Determining the integrity of single-source condition-based maintenance data

J. N. de Meyer

orcid.org/0000-0002-9044-4523

Thesis accepted in fulfilment of the requirements for the degree Doctor of Philosophy in Computer and Electronic Engineering at the North-West University

Promoter:	Dr P. Goosen
Co-Promoter:	Dr J. F. van Rensburg

Graduation:	July 2022
Student number:	23452250

Acknowledgements

Firstly, I would like to thank God for guiding me through this process and giving me the strength and endurance to embark on this journey. Without His blessing, this milestone would forever be out of my reach.

To my wife, Marcia, thank you for your never-ending love and encouragement. Thank you for all of the time, energy, words and tears that you spent during this time to help me realise a lifelong goal that I believed was out of my reach. You truly are my greatest inspiration and enable me to achieve so much more than I thought possible.

To my parents, Floors and Sarina, thank you for all of the sacrifices you made for me to reach this point in my life. Without you, none of this would be possible, and words won't ever be enough to express the gratitude I feel for everything with which you have blessed me.

To my new parents, Hendrik and Elanie, thank you for your support during these past few years. Thank you for always giving, motivating, and pushing me forward.

To my study leader, Dr P. Goosen, thank you for the hours and late nights you put in to help me achieve this goal. Despite all of your other responsibilities and my slow working pace, you put in a great deal of effort to ensure that this thesis was completed.

To my co-study leader, Dr J. F. van Rensburg, thank you for your wisdom and help during the final stages of this thesis. Your inputs elevated this document to a new level.

To Prof. E. H. Matthews, thank you for giving me the opportunity to pursue higher education at CRCED Pretoria and ETA Operations (Pty) Ltd for support to complete this study.

To my proofreader, Megan, thank you for the hours you put into this document to ensure the highest possible quality.

To my colleagues, thank you for your support and help during this time. Your contributions might not seem significant to you, but every word of encouragement and cup of coffee pushed me closer to the end.

To my friends and family that supported and motivated me during this time, thank you very much.

Abstract

Data plays a vital role in modern society. With an influx of sensors, data generation is rapidly increasing. Having access to these vast amounts of data enables data-driven decision-making. However, reliable data are needed to make informed decisions, as unreliable data can lead to negative consequences.

Reducing costs has become crucial with the mining industry under enormous pressure to remain profitable. By implementing preventative condition-based maintenance, unnecessary maintenance, downtime and expenses can be avoided. However, reliable data are needed to be effective. Due to financial pressures, often only single-source data are available for planning purposes. Data can be analysed using data quality dimensions from intrinsic and contextual perspectives to evaluate the reliability. The literature identified a need to use the combination of intrinsic and contextual methods to estimate the integrity of single-source condition-based maintenance data.

To address the identified need, this study proposes a novel method to estimate the integrity of single-source condition-based maintenance data using a combination of intrinsic and contextual methods. This method was developed by using data quality dimensions and existing methods from literature.

To supplement the proposed novel method, software system design to estimate the integrity of single-source condition-based maintenance data using a combination of intrinsic and contextual methods is proposed as an additional novel contribution. The software system design makes use of the proposed novel estimation method to evaluate the running status, electrical current drawn, temperature and vibration data streams of pumps, compressors, fridge plants and fans.

A software system was created using the novel contributions. The system was verified using three datasets; two clean datasets and one erroneous dataset, correctly identifying roughly 94.5% of data points. The system was implemented on twenty case study components and identified 22% of data points as unreliable. From the data points identified as unreliable, 69% were identified using the contextual methods, highlighting the need for the combination of intrinsic and contextual methods. A few

common data-related problems were identified, with uncalibrated sensors being the most commonly occurring issue.

Some system limitations were discussed, such as its susceptibility to user errors and the system accuracy being influenced by the data resolution. Future work to improve both the method and the software system is proposed, including an event-based implementation, information display system and notification system. Ultimately, all the identified study objectives were met, delivering the two proposed novel contributions and addressing the identified need for the study.

Keywords: Data integrity, single-source data, condition-based maintenance, condition monitoring.

Publication resulting from the research

An article based on this thesis was published in the South African Journal of Industrial Engineering (SAJIE) on 29 November 2021.

