coding Symposium, Nagoya, Japan, 8–10 December 2010; pp. 66–69. [CrossRef]
- Krithivasan, D.; Pradhan, S.S. Distributed Source Coding Using Abelian Group Codes: A New Achievable Rate-Distortion Region. *IEEE Trans. Inf. Theory* 2011, 57, 1495–1519. [CrossRef]
- Ergude, B.; Weisheng, L.; Dongrui, F.; Xiaoyu, M. A Study and Implementation of the Huffman Algorithm Based on Condensed Huffman Table. In Proceedings of the 2008 International Conference on Computer Science and Software Engineering, Wuhan, China, 12–14 December 2008; Volume 6, pp. 42–45. [CrossRef]
- 29. Huang, J.Y.; Liang, Y.C.; Huang, Y.M. Secure integer arithmetic coding with adjustable interval size. In Proceedings of the 2013 19th Asia-Pacific Conference on Communications (APCC), Denpasar, Indonesia, 29–31 August 2013; pp. 683–687. [CrossRef]
- Zribi, A.; Zaibi, S.; Pyndiah, R.; Bouallègue, A. Low-Complexity Joint Source/Channel Turbo Decoding of Arithmetic Codes with Image Transmission Application. In Proceedings of the 2009 Data Compression Conference, Snowbird, UT, USA, 16–18 March 2009; p. 472. [CrossRef]
- 31. Berger, T.; Gibson, J. Lossy source coding. IEEE Trans. Inf. Theory 1998, 44, 2693–2723. [CrossRef]
- 32. Fresia, M.; Vandendorpe, L.; Poor, H.V. Distributed Source Coding Using Raptor Codes for Hidden Markov Sources. In Proceedings of the Data Compression Conference (DCC 2008), Snowbird, UT, USA, 25–27 March 2008; p. 517. [CrossRef]
- 33. Golomb, S. Run-length encodings (Corresp.). IEEE Trans. Inf. Theory 1966, 12, 399-401. [CrossRef]
- 34. Tamakoshi, Y.; Tomohiro, I.; Inenaga, S.; Bannai, H.; Takeda, M. From Run Length Encoding to LZ78 and Back Again. In Proceedings of the 2013 Data Compression Conference, Snowbird, UT, USA, 20–22 March 2013; pp. 143–152. [CrossRef]
- 35. Malvar, H. Adaptive run-length/Golomb-Rice encoding of quantized generalized Gaussian sources with unknown statistics. In Proceedings of the Data Compression Conference (DCC'06), Snowbird, UT, USA, 28–30 March 2006; pp. 23–32. [CrossRef]
- 36. Graf, U. Sorted sliding window compression. In Proceedings of the Proceedings DCC'99 Data Compression Conference (Cat. No. PR00096), Snowbird, UT, USA, 29–31 March 1999; p. 527. [CrossRef]
- Cha, H.; Shon, T.; Kim, K.; Hong, M. Improving packet header compression with adaptive sliding window size. In Proceedings of the 2015 International Conference on Information Networking (ICOIN), Siem Reap, Cambodia, 12–14 January 2015; pp. 541–543. [CrossRef]
- 38. Huffman, D.A. A Method for the Construction of Minimum-Redundancy Codes. Proc. IRE 1952, 40, 1098–1101. [CrossRef]

- Zeinali, M.; Thompson, J.S. Impact of compression and aggregation in wireless networks on smart meter data. In Proceedings of the 2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Edinburgh, UK, 3–6 July 2016; pp. 1–5. [CrossRef]
- 40. Yan, L.; Han, J.; Xu, R.; Li, Z. Model-Free Lossless Data Compression for Real-Time Low-Latency Transmission in Smart Grids. *IEEE Trans. Smart Grid* 2021, 12, 2601–2610. [CrossRef]
- 41. Ferreira, A.; Oliveira, A.; Figueiredo, M. On the Use of Suffix Arrays for Memory-Efficient Lempel-Ziv Data Compression. In Proceedings of the 2009 Data Compression Conference, Snowbird, UT, USA, 16–18 March 2009; p. 444. [CrossRef]
