Big Data Techniques for Wind Turbine Condition Monitoring

David Ferguson
University of Strathclyde
Glasgow, UK
david.ferguson@strath.ac.uk

Victoria M. Catterson
University of Strathclyde
Glasgow, UK
v.m.catterson@strath.ac.uk

Abstract

The desire to reduce the cost of energy from wind turbine generation has seen an increase in the research applied to the field of wind turbine condition monitoring. Wind turbine condition monitoring has the potential to reduce operation and maintenance costs through optimised maintenance scheduling and the avoidance of major breakdowns. To aid this research, increasing volumes of data are being captured and stored. These large volumes of data may be deemed 'Big Data', and require improved handling techniques in order to work with the data efficiently. This paper introduces a wind turbine condition monitoring system which has been installed in an operational Vestas V47 wind turbine for the purpose of developing algorithms to detect machine deterioration. The system's ability to capture large volumes of data (approx. 2TB per month) has led to the necessity of using enhanced data handling techniques. This paper will discuss these 'Big Data' techniques and suggest how they may ultimately be used for condition monitoring of multiple wind turbines or wind farms.

Keywords: Big data techniques, wind turbine condition monitoring.

1.0 Introduction

The continual development of sensor and storage technology has led to a dramatic increase in volumes of data being captured for condition monitoring and machine health assessment. Beyond wind energy, many sectors are dealing with the same issue, and these large, complex data sets have been termed 'Big Data'. Companies are increasingly looking towards Big Data tools and approaches for managing huge volumes of data [1]. Until recently, the majority of these companies have been in the marketing or financial sectors dealing with the behaviours of their customers [2]. Other companies involved in Big Data include delivery companies such as UPS who are tracking millions of packages worldwide [3].

As technology improves and it becomes easier to store large volumes of data, the definition of Big Data moves from the terabyte scale to the petabyte scale. Big Data can be described in terms of the 3Vs model [4]: velocity (the speed at which the data can be processed), volume (the volume of data that is being stored or analysed), and variety (the different types of data that are being stored or analysed). Big Data differs from standard data due to the complexity introduced by these three parameters.

One area where Big Data practices are receiving greater attention and which is not dissimilar to wind energy generation, is power system operation and the management of power networks [5, 6]. These papers highlight the potential benefits of using Big Data practices in order to fully utilise large volumes of data. Having the capabilities to work with large volumes of data can provide valuable insight, which for the power network, may include unprecedented capabilities for forecasting demand, preventing outages and optimising unit commitment. Similarly, data from wind turbine Condition Monitoring

Systems (CMS) may be able to optimise maintenance operations, prevent major breakdowns and reduce wind turbine downtime.

When implementing Big Data practices there are a number of considerations to take into account. At a high level these can be split into hardware and software considerations. Hardware considerations may include the method for storing data and what volume of data will have to be stored locally (as opposed to being stored remotely, or "in the cloud"). The processing speed should also be considered and whether there is a requirement for redundancy of the hardware or the data itself.

Software considerations fall into two broad categories: what approach will be utilised for data storage, and what type of platform will be used to process the data.

Data storage refers to how individual parameters and data values can be saved and retrieved. One common tool for this is MySQL, which is an open source relational database management system which uses structured query language (SQL) to manage data held within the database. Due to many years of information retrieval research and optimisation, SQL databases have the advantage of allowing large volumes of data to be retrieved quickly and efficiently [7]. So-called 'flat files' and more recent paradigms such as NoSQL do not require a rigorous data schema to be developed before application deployment. However, any perceived advantages of flexibility given by this approach may be undermined by the lack of error checking possible at the application level [7].

Big Data processing can be achieved using specialist frameworks such as Hadoop, which allow very large data sets to be processed efficiently [8]. The framework makes use of clusters of standard computing hardware in order to perform parallel computation. This has the result of increasing the speed of processing without the expense of buying special purpose (supercomputer) hardware.

Hadoop is based on the MapReduce programming model, which performs filtering and sorting functions followed by a summary operation, resulting in enhanced handling of very large data sets [8].

Analysis carried out in [9] looks specifically at the performance of the different database frameworks discussed above. Taking large datasets from multiple wind farms, the authors show how execution times differ for three different architectures: MySQL, Hadoop with two nodes, and Hadoop with 48 nodes. The results from the investigations show that the use of Big Data techniques for data storage can reduce execution times. This paper looks more specifically at the data analysis techniques related to Big Data, as opposed to the capture and storage of the data.

This paper discusses the application of Big Data analysis practices for use in wind turbine condition monitoring, with reference to a deployed system capturing 2 TB of data per month. It focuses specifically on the software considerations for Big Data processing of this data.