J. N. de Meyer, P. Goosen, J. F. van Rensburg, J. N. du Plessis, J. H. van Laar, "Estimating the reliability of condition-based maintenance data using contextual machine-specific characteristics", *South African Journal of Industrial Engineering (SAJIE) Special Edition*, vol. 32, no. 3, pp. 173-184, Nov, 2021, doi: 10.7166/32-3-2625.

The final accepted manuscript, which is inserted as Appendix A, discusses both novel contributions of this study and presents an implementation of the software system design discussed in Section 1.3.2 of the thesis.

The article was also presented at the 32nd annual conference of the Southern African Institute for Industrial Engineering (SAIIE), held from 4-6 October 2021 in Muldersdrift, South Africa.

Table of contents

Acknowledgements i

Abstract ii

Publication resulting from the research iv

Table of contents v

List of figures vi

List of tables ix

1) Introduction 1

1.1 Background ... 2

1.2 Need for the study ... 10

1.3 Novel contributions ... 16

1.4 Dissertation layout ... 19

2) Design of proposed method and software 20

2.1 Introduction ... 21

2.2 Study restrictions ... 21

2.3 Methodology ... 23

2.4 Investigation ... 24

2.5 Design ... 33

2.6 Summary ... 60

3) Results 61

3.1 Introduction ... 62

3.2 System specifics and verification ... 62

3.3 Criteria for implementation ... 67

3.4 Case studies ... 68

3.5 Discussion ... 83

3.6 Summary ... 85

4) Conclusion 86

4.1 Discussion ... 87

4.2 Recommendations for future work ... 89

References 92

Appendix A: Publication 99

Appendix B: Verification results 112

Appendix C: Case study results 120

List of figures

Figure 1: Simplified organisational diagram 22
Figure 2: Methodology diagram 23
Figure 3: Illustration of relations between measured characteristics 27
Figure 4: Diagram illustrating the data collection process for condition monitoring 34
Figure 5: SQL database design 36
Figure 6: Overview of the proposed software system 38
Figure 7: Diagram illustrating the overview of the data integrity system 38
Figure 8: Diagram of system flow for intrinsic integrity calculation 39
Figure 9: System flow for calculating contextual integrity 40
Figure 10: Illustration of normalised value sections 40
Figure 11: Overview of characteristic contextual integrity calculations 42
Figure 12: Contextual integrity flow of flat sections for the running status 44
Figure 13: Contextual integrity flow of positive sections for the running status 45
Figure 14: Contextual integrity flow of negative sections for the running status 46
Figure 15: Contextual integrity flow of flat sections for electrical current 48
Figure 16: Contextual integrity flow of positive sections for electrical current 49
Figure 17: Contextual integrity flow of negative sections for electrical current 50
Figure 18: Contextual integrity flow of flat sections for temperature 53
Figure 19: Contextual integrity flow of positive sections for temperature 54
Figure 20: Contextual integrity flow of negative sections for temperature 55
Figure 21: Contextual integrity flow of flat sections for vibration 57
Figure 22: Contextual integrity flow of positive sections for vibration 58
Figure 23: Contextual integrity flow of negative sections for vibration 59
Figure 24: Normalised values of an always-running clean dataset 63
Figure 25: Clean dataset verification summary 63
Figure 26: Normalised values of a state-switching clean dataset 64
Figure 27: State-switching clean dataset verification summary 65
Figure 28: Normalised values of the erroneous dataset 66
Figure 29: Erroneous dataset verification results 66
Figure 30: Component B data reliability for January to December 2020 69
Figure 31: Analysis of component B reliability by characteristic 69
Figure 32: Uncalibrated vibration sensor for Component B (4 January 2020) 70
Figure 33: Severe sensor issues identified for Component B (31 December 2020) 71
Figure 34: Component D data reliability 72
Figure 35: Analysis of component D reliability by characteristic 72
Figure 36: Uncalibrated vibration sensor for Component D 73
Figure 37: Component H data reliability 73
Figure 38: Analysis of component H reliability by characteristic 74
Figure 39: Component H characteristic profile (13 April 2020) 75
Figure 40: Component P data reliability 75
Figure 41: Analysis of component P reliability by characteristic 76
Figure 42: Diurnal characteristic profiles for Component P (25 September 2020) 76