- 42. Köppl, D.; Sadakane, K. Lempel-Ziv Computation in Compressed Space (LZ-CICS). In Proceedings of the 2016 Data Compression Conference (DCC), Snowbird, UT, USA, 30 March–1 April 2016; pp. 3–12. [CrossRef]
- 43. Wyner, A.; Ziv, J. The sliding-window Lempel-Ziv algorithm is asymptotically optimal. Proc. IEEE 1994, 82, 872–877. [CrossRef]
- 44. Ahmed, N.; Natarajan, T.; Rao, K. Discrete Cosine Transform. IEEE Trans. Comput. 1974, C-23, 90–93. [CrossRef]
- 45. Dong, Y.; Pan, W.D. A Survey on Compression Domain Image and Video Data Processing and Analysis Techniques. *Information* **2023**, *14*, 184. [CrossRef]
- Chen, W.H.; Smith, C.; Fralick, S. A Fast Computational Algorithm for the Discrete Cosine Transform. *IEEE Trans. Commun.* 1977, 25, 1004–1009. [CrossRef]
- 47. Shabaan Samra, A.; El Taweel Gad Allah, S.; Mahmoud Ibrahim, R. Face recognition using wavelet transform, fast Fourier transform and discrete cosine transform. In Proceedings of the 2003 46th Midwest Symposium on Circuits and Systems, Cairo, Egypt, 27–30 December 2003; Volume 1, pp. 272–275. [CrossRef]
- Shukla, S.; Ahirwar, M.; Gupta, R.; Jain, S.; Rajput, D.S. Audio Compression Algorithm using Discrete Cosine Transform (DCT) and Lempel-Ziv-Welch (LZW) Encoding Method. In Proceedings of the 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 14–16 February 2019; pp. 476–480. [CrossRef]
- 49. *IEEE Std* 1180-1990; IEEE Standard Specifications for the Implementations of 8 × 8 Inverse Discrete Cosine Transform. IEEE: Piscataway, NJ, USA, 1991; pp. 1–12. [CrossRef]
- Zhou, J.; Chen, P. Generalized Discrete Cosine Transform. In Proceedings of the 2009 Pacific-Asia Conference on Circuits, Communications and Systems, Chengdu, China, 16–17 May 2009; pp. 449–452. [CrossRef]
- 51. Lai, C.S. Compression of power system signals with wavelets. In Proceedings of the 2014 International Conference on Wavelet Analysis and Pattern Recognition, Lanzhou, China, 13–16 July 2014; pp. 109–115. [CrossRef]
- 52. Mehra, R.; Patel, V.; Kazi, F.; Singh, N.; Wagh, S. Modes preserving wavelet based multi-scale PCA algorithm for compression of smart grid data. In Proceedings of the 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Mysore, India, 22–25 August 2013; pp. 817–821. [CrossRef]
- 53. Arts, L.P.A.; van den Broek, E.L. The fast continuous wavelet transformation (fCWT) for real-time, high-quality, noise-resistant time–frequency analysis. *Nat. Comput. Sci.* 2022, 2, 47–58. [CrossRef]
- Cheng, L.; Ji, X.; Zhang, F.; Huang, H.; Gao, S. Wavelet-based data compression for wide-area measurement data of oscillations. J. Mod. Power Syst. Clean Energy 2018, 6, 1128–1140. [CrossRef]
- Li, Q.; Zhang, M.; He, S.; Li, S. Instructions Data Compression for Smart Grid Monitoring using Wavelet Domain Singular Value Decomposition. In Proceedings of the 2018 Chinese Control And Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; pp. 5171–5175. [CrossRef]
- 56. Ramchandran, K.; Vetterli, M. Best wavelet packet bases in a rate-distortion sense. *IEEE Trans. Image Process.* **1993**, *2*, 160–175. [CrossRef]
- 57. Ding, H.; Liu, H.; Yang, F. A Real-Time Compression Method of Power System Waveform Data Based on 2-D Lifting Wavelet Transform and Deflate Algorithm. In Proceedings of the 2021 3rd Asia Energy and Electrical Engineering Symposium (AEEES), Chengdu, China, 26–29 March 2021; pp. 626–630. [CrossRef]