2.0 Case study: Wind turbine condition monitoring system

A comprehensive wind turbine CMS described in [10] and [11] has been installed in an operational Vestas V47 wind turbine for the purpose of developing algorithms to detect machine deterioration. This system measures a number of parameters, given in Tables 1 and 2, at two sampling rates – a lower sampling rate of 50Hz and a higher sampling rate of 20kHz. The system captures approximately 2 TB of data each month in the form of MySQL MyISAM tables and saves this data onto 2 TB external hard drives which can be 'hot swapped' (taken out while the system is running) as required. The data is then taken to an offsite computer, where detailed analysis may be carried out.

Parameter	Sensor
External air temperature	PT100
Gearbox casing temp	PT100
Nacelle ambient temp	PT100
Generator casing temp	PT100
CMS enclosure temp	PT100
Wind speed	Anemometer
Wind direction	Wind vane
Humidity	Humidity sensor
Low speed shaft rotational speed	Hall effect sensor
XY tower movement	Dual axis accelerometer
Atmospheric pressure	Pressure sensor
Yaw error	Digital compass

Table 1: Parameters measured at 50Hz sampling rate

Parameter	Sensor
Voltage (per phase)	Voltage transducers
Current (per phase)	Rogowski coils
Gearbox (XYZ axis)	Accelerometer
Generator (XY axis)	Accelerometer
Main bearing (XY axis)	Accelerometer
CMS enclosure	Accelerometer

Table 2: Parameters measured at 20kHz sampling rate

The size of the data set had two key impacts on its analysis. Due to the volume of data and the lack of infrastructure at the remote site, it was not possible to transmit the data via the internet, so manual collection was required. Secondly, a number of difficulties arise when trying to work with this volume of data on a standard desktop computer. Programs such as MS Excel, while very familiar to engineers, are not suited to dealing with data files of this size, and so initial analysis was handled using a MySQL server and database with import to Matlab for analysis.

In most scenarios the practicality of working with very large volumes of data results in the data being filtered and/or averaged. This will without doubt significantly reduce the volume of data and therefore the complexity in working with it. As a research tool however, reducing the volume of data through filtering or averaging may result in the loss of valuable information. It is not known ahead of time what knowledge the research is likely to generate, but by storing all of the recorded data there is a greater chance of detecting patterns or anomalies related to the inception of faults.

Using the collection and analysis of data from this system as a case study, this paper investigates and addresses the issues of working with Big Data for the purpose of wind turbine condition monitoring. Comparisons will be made between different approaches of data handling, and recommendations will be given on which are appropriate to the given volumes of data.

2.1 Analysis of CMS data

Wind turbine condition monitoring has the potential to reduce operation and maintenance costs through reduced downtime and optimised maintenance scheduling. Current systems, however, may not necessarily be able to detect all types of deterioration and faults. One reason for this is a lack of high frequency data, which contains enough information to be able to detect machine degradation far enough in advance as to allow remedial actions to be planned effectively. At present, the majority of wind turbines are monitored by supervisory control and data acquisition (SCADA) systems, which provide data only at 10 minute averages, whereas standard machinery diagnostics practices require high frequency vibration monitoring [12].

Regardless of the data source, the detection of deterioration or impending failure may also be described as anomaly detection. The challenges posed by anomaly detection differ depending on the application area, however they are likely to include: defining the threshold of an anomaly, dealing with noise in the data, and the selection of appropriate features to detect [13]. Not only do Chandola et. al [13] highlight the difficulty in selecting the correct detection method, but they also highlight the requirement for efficient computation based on a number of factors including the nature of the data and the type of anomaly being detected.

This detection requires a number of potentially computationally intensive operations, depending on the volume of data required for

detection. The operations required may consist of computationally simple operations, such as threshold detection, or may be highly computationally complex, such as hidden Markov models [14]. It is more likely however that for accurate detection of faults, a combination of high and low complexity operations will be required. Given the large volumes of data under study here, the method of implementation can have a significant effect on the time and computational resources taken to complete data processing.

2.2 Case Study Computations

Computational complexity of an operation can be represented using Big-O Notation [15], which refers to how the computation time scales with respect to the number of data points, n. A computationally efficient algorithm which loops over each data point once would be termed O(n); while a less efficient algorithm which loops over the whole set of data once for each point is $O(n^2)$.

A comparison has been made between CMS data and SCADA data for the case study turbine, to illustrate the necessity for improved data handling techniques. One hour of data sampled at 50Hz from the in-service turbine was read into MS Excel and basic calculation of the mean of 14 variables (including wind speed, rotor speed, and generator temperature) was carried out. This is a relatively computationally simple calculation (O(n)), which can be used to detect trends in behaviour by comparing means between measurement periods (one hour in this case). Table 3 illustrates the difference in the number of rows and the processing time.