Figure 43: Component Q data reliability 77
Figure 44: Analysis of component Q reliability by characteristic 78
Figure 45: Component R data reliability 78
Figure 46: Analysis of component R reliability by characteristic 79
Figure 47: Compressor combined data reliability 79
Figure 48: Fan combined data reliability 80
Figure 49: Fridge plant combined data reliability 80
Figure 50: Pump reliability results percentage breakdown 81
Figure 51: Case study summarised reliability results percentage breakdown 83
Figure 52: Summarised method reliability results percentage breakdown 83
Figure 53: Clean running verification results 112
Figure 54: Clean electrical current verification results 113
Figure 55: Clean temperature verification results 113
Figure 56: Clean vibration verification results 114
Figure 57: State-switching clean running verification results 114
Figure 58: State-switching clean electrical current verification results 115
Figure 59: State-switching clean temperature verification results 116
Figure 60: State-switching clean vibration verification results 116
Figure 61: Erroneous dataset running verification results 117
Figure 62: Erroneous dataset electrical current verification results 118
Figure 63: Erroneous dataset temperature verification results 118
Figure 64: Erroneous dataset vibration verification results 119
Figure 65: Component A reliability results percentage breakdown 120
Figure 66: Component A reliability results breakdown 120
Figure 67: Component B reliability results percentage breakdown 121
Figure 68: Component B reliability results breakdown 121
Figure 69: Uncalibrated vibration sensor for Component B (4 January 2020) 122
Figure 70: Severe sensor issues identified for Component B (31 December 2020) 123
Figure 71: Component C reliability results percentage breakdown 124
Figure 72: Component C reliability results breakdown 124
Figure 73: Component D reliability results percentage breakdown 125
Figure 74: Component D reliability results breakdown 125
Figure 75: Uncalibrated vibration sensor for component D (1 January 2020) 126
Figure 76: Component E reliability results percentage breakdown 127
Figure 77: Component E reliability results breakdown 127
Figure 78: Component F reliability results percentage breakdown 128
Figure 79: Component F reliability results breakdown 128
Figure 80: Component G reliability results percentage breakdown 129
Figure 81: Component G reliability results breakdown 129
Figure 82: Component H reliability results percentage breakdown 130
Figure 83: Component H reliability results breakdown 130
Figure 84: Diurnal characteristic profiles for Component H (13 April 2020) 131
Figure 85: Component I reliability results percentage breakdown 132
Figure 86: Component I reliability results breakdown 132

Figure 87: Component J reliability results percentage breakdown 133
Figure 88: Component J reliability results breakdown 133
Figure 89: Component K reliability results percentage breakdown 134
Figure 90: Component K reliability results analysis 134
Figure 91: Component L reliability results percentage breakdown 135
Figure 92: Component L reliability results breakdown 135
Figure 93: Component M reliability results percentage breakdown 136
Figure 94: Component M reliability results breakdown 136
Figure 95: Component N reliability results percentage breakdown 137
Figure 96: Component N reliability results breakdown 137
Figure 97: Component O reliability results percentage breakdown 138
Figure 98: Component O reliability results breakdown 138
Figure 99: Component P reliability results percentage breakdown 139
Figure 100: Component P reliability results breakdown 139
Figure 101: Diurnal profiles for Component P (25 September 2020) 140
Figure 102: Component Q reliability results percentage breakdown 141
Figure 103: Component Q reliability results breakdown 141
Figure 104: Component R reliability results percentage breakdown 142
Figure 105: Component R reliability results breakdown 142
Figure 106: Component S reliability results percentage breakdown 143
Figure 107: Component S reliability results breakdown 143
Figure 108: Component T reliability results percentage breakdown 144
Figure 109: Component T reliability results breakdown 144
Figure 110: Two-minute resolution dataset reliability percentage results 145
Figure 111: Two-minute resolution dataset reliability results 145
Figure 112: Thirty-minute resolution dataset reliability percentage results 146
Figure 113: Thirty-minute resolution dataset reliability results 146