- Bhuiyan, B.A.; Absar, M.W.; Roy, A. Performance comparison of various wavelets in compression of PMU generated data in smart grid. In Proceedings of the 2017 3rd International Conference on Electrical Information and Communication Technology (EICT), Khulna, Bangladesh, 7–9 December 2017; pp. 1–6. [CrossRef]
- Su, S.-P.; Luo, X.; Qin, Z.-Q. Study on Transient Power Quality Detection of Grid-Connected Wind Power Generation System Based on Wavelet Transform. In Proceedings of the 2009 International Conference on Energy and Environment Technology, Guilin, China, 16–18 October 2009; Volume 1, pp. 861–864. [CrossRef]
- Kraus, J.; Štěpán, P.; Kukačka, L. Optimal data compression techniques for Smart Grid and power quality trend data. In Proceedings of the 2012 IEEE 15th International Conference on Harmonics and Quality of Power, Hong Kong, China, 17–20 June 2012; pp. 707–712. [CrossRef]
- Sarkar, S.J.; Kundu, P.K.; Sarkar, G. DBEA: A novel approach of repetitive data array compression for power system application. In Proceedings of the 2017 2nd International Conference for Convergence in Technology (I2CT), Mumbai, India, 7–9 April 2017; pp. 824–827. [CrossRef]
- Aleshinloye, A.; Bais, A. Modified Differential Golomb Arithmetic Lossless Compression Algorithm for Smart Grid Applications. In Proceedings of the 2018 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), Sarajevo, Bosnia and Herzegovina, 21–25 October 2018; pp. 1–6. [CrossRef]
- 63. Fiergolla, S.; Wolf, P. Improving Run Length Encoding by Preprocessing. In Proceedings of the 2021 Data Compression Conference (DCC), Snowbird, UT, USA, 23–26 March 2021; p. 341. [CrossRef]

- 64. Arnavut, Z. ECG Signal Compression Based on Burrows–Wheeler Transformation and Inversion Ranks of Linear Prediction. *IEEE Trans. Biomed. Eng.* **2007**, *54*, 410–418. [CrossRef]
- 65. Begum, M.B.; Deepa, N.; Uddin, M.; Kaluri, R.; Abdelhaq, M.; Alsaqour, R. An efficient and secure compression technique for data protection using Burrows–Wheeler transform algorithm. *Heliyon* **2023**, *9*, e17602. [CrossRef] [PubMed]
- 66. Effros, M.; Visweswariah, K.; Kulkarni, S.; Verdu, S. Universal lossless source coding with the Burrows Wheeler transform. *IEEE Trans. Inf. Theory* **2002**, *48*, 1061–1081. [CrossRef]
- 67. Fenwick, P.; Titchener, M.; Lorenz, M. Burrows Wheeler—Alternatives to move to front. In Proceedings of the Data Compression Conference, Snowbird, UT, USA, 25–27 March 2003; p. 428. [CrossRef]
- Elsayed, H.A. Burrows–Wheeler Transform and combination of Move-to-Front coding and Run Length Encoding for lossless audio coding. In Proceedings of the 2014 9th International Conference on Computer Engineering & Systems (ICCES), Cairo, Egypt, 22–23 December 2014; pp. 354–359. [CrossRef]
- 69. Cenzato, D.; Lipták, Z. On different variants of the Burrows–Wheeler-Transform of string collections. In Proceedings of the 2022 Data Compression Conference (DCC), Snowbird, UT, USA, 22–25 March 2022; p. 448. [CrossRef]
- Li, B.; Ni, G.; Luo, J.; Zhang, X. BWT-based Data Preprocessing for LZW. In Proceedings of the 2011 International Conference on Multimedia and Signal Processing, Guilin, China, 14–15 May 2011; Volume 1, pp. 37–40. [CrossRef]
- 71. Ryabko, B.Y. Data Compression by Means of a "Book Stack". Probl. Inform. Transm. 1980, 16, 265–269.