	CMS Data	SCADA Data
File Size	8.4MB	10.6kB
Number of Rows (measurements)	180,000	6
Time to calculate mean (seconds)	0.06	0.001

Table 3: Comparison between CMS and SCADA Data

For one hour of data the difference in processing time may seem insignificant. However a day's worth of CMS data will produce 4,320,000 rows of data of which MS Excel can only handle 1,048,576 rows at a time. In comparison, a day's worth of SCADA data is 144 rows, which is many orders of magnitude smaller, and therefore almost any tools can be used for calculation.

There are also significant performance implications for the CMS data when the operation performed becomes more complex than calculation of the mean. The Fast Fourier transform (FFT) (shown in Figure 1) allows time series signals to be analysed in the frequency domain, and is a very common method of analysing vibration data. The standard implementation of the FFT is of order O(n log n) [15].

The amplitude spectrum shown in Figure 1 is the result of performing an FFT on data from an accelerometer mounted on the main bearing. Adjusting the number of samples will have an effect on the frequency resolution of the spectrum.

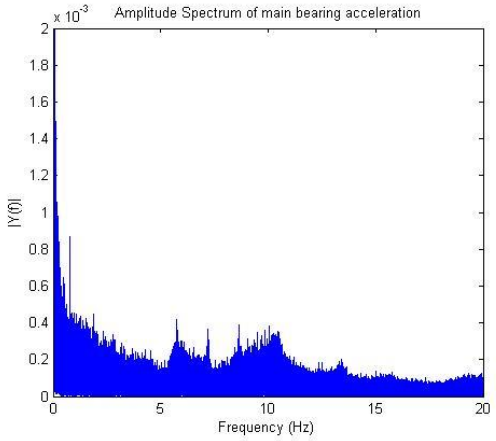

Figure 1: Amplitude spectrum of main bearing acceleration

MS Excel has an add-on function which carries out the FFT. The function has a limit of only being able to process 4096 data values, which for data sampled at 50Hz means only 81.92 seconds of data can be used for the operation. This limit significantly reduces the frequency resolution that can be achieved. For comparison, 4096 data samples were used to calculate the FFT in Matlab.

Table 4 shows the results of the comparison. The statistics show how many milliseconds it

takes to compute each operation. The computation times were recorded 10 times and averaged to reduce variability due to external factors. As can be seen from Table 4 the computation times for performing the operations were an order of magnitude faster when performed in Matlab.

Statistics	Matlab	MS Excel
Minimum (ms)	0.15843	3.48
Maximum (ms)	0.34834	39.43
Mean (ms)	0.19	3.5
Standard Deviation	0.05455	0.02828

Table 4: Computation times for performing an FFT in Matlab and MS Excel

3.0 System Design for Big Data Approach to Wind Turbine Condition Monitoring

As discussed above the FFT is a commonly used technique in signal processing and data analysis. However performing an FFT over large volumes of multiple datasets can be computationally intensive. In this application the requirement is to perform an FFT on 14 different high frequency sensors in order to extract useful information such as key bearing or fault frequencies.

The standard approach is to develop a program which iteratively computes the FFT for each parameter. In this application, for each period of study the program would iteratively compute the FFT for each sensor. This is illustrated by the pseudocode shown in Figure 2.

```
SET n = data length
SET frequencies of interest array

FOR parameter = 1 to 14
    CALL fft with data[parameter]

    FOR i = 1 to n/2
        COMPUTE magnitude of fft[i]
    END FOR

    FOR f in frequencies of interest
        SAVE magnitude[f]
    END FOR
END FOR
```

Figure 2: Pseudocode for computing an FFT

A Big Data approach to developing the same program is shown in Figure 3. Each iteration of the previous program is broken down into a sequence of Map, Reduce, or other higher order functions [8]. In this application, this translates to a Map stage, to perform the FFT to convert the time series data into the frequency domain; a Filter stage, to remove the complex conjugate values; and a set of Reduce operations, which summarise the data by extracting useful frequencies of interest. These steps are replicated for all 14 parameters as illustrated in Figure 4.

The key benefit of this Big Data approach is that each function stage becomes a separate work task, and the work tasks can be distributed over a number of computers (also known as nodes). This reduces the computational load on a single processor. This approach to parallel processing can reduce the total computation time and allow greater data throughput, which may result in earlier anomaly detection. It also allows for future expansion by adding additional nodes to the framework.