List of tables

Table 1: Commonly used maintenance strategies in the mining industry 4
Table 2: Common measurement techniques used for condition-based maintenance 6
Table 3: Commonly measured characteristics of critical mining equipment 7
Table 4: Common data quality dimensions 9
Table 5: Data quality dimensions used for different data categories 10
Table 6: Literature review matrix 12
Table 7: Truth table used for contextual data reliability calculations 32
Table 8: Contextual integrity acceptable score ratio 33
Table 9: Breakdown of tag document attributes 35
Table 10: Breakdown of value document attributes 35
Table 11: Breakdown of SQL database design components 36
Table 12: Software system model layout 37
Table 13: Software system model custom class layout 37
Table 14: The main components in the proposed software system 38
Table 15: Description of the intrinsic data quality steps 39
Table 16: Characteristic section analysis 41
Table 17: Running status perspective flat section evaluation 43
Table 18: Running status perspective positive and negative section evaluation 43
Table 19: Electrical current perspective characteristic evaluations 47
Table 20: Temperature perspective flat section characteristic evaluation 51
Table 21: Temperature perspective characteristic evaluation for positive and negative sections 52
Table 22: Vibration perspective characteristic evaluation 56
Table 23: Implemented software system specifics 62
Table 24: Case study components by type and site 67
Table 25: Summarised data reliability by site (number of data points) 81
Table 26: Summarised data reliability by characteristic (number of data points) 82
Table 27: Reliability methods results summarised 82
Table 28: Summarised results for data resolution difference case study 85
Table 29: Breakdown of implemented study objectives 89

Chapter 1

Introduction

1.1 Background

1.1.1 Importance of data

Data plays a vital role in modern society [1]. As newer technologies emerge, data becomes more accessible. Technologies such as the Internet of Things (IoT), Cloud computing and Artificial Intelligence (AI) have facilitated a shift in society where data is now considered an asset [1]-[11]. With digital networks now connecting more people, devices and sensors; an astonishing amount of data is generated daily [12], [13]. This expansion of networks can be partly attributed to the improvements in sensing technologies, wireless networks, low power electronics, and battery technology [14]. Due to these advancements, sensors can now be found in an abundance of devices [15], [16].

Data generation has steadily increased over the last few years due to this influx of sensors [6]. This is evident when data measured in terabytes[1] a decade ago is now measured in petabytes[2] [17]. It was predicted that the size of the digital universe would grow by a factor of 300 between 2005 and 2020, and that the rate of data generation would double every two years [18]-[20]. In 2010, over 1 Zettabyte[3] (ZB) was generated worldwide, which increased to 1.8 ZB by 2011 [6], [15], [20] and 7 ZB [20] by 2014. By 2017, it was estimated that 90% of all data in the world was generated during the previous two years [20], [21]. In 2018, 2.5 quintillion[4] bytes were generated every day [20], and by 2020, the total amount of data generated was predicted to exceed 35 ZB [6], [15], [22], [23].

This increase in available data creates an opportunity to improve business decision-making [13], [15], [22]. Actions initiated by informed decisions can lead to a market advantage [24]. As a result, business leaders have started adopting data-driven decision-making [10], [13], [25], [26].

[1] Terabyte = 10^{12} bytes
[2] Petabyte = 10^{15} bytes
[3] Zettabyte = 10^{21} bytes
[4] Quintillion = 10^{18}

Data-driven decision-making refers to the use of data in decision-making to promote more effective insights [25]. Some of the advantages of effective data-driven decision-making include [22], [25]:

- improved insights of information value,
- creation of new business opportunities, and
- more timely decision-making.

Large quantities of data have to be analysed in order to improve the decision-making process [2]. However, for decision making to be effective, high-quality data is required [1], [13], [27]-[30]. Conversely, making decisions based on low-quality data can have negative results, such as [1], [22], [25], [27], [29], [31]:

- employee dissatisfaction,
- customer dissatisfaction,
- increased costs,
- wasted time,
- low-data utilisation,
- inefficient decision-making processes, and
- lower business performance.

Research has shown that decisions are often made without considering the quality of the data, and the quality of the data is overestimated [22], [28], which can result in businesses facing extreme financial consequences [27], [32]. For industries under financial pressure, making decisions based on poor quality data is becoming increasingly high-risk.