- 72. Bentley, J.L.; Sleator, D.D.; Tarjan, R.E.; Wei, V.K. A Locally Adaptive Data Compression Scheme. *Commun. ACM* **1986**, *29*, 320–330. [CrossRef]
- 73. Patel, R.A.; Zhang, Y.; Mak, J.; Davidson, A.; Owens, J.D. Parallel lossless data compression on the GPU. In Proceedings of the 2012 Innovative Parallel Computing (InPar), San Jose, CA, USA, 13–14 May 2012; pp. 1–9. [CrossRef]
- 74. Arnavut, Z. Move-to-front and inversion coding. In Proceedings of the DCC 2000, Data Compression Conference, Snowbird, UT, USA, 28–30 March 2000; pp. 193–202. [CrossRef]
- 75. Nanda, S.S.; Das, K.; Padhi, J.; Hota, S. Advanced move-to-front for list access problem. In Proceedings of the 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Nagercoil, India, 18–19 March 2016; pp. 1–4. [CrossRef]
- Elsayed, H.A.; Alghoniemy, M. Lossless audio coding using Burrows–Wheeler Transform and move-to-front coding. In Proceedings of the 2007 International Conference on Computer Engineering & Systems, Cairo, Egypt, 27–29 November 2007; pp. 209–212. [CrossRef]
- Abuadbba, A.; Khalil, I.; Yu, X. Gaussian Approximation-Based Lossless Compression of Smart Meter Readings. *IEEE Trans. Smart Grid* 2018, 9, 5047–5056. [CrossRef]
- Liu, W.; Hei, X.; Fukuma, S.; Mori, S.I. A Simple and Real-Time Parallel Compression of Time Series Scientific Simulation Data for Interactive and Cooperative Supercomputing. In Proceedings of the 2014 Tenth International Conference on Computational Intelligence and Security, Kunming, China, 15–16 November 2014; pp. 578–582. [CrossRef]
- Shu, S.h.; Shu, Y. A two-stage data compression method for real-time database. In Proceedings of the 2012 3rd International Conference on System Science, Engineering Design and Manufacturing Informatization, Chengdu, China, 20–21 October 2012; Volume 2, pp. 204–207. [CrossRef]
- Huang, B.; Pan, Y.; Wang, Z. Application of Steady State Data Compressed Sensing Based on LSTM and RNN in Rural Power Grid. In Proceedings of the 2023 3rd International Conference on Electrical Engineering and Mechatronics Technology (ICEEMT), Nanjing, China, 21–23 July 2023; pp. 798–801. [CrossRef]
- 81. Sinha, P.; Suri, N. On the use of formal techniques for analyzing dependable real-time protocols. In Proceedings of the 20th IEEE Real-Time Systems Symposium (Cat. No.99CB37054), Phoenix, AZ, USA, 1–3 December 1999; pp. 126–135. [CrossRef]
- 82. Ziv, J.; Lempel, A. A universal algorithm for sequential data compression. *IEEE Trans. Inf. Theory* **1977**, 23, 337–343. [CrossRef]
- Mahmoud, A.; Farid, S.; Maged, M.; Mohamed, O.; Karam, R.; Salah, K.; El-Kharashi, M.W. An Efficient Hardware Accelerator For Lossless Data Compression. In Proceedings of the 2022 5th International Conference on Communications, Signal Processing, and their Applications (ICCSPA), Cairo, Egypt, 27–29 December 2022; pp. 1–4. [CrossRef]
- Fowers, J.; Kim, J.Y.; Burger, D.; Hauck, S. A Scalable High-Bandwidth Architecture for Lossless Compression on FPGAs. In Proceedings of the 2015 IEEE 23rd Annual International Symposium on Field-Programmable Custom Computing Machines, Vancouver, BC, Canada, 2–6 May 2015; pp. 52–59. [CrossRef]
- Hauck, S.; Wilson, W. Runlength compression techniques for FPGA configurations. In Proceedings of the Seventh Annual IEEE Symposium on Field-Programmable Custom Computing Machines (Cat. No.PR00375), Napa Valley, CA, USA, 23 April 1999; pp. 286–287. [CrossRef]