A Big Data framework such as Hadoop will automatically handle the assignment of work tasks to each computer. This will ensure all nodes are processing a work task while there are still tasks to be performed. In Figure 3, the tasks relating to different sensors are completely independent, and can be assigned to any node at any time. The tasks relating to a single parameter must be executed in sequence (that is, the FFT must be calculated before the complex conjugates are removed), but each of the Map, Filter, and Reduce tasks can be performed by different nodes as they become available.

This approach to data processing can be extended to other types of analysis. While the FFT is a commonly-used technique, there are other anomaly detection approaches such as parameter correlation, threshold analysis, and hidden Markov modelling. Future work will investigate these other approaches, taking advantage of the Big Data techniques for data handling. It is expected that by allowing larger datasets to be analysed more efficiently, wind

turbine CMS can be fully utilised for the detection of faults and deterioration.

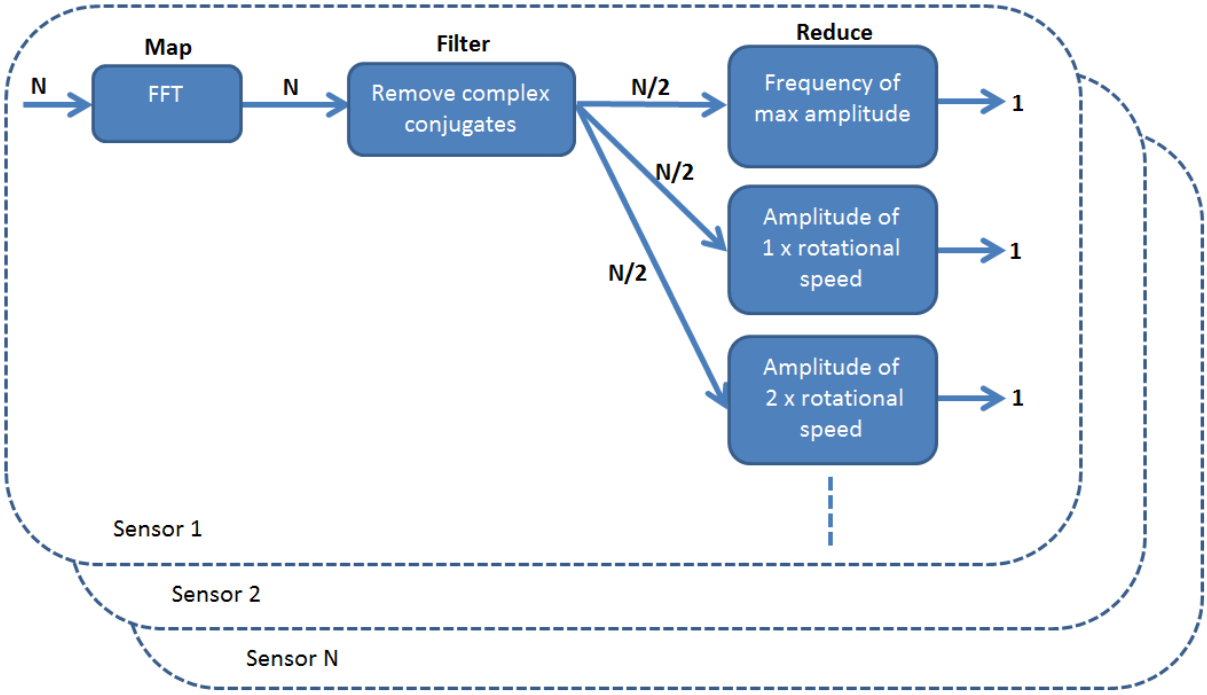

Figure 3: Big Data Technique for reducing computation time

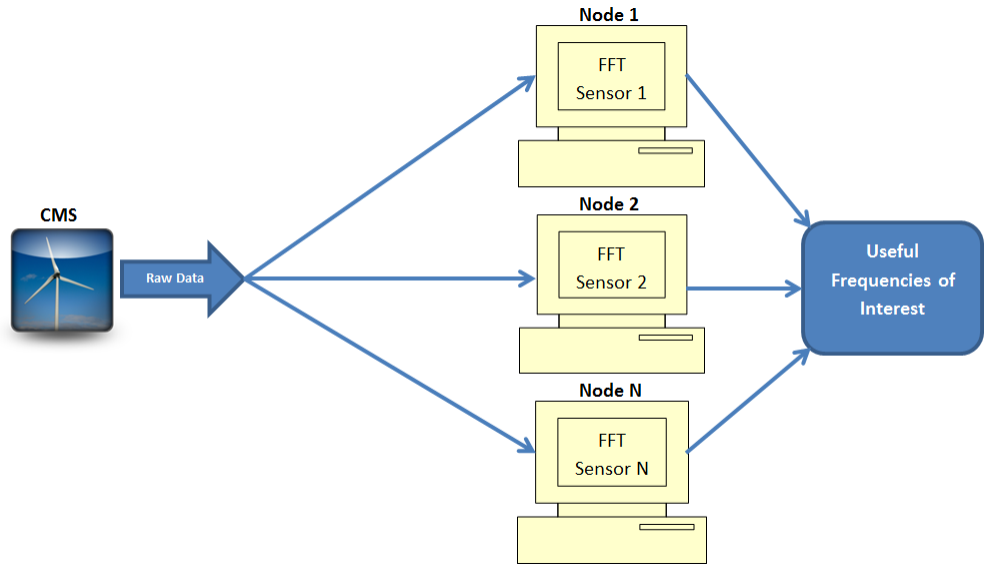

Figure 4: Task sharing across multiple nodes

4.0 Conclusion

Through the application of a wind turbine condition monitoring system, this paper has shown that large volumes of data require improved data handling techniques compared to that of conventional SCADA systems. Where investigations in [9] have shown the need for the correct database frameworks, this paper has highlighted that data analysis programs such as MS Excel are unable to handle the large files which are in excess of 1,000,000 rows of data. One way to improve data handling is through the use of Big Data platforms that can manage and process large volumes of data more efficiently.

Taking the widely-used FFT as a data analysis case study, this paper showed how a traditional iterative approach to data processing could be transformed into a set of functional work tasks for a Big Data platform. This allows tasks to be distributed across different computers, thus increasing throughput and decreasing total analysis time.