One industrial sector currently under severe pressure is the mining industry [17], [33]-[35]. In South Africa, the mining industry is struggling to remain profitable due to [22], [33], [35]-[39]:

- lower global demand,
- increased domestic costs,
- labour disruptions,
- increased electricity tariffs,
- decreased commodity prices,
- declining ore grades, and
- reduced productivity.

By reducing unnecessary expenditure, South African mines increase their hopes of remaining profitable.

One of the largest operational expenses on South African mines is maintenance, accounting for 25-40% of overall equipment costs and 20-50% of the total operational costs [40], [41]. This is partly due to the harsh conditions under which these equipment operate [41]. The equipment is critical for the mine's productivity, leaving little room for downtime [41], [42]. Regular maintenance is required to ensure the equipment remains operational and reduce unexpected breakdowns [43].

The different maintenance strategies that are commonly applied to the equipment for schedule planning are described in Table 1 [41], [44], [45].

Table 1: Commonly used maintenance strategies in the mining industry

Maintenance strategy	Description
Breakdown	Repairs are reserved for failed or broken-down equipment.
Corrective	Component upgrades to improve reliability with a flawed design.
Preventative time-based	Makes use of a predetermined schedule.
Preventative condition-based	Applied depending on the current condition of the equipment.
Reliability-centred	Only perform preventative maintenance on system-critical equipment.
Total productive	Operators are responsible for production responsibilities and reporting on maintenance needs.

From the commonly used maintenance strategies described in Table 1, preventative maintenance is most often applied, as it aims to avoid expensive repair and replacement costs while also avoiding unplanned downtime [40], [41]. Condition-based maintenance can be used to further reduce unnecessary expenses [43].

1.1.2 Condition-based maintenance

Condition-based maintenance is a preventative maintenance strategy aimed at reducing maintenance costs. This is achieved by scheduling maintenance based on the equipment's current condition. Condition-based maintenance comprises three steps, namely [22], [43], [46], [47]:

- Data acquisition – This step consists of collecting data from different sources to learn about the equipment health and operating condition.
- Data processing – This step analyses the data gathered during the data acquisition step to better understand the system’s behaviour and patterns.
- Maintenance decision and planning - This is the final step in which the best maintenance times are recommended based on the insight obtained during the data processing step.

Effective condition-based maintenance depends on reliable data [43]. For the data processing step to be effective, the data acquisition step needs to deliver reliable data. Condition-based maintenance strategies rely on condition-monitoring infrastructure [48].

Condition monitoring uses sensors to monitor the condition of critical equipment and is a central part of condition-based maintenance [48]. The data collected from a condition-monitoring system gives an in-depth view of the equipment's operation and health, assisting with fault diagnostics and prognostics [44], [47].

To this end, condition monitoring uses many sensors [49]; however, condition-monitoring equipment is expensive to maintain, and operation requires specialist knowledge [40]. Due to these limitations, different monitoring strategies can be utilised depending on the installed equipment and monitoring needs.

Different measurement techniques can be used depending on the equipment that needs to be measured and the measurement purposes. Standard measurement techniques are discussed in Table 2 [48].

Table 2: Common measurement techniques used for condition-based maintenance

Technique	Measurements
Dynamic monitoring	Vibration Sound
Temperature monitoring	Temperature
Chemical monitoring	Wear Leaks Corrosion
Particle monitoring	Wear Fatigue Corrosion Contaminants
Physical monitoring	Cracks Fracture Wear Deformation
Electrical monitoring	Resistance Conductivity Dielectric strength Equipment used
Human inspection	Based on human senses

1.1.3 Single-source data

The accuracy of the measurements can vary depending on the quality and the number of sensors used. Multiple sensors per piece of equipment can be installed to increase the number of available data streams per characteristic [19], [22] and hence increase the effectiveness of condition-based maintenance. However, due to the financial pressure of remaining profitable, installing additional sensors is not always viable [50]. Financially constrained mines may have limited sensor equipment, making single-source data the only option [34].

Single-source data refers to data points measured by only a single source. Bound by these constraints, condition monitoring systems are often installed on the most critical components and only measure the most critical characteristics.

Critical components can be classified as those crucial to operations and production in the mining industry. These components can often be identified by investigating which components are most commonly used in literature. From literature, the main components in the mining industry that are either investigated, optimised or commonly monitored with condition-monitoring systems are [51]-[58]:

- pumps,
- fans,
- compressors, and
- cooling systems, such as fridge plants.