- Kokovin, V.A.; Uvaysov, S.U.; Uvaysova, S.S. Real-time sorting and lossless compression of data on FPGA. In Proceedings of the 2018 Moscow Workshop on Electronic and Networking Technologies (MWENT), Moscow, Russia, 14–16 March 2018; pp. 1–5. [CrossRef]
- Chandraprabha, K.; Shiak, N.A.; Lakshmi, A.; Anantha Shayanam, G.R.; Subramanian, V.R.; Srividhya, S.; Soman, J.; Vanitha, M. An Efficient FPGA Implementation of DWT Based Image Data Compression System for Microsatellites. In Proceedings of the 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 25–26 March 2022; Volume 1, pp. 745–750. [CrossRef]

- El Ghany, M.A.A.; Salama, A.E.; Khalil, A.H. Design and Implementation of FPGA-based Systolic Array for LZ Data Compression. In Proceedings of the 2007 IEEE International Symposium on Circuits and Systems (ISCAS), New Orleans, LA, USA, 27–30 May 2007; pp. 3691–3695. [CrossRef]
- Tan, G.; Wang, Y. A Compression Error and Optimize Compression Algorithm for Vector Data. In Proceedings of the 2009 International Conference on Environmental Science and Information Application Technology, Wuhan, China, 4–5 July 2009; Volume 2, pp. 522–525. [CrossRef]
- 90. Donoho, D. Compressed sensing. IEEE Trans. Inf. Theory 2006, 52, 1289–1306. [CrossRef]
- Uribe-Pérez, N.; Angulo, I.; de la Vega, D.; Arrinda, A.; Arzuaga, T.; Marrón, L.; Martínez, S.; Sendín, A.; Urrutia, I. TCP/IP capabilities over NB-PLC for Smart Grid applications: Field validation. In Proceedings of the 2017 IEEE International Symposium on Power Line Communications and its Applications (ISPLC), Madrid, Spain, 3–5 April 2017; pp. 1–5. [CrossRef]
- Fufachev, A.S.; Sintsov, A.V.; Krasikov, M.I.; Luppov, A.V. Data compression techniques for improving of performance and reliability of a multifunctional radio receiving center. In Proceedings of the 2017 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), St. Petersburg, Russia, 16–19 May 2017; pp. 1–4. [CrossRef]
- Iyer, V.; Rammurthy, G.; Srinivas, M.B. Training Data Compression Algorithms and Reliability in Large Wireless Sensor Networks. In Proceedings of the 2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (sutc 2008), Taichung, Taiwan, 11–13 June 2008; pp. 480–485. [CrossRef]
- 94. Kwon, B.; Lee, S. Error detection algorithm for Lempel-Ziv-77 compressed data. J. Commun. Netw. 2019, 21, 100–112. [CrossRef]
- 95. Huang, J.F.; Zhang, G.H.; Hsieh, S.Y. Real-time energy data compression strategy for reducing data traffic based on smart grid AMI networks. *J. Supercomput.* 2021, 77, 10097–10116. [CrossRef]
- 96. Almasoudi, F.M. Enhancing Power Grid Resilience through Real-Time Fault Detection and Remediation Using Advanced Hybrid Machine Learning Models. *Sustainability* **2023**, *15*, 8348. [CrossRef]
- Qaisar, S.M.; Alyamani, N. Adaptive Rate Sampling and Machine Learning Based Power Quality Disturbances Interpretation. In Proceedings of the 2021 7th International Conference on Event-Based Control, Communication, and Signal Processing (EBCCSP), Krakow, Poland, 22–25 June 2021; pp. 1–6. [CrossRef]
- 98. Wang, Y.; Gao, S.; Wang, F. Measurement of Power Grid Resilience Based on a Dynamic Inoperability Input–Output Model. *Front. Phys.* **2022**, *10*, 895267. [CrossRef]
- 99. Demir, U.; Aktas, O. Raptor versus Reed Solomon forward error correction codes. In Proceedings of the 2006 International Symposium on Computer Networks, Istanbul, Turkey, 16–18 June 2006; pp. 264–269. [CrossRef]