The system described in section 2 can produce approximately 2 TB of data per month from one turbine. As technology improves, 2 TB of data may not be considered Big Data; however 2 TB of data from each wind turbine across a wind farm generates data in the petabyte range. Implementing Big Data practices within a wind farm provides the infrastructure to handle this volume, and unlock the information within the data. This may benefit different parties such as technicians, operators, or manufacturers, and allow real time decision making for maintenance action.

References

1. J. Hurwitz, A.N., F. Halper, M. Kaufman, *Big Data For Dummies*. 2013: John Wiley & Sons, Inc.
2. Intel and IBM, *Combat Credit Card Fraud with Big Data*, 2013, Intel Corporation.
3. T.H. Davenport and J. Dyche, *Big Data in Big Companies*, 2013, SAS Institute Inc.
4. Beyer, M. *Gartner Says Solving 'Big Data' Challenge Involves More Than Just Managing Volumes of Data*. 2011 [cited 2013 7/10/2013]; Available from: http://www.gartner.com/newsroom/id/1731916.
5. Kezunovic, M., L. Xie, and S. Grijalva, *The Role of Big Data in Improving Power System Operation and Protection*, in *IREP Symposium - Bulk Power System Dynamics and Control - IX (IREP)*2013: Rethymnon, Greece.
6. Software, I., *Managing big data for smart grids and smart meters*, May 2012.
7. Leavitt, N., *Will NoSQL Database Live Up to Their Promise?* Computer, Feb 2010. **43**(2): p. 12,14.
8. Dean, J. and S. Ghemawat, *MapReduce: a flexible data processing tool.* Communications of the ACM, Jan 2010. **53**(1): p. 72-77.
9. Viharos, Z.J., et al., *Big Data Initiative as an IT Solution for Improved Operation and Maintenance of Wind Turbines*, in *EWEA*2013, EWEA: Vienna. p. 184-188.
10. Zaher, A., et al., *Database Management for High Resolution Condition Monitoring of Wind Turbines*. UPEC: 2009 44th International Universities Power Engineering Conference. 2009.
11. D. Ferguson, et al., *Designing Wind Turbine Condition Monitoring Systems Suitable for Harsh Environments*, in *IET Renewable Power Generation*2013, IET: Beijing.
12. Tavner, P., et al., *Condition Monitoring of Rotating Electrical Machines*. 2008: Institution of Engineering and Technology.
13. Chandola, V., A. Banerjee, and V. Kumar, *Anomaly detection: A survey*, September 2009, ACM Computing Surveys. p. pp. 1-72.

14. Kenyon, A., et al., *An Agent-Based Implementation of Hidden Markov Models for Gas Turbine Condition Monitoring.* IEEE Transactions on Systems, Man, and Cybernetics: Systems, February 2014. **44**(2).
15. Chu, E. and A. George, *Inside the FFT Black Box: Serial and Parallel Fast Fourier Transform Algorithms*, ed. C.P. LLC. 2000.