Each of the components listed above differs, and different characteristics can be measured. However, some characteristics are found on each component regularly measured in condition-monitoring systems [50], [59], [60]. The shared commonly measured characteristics for the abovementioned components are listed in Table 3. The combined four characteristics in Table 3 give an overview of the operational status of each component at any point in time.

Table 3: Commonly measured characteristics of critical mining equipment

Characteristic	Description	Unit
Running status	Records the on (1) or off (0) status. It is commonly connected to the on-off switch of the component or controlled by programmable logic controllers (PLCs) or SCADA.	–
Electrical current	Records the current drawn by the component. The value range is specific to the component and can vary depending on the component type.	A
Temperature	Records the temperature of the component at a specific place. The values differ depending on the component and the conditions.	°C
Vibration	Records the vibration experienced by the component. The vibration experienced by a component is the sum of the vibration generated by the component and caused by other components.	mm/s

Measurements are never 100% error-free due to sensor accuracy, sampling rate, data integration and sensor drift [61]-[65]. Another factor influencing data quality is the harsh environment in which the sensors operate [49]. As a side effect, the condition of the equipment deteriorates over time, which in turn could produce more errors [43]. When sensors include errors in the measurements, even the best data source can become a liability [19]. To mitigate these risks, the integrity of the data can be investigated.

1.1.4 Data integrity and quality

Often used interchangeably, data quality and integrity are related, yet they are not the same. Data quality describes the fitness or suitability of data and is specific to the use thereof [6], [22], [31], [64]. Data integrity is the completeness of data compared to the integrity of the objective world, requiring all data values to be in an objective and true state and not empty [66]. Simply put, data integrity can be seen as the trustworthiness of data [22].

Quality data produces trustworthy knowledge; as a result, data quality can be used as a building block towards data integrity [22]. As data quality is specific to the use of the data, many dimensions exist to characterise data [1], [22]. These dimensions are measurable data quality properties representing some aspect of the data [1]. Data quality can be estimated using a combination of dimensions [67]. The most commonly used data quality dimensions identified in literature are listed in Table 4 [1], [4], [6], [22], [28], [67]-[94].

As data quality is application-specific, deciding which dimensions to use is influenced by the data and the application. With the increasing interest in data quality as a crucial research topic, two main categories of data quality have been established, namely intrinsic and contextual [28], [67], [71], [72], [76].

Table 4: Common data quality dimensions

Dimension	Description
Accessibility	Is the data easily accessible, usable and retrievable?
Accuracy	Is the data correct, objective, reliable, certified and validated?
Availability	Is the data available?
Believability	Is the data true and credible?
Completeness	Are all the values that are supposed to be in the collection there?
Compliance	Does the data comply with regulatory and industry standards?
Consistency	Is the data consistent and presented in the same format?
Integrity	Is the data coherent?
Objectivity	Is the data unbiased, unprejudiced and impartial?
Relevance	Is the data applicable to the task at hand?
Reliability	Is the data correct and reliable?
Timeliness	How long does a change take to reflect between the real-world state and the corresponding data?
Validity	Is the data within acceptable parameters?

Intrinsic data quality, also known as isolated data quality, refers to data analysed in isolation and depends on the data itself without considering the context in which the data is used [22], [28], [76]. Previous studies have successfully used intrinsic data quality to detect faulty sensors, as described by Byabazaire et.al [90].

Contextual data quality takes other factors, such as the purpose of the data and physical limitations, into consideration [22], [28], [76]. Adding context to the data becomes paramount when combining different data sources [90].

As these two categories of data quality differ, different dimensions are used when determining the quality. Common dimensions for each category are listed in Table 5 [6], [22], [67], [72], [84], [85], [90].

As seen in Table 5, the dimensions used to determine data quality vary depending on the data used and the application. A more accurate result can be calculated by using both methods. Thus, the combined integrity can be calculated by combining intrinsic and contextual data quality methods [28].