- 100. Zhang, L.; Xiao, D.; Tang, Y.; Zhang, Y. Adaptive error control in Wireless Sensor Networks. In Proceedings of the IET International Conference on Wireless Sensor Network 2010 (IET-WSN 2010), Beijing, China, 15–17 November 2010; pp. 360–366. [CrossRef]
- Pfeifer, P.; Vierhaus, H.T. Forward error correction in wireless communication systems for industrial applications. In Proceedings of the 2017 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), Poznan, Poland, 20–22 September 2017; p. 14. [CrossRef]
- Ding, J.J. Binary Bit Stream Compression Based on Mode Encoding Algorithm. In Proceedings of the 2023 IEEE 3rd International Conference on Electronic Communications, Internet of Things and Big Data (ICEIB), Taichung, Taiwan, 14–16 April 2023; pp. 155–157. [CrossRef]
- Trein, J.; Schwarzbacher, A.T.; Hoppe, B.; Noffz, K.H. A hardware implementation of a run length encoding compression algorithm with parallel inputs. In Proceedings of the IET Irish Signals and Systems Conference (ISSC 2008), Galway, Ireland, 18–19 June 2008; pp. 337–342. [CrossRef]
- 104. Amin, A.; Qureshi, H.A.; Junaid, M.; Habib, M.Y.; Anjum, W. Modified run length encoding scheme with introduction of bit stuffing for efficient data compression. In Proceedings of the 2011 International Conference for Internet Technology and Secured Transactions, Abu Dhabi, United Arab Emirates, 11–14 December 2011; pp. 668–672.
- 105. Joshi, A.; Das, L.; Natarajan, B.; Srinivasan, B. Effect of Transformation in Compressed Sensing of Smart Grid Data. In Proceedings of the 2019 IEEE PES GTD Grand International Conference and Exposition Asia (GTD Asia), Bangkok, Thailand, 19–23 March 2019; pp. 177–182. [CrossRef]
- Ruiz, M.; Jaramillo, M.; Aguila, A.; Ortiz, L.; Varela, S. A Novel Data Compression Methodology Focused on Power Quality Signals Using Compressive Sampling Matching Pursuit. *Energies* 2022, 15, 9345. [CrossRef]
- Liu, G.; Jia, L.; Hu, T.; Deng, F.; Chen, Z.; Sun, T.; Feng, Y. Novel Data Compression Algorithm for Transmission Line Condition Monitoring. *Energies* 2021, 14, 8275. [CrossRef]
- Jumar, R.; Maaß, H.; Hagenmeyer, V. Comparison of lossless compression schemes for high rate electrical grid time series for smart grid monitoring and analysis. *Comput. Electr. Eng.* 2018, 71, 465–476. [CrossRef]
- Li, Z.; Xu, H.; Zhai, F.; Zhao, B.; Xu, M.; Guo, Z. A Privacy-Preserving, Two-Party, Secure Computation Mechanism for Consensus-Based Peer-to-Peer Energy Trading in the Smart Grid. Sensors 2022, 22, 9020. [CrossRef]
- 110. Al-Hazaimeh, O.; Al-Jamal, M.; Alomari, A.; Bawaneh, M.; Tahat, N. Image encryption using anti-synchronisation and Bogdanov transformation map. *Int. J. Comput. Sci. Math.* **2022**, *15*, 43. [CrossRef]
- 111. Kou, L.; Wu, J.; Zhang, F.; Ji, P.; Ke, W.; Wan, J.; Liu, H.; Li, Y.; Yuan, Q. Image encryption for offshore wind power based on 2D-LCLM and Zhou Yi eight trigrams. *Int. J. Bio-Inspired Comput.* **2023**, *22*, 53–64. [CrossRef]

- 112. Fehér, M.; Yazdani, N.; Aranha, D.F.; Lucani, D.E.; Hansen, M.T.; Vester, F.E. Side Channel Security of Smart Meter Data Compression Techniques. In Proceedings of the 2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Tempe, AZ, USA, 11–13 November 2020; pp. 1–6. [CrossRef]