Table 5: Data quality dimensions used for different data categories

Dimension	Intrinsic	Contextual
Accessibility		X
Accuracy	X	X
Availability	X	
Believability	X	X
Completeness	X	X
Compliance		
Consistency	X	X
Integrity		X
Objectivity		
Relevance		X
Reliability		X
Timeliness	X	X
Validity	X	

1.2 Need for the study

1.2.1 Research methodology

Based on the initial research from Section 1.1, further research was conducted to identify literature related to the following topics:

- data integrity,
- single-source condition-monitoring data,
- intrinsic data integrity,
- contextual data integrity, and
- condition-based maintenance in the mining industry.

A focus was placed on finding relevant literature through accredited search portals and online libraries: IEEE Xplore, Google Scholar, Scopus, Science Direct, Springer Link, and NWU online library. Literature found through the abovementioned portals was considered if it met the following criteria:

- Was the study conducted in the past ten years?
- Was the study of high quality (journal, reviewed conference proceedings or thesis)?

After sufficient literature was identified, they were categorised according to the following criteria:

- Did the study evaluate the integrity/trustworthiness of data?
- If the study evaluated data integrity, was it done intrinsically or contextually?
- Was the study done on condition-based maintenance data?
- Was the study conducted in the mining industry?
- Did the study have access to single-source or multi-source data?

The categorised literature was evaluated and only considered if it met at least one of the above-mentioned criteria. During the evaluation, shortcomings were identified in existing literature. Data integrity is commonly investigated in studies relating to data stored in databases or the cloud environment. As such, studies that investigate the integrity of condition-based maintenance data are scarce. The existing studies investigating the data integrity of condition-based maintenance focus on either intrinsic integrity or contextual integrity, never a combination of methods. This highlights the need for using the combination of intrinsic and contextual methods to determine the integrity of condition-based maintenance data.

Despite the increase in sensor availability and affordability, the mining industry seldom implements multiple sensors to measure the same characteristic on a piece of equipment, leading to predominantly single-source data. Combining single-source data availability with the financial pressures of remaining profitable, a need was identified to investigate the reliability of single-source condition-based maintenance data to improve data-driven decision-making confidence.

A further need was thus identified to estimate the integrity of single-source condition-based maintenance data using the combination of intrinsic and contextual integrity methods to help with data-driven decision-making.

1.2.2 Literature review

Using the research methodology described in Section 1.2.1, literature related to this study was identified and is listed in Table 6.

Table 6: Literature review matrix

Reference	Data		Data integrity			System	
	Condition-based maintenance	Single-source	Intrinsic	Contextual	Combined	Implemented	Generic
[61]			✓				
[34]		✓				✓	✓
[49]		✓				✓	
[1]		✓	✓				
[63]			✓			✓	
[28]		✓		✓			
[43]	✓			✓			
[6]	✓	✓	✓			✓	✓
[22]	✓	✓		✓		✓	✓
[19]			✓			✓	
[94]		✓	✓				
[70]		✓	✓			✓	
[73]			✓				
[67]				✓		✓	
[75]		✓	✓				

Hamer [61] created a framework to evaluate the data quality of data used for tax-rebate purposes from a measurement and verification perspective. The measurement and verification process emphasises the quality of the data and promotes multi-source data to ensure quality. For this study, Hamer discussed some of the common data errors found in electrical measurement data in the mining industry and evaluated the quality from an intrinsic perspective.

Van Jaarsveld et al. [34] created a system to simplify the information retrieved from single-source condition-monitoring data. They implemented the system in the mining industry and used a variety of components in their case study, showcasing the

genericism of their system. Unfortunately, they did not evaluate the quality of the data used for the system.

Xu et al. [49] created a system to detect abnormal sections in condition-monitoring data. Single-source data was used for their system; however, they did not use the data quality dimensions when analysing the data streams. The system was implemented on a case study for a wind turbine without indicating whether the system could be applied to different components.

Laranjeiro et al. [1] thoroughly investigated data quality in the existing literature and presented the results. They discussed data quality dimensions on both single-source and multi-source data. Unfortunately, this was only an investigative study and was limited to a literature review.

Gous et al. [63] investigated the quality of electricity data for measurement and verification purposes. To ensure they evaluated the quality accurately, they used multi-source data and investigated the quality from an intrinsic perspective. The system was implemented on a case study; however, it was applied to a specific case and did not discuss the system's ability to be applied generically for different types of data.