- 113. Sangodoyin, S.; Werner, F.T.; Yilmaz, B.B.; Cheng, C.L.; Ugurlu, E.M.; Sehatbakhsh, N.; Prvulović, M.; Zajic, A. Side-Channel Propagation Measurements and Modeling for Hardware Security in IoT Devices. *IEEE Trans. Antennas Propag.* 2021, 69, 3470–3484. [CrossRef]
- 114. Hamamreh, J.M.; Furqan, H.M.; Arslan, H. Classifications and Applications of Physical Layer Security Techniques for Confidentiality: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 1773–1828. [CrossRef]
- 115. Yuan, Y.; Zhang, Q.; Dehghanpour, K.; Bu, F.; Wang, Z. Smart Meter Data Compression and Reconstruction Using Deep Convolutional Autoencoders. In Proceedings of the 2020 52nd North American Power Symposium (NAPS), Tempe, AZ, USA, 11–13 April 2021; pp. 1–5. [CrossRef]
- 116. Shiu, Y.S.; Chang, S.Y.; Wu, H.C.; Huang, S.C.H.; Chen, H.H. Physical layer security in wireless networks: A tutorial. *IEEE Wirel. Commun.* **2011**, *18*, 66–74. [CrossRef]
- 117. Zhao, Y.; Jia, X.; An, D.; Yang, Q. LSTM-Based False Data Injection Attack Detection in Smart Grids. In Proceedings of the 2020 35th Youth Academic Annual Conference of Chinese Association of Automation (YAC), Zhanjiang, China, 16–18 October 2020; pp. 638–644. [CrossRef]
- 118. Andic, C.; Ozturk, A.; Turkay, B. False Data Injection Attacks on CSA-Based State Estimation in Smart Grid. In Proceedings of the 2022 Global Energy Conference (GEC), Batman, Turkey, 26–29 October 2022; pp. 253–257. [CrossRef]
- 119. Tian, J.; Wang, B.; Li, J.; Wang, Z.; Ma, B.; Ozay, M. Exploring Targeted and Stealthy False Data Injection Attacks via Adversarial Machine Learning. *IEEE Internet Things J.* 2022, *9*, 14116–14125. [CrossRef]
- 120. Tian, J.; Wang, B.; Wang, Z.; Cao, K.; Li, J.; Ozay, M. Joint Adversarial Example and False Data Injection Attacks for State Estimation in Power Systems. *IEEE Trans. Cybern.* **2022**, *52*, 13699–13713. [CrossRef] [PubMed]
- 121. Zhang, G.; Wu, Q.; Cui, M.; Zhang, R. Securing UAV Communications via Joint Trajectory and Power Control. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 1376–1389. [CrossRef]
- 122. Tian, J.; Wang, B.; Guo, R.; Wang, Z.; Cao, K.; Wang, X. Adversarial Attacks and Defenses for Deep-Learning-Based Unmanned Aerial Vehicles. *IEEE Internet Things J.* 2022, *9*, 22399–22409. [CrossRef]
- Gui, Y.; Siddiqui, A.S.; Tamore, S.M.; Saqib, F. Security Vulnerabilities of Smart Meters in Smart Grid. In Proceedings of the IECON 2019—45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal, 14–17 October 2019; Volume 1, pp. 3018–3023. [CrossRef]
- 124. Zhong, X.; Fan, C.; Zhou, S. Eavesdropping area for evaluating the security of wireless communications. *China Commun.* **2022**, 19, 145–157. [CrossRef]
- Subbarao, D.; Kumar, A.S.; Bikshapthi, S.K.; Malathi, N.; Ashraf, M. Analysis of Data Compression Techniques in Smart Grids for Optimizing Mean-Square-Error. Appl. Nanosci. 2023, 13, 1591–1599. [CrossRef]
- Alsaigh, R.; Mehmood, R.; Katib, I. AI Explainability and Governance in Smart Energy Systems: A Review. Front. Energy Res. 2023, 11, 1071291. [CrossRef]
- 127. Eichinger, F.; Efros, P.; Karnouskos, S.; Böhm, K. A Time-series Compression Technique and Its Application to the Smart Grid. *VLDB J.* **2015**, *24*, 193–218. [CrossRef]
- Escobar, J.J.M.; Matamoros, O.M.; Padilla, R.T.; Reyes, I.L.; Espinosa, H.Q. A Comprehensive Review on Smart Grids: Challenges and Opportunities. Sensors 2021, 21, 6978. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.