Moges et al. [28] investigated the impact of data quality on decision making by taking data context into account. They evaluated relevant literature to determine whether contextual information should be stored in data warehouses and concluded that the information should be investigated per case to establish whether it would be beneficial.

Madhikermi et al. [43] performed two case studies in which they investigated key data quality pitfalls for condition-based maintenance data. The available data were investigated for this study to determine where issues arose. Having access to multiple data sources, the researchers discovered that data context was not fully applied in the existing data, resulting in distinct datasets with no relation.

Goosen [6] created a system to determine the quality of single-source condition-based maintenance data from an intrinsic perspective. Goosen developed the system using existing literature and ultimately applied it to multiple components. She suggested that contextual information would benefit the system.

De Meyer [22] developed a software system to investigate the integrity of single-source condition-based maintenance data. He applied the system to a case study involving four different component types following a simplified contextual approach. The results suggested that a combination of intrinsic and contextual methods should be investigated.

Coetzee et al. [19] applied a system to multi-source electricity usage data to determine the quality from an intrinsic perspective. They discussed a few common quality-degrading occurrences and their impact on the overall dataset, which has various ripple effects in the measurement and verification space.

Zhang [94] investigated a method to improve data quality in a big data environment. A method was created and applied on a case study using single-source data and an intrinsic approach to show how it can improve the data quality.

Ferney et al. [70] implemented a software system to evaluate data quality received from an API[5]. They evaluated the single-source data from an intrinsic perspective using three data quality dimensions: traceability, completeness, and compliance.

Lee [73] used an intrinsic approach to investigate data quality employing key concepts such as measurement, traceability, and uncertainty. The researcher compiled a matrix of data quality dimensions from other recent studies that could be used for her study. Despite a simplistic method described to evaluate data quality, the author did not apply it to a case study.

Juneja and Das [67] used existing literature to discuss the difference between intrinsic and contextual data quality dimensions. Using the identified contextual dimensions, they created a weather monitoring and forecasting application that accepts data from multiple sources; however, it was not implemented on a case study.

Tian Hongxun et al. [75] created a method to assess the data quality for online monitoring. Using this method, they evaluated single-source transformer power quality data with an intrinsic approach.

[5] API – Application Programming Interface

1.2.3 Problem statement

From Table 6 and the literature review, the following shortcomings were identified:

a) There are limited studies that investigate the integrity of single-source condition-monitoring data.
b) Although there is a fair split between intrinsic and contextual data quality methods, no study was identified that uses the combination of these methods to determine the integrity of single-source condition-based maintenance data.
c) When a system was implemented to evaluate data integrity, it was often designed with a specific implementation in mind and was thus not reusable across different applications.

Based on the gaps identified in the literature, it was determined that a formalised estimation method to determine the overall integrity of single-source condition-based maintenance data is lacking. There is also no available system design to systematically determine the overall integrity of single-source condition-based maintenance data, disregarding the component type.

1.2.4 Study objectives

To satisfy the aim of this study and address the identified shortcomings in literature, the following objectives and sub-objectives were identified:

- Create a method for estimating the integrity of single-source condition-based maintenance data.
  - Compile a list of usable intrinsic data quality methods:
    - identify suitable methods from literature, and
    - adapt methods, if needed.
  - Compile a list of usable contextual data quality methods:
    - identify suitable methods from literature, and
    - adapt methods, if needed.
  - Combine intrinsic and contextual data quality methods into a formalised method.
- Develop a software system design that estimates the integrity of single-source condition-based maintenance data.
  - Implement newly created method into a software system design
  - Create a software system using the design
  - Verify the software system
  - Implement the software system on case studies

1.3 Novel contributions

This study proposes two novel contributions to address the need identified in Section 1.2. Each novel contribution is discussed in four sections, namely:

- problem statement,
- limitations of existing research,
- research question, and
- contribution of this study.

1.3.1 Novel contribution 1

A method to estimate the integrity of single-source condition-based maintenance data using the combination of intrinsic and contextual data integrity methods

Problem statement

The South African mining industry is under financial pressure to remain profitable. To reduce unnecessary expenditure, maintenance schedules can be optimised by using condition-based maintenance; however, for this to be effective, reliable data is needed. Due to financial constraints, single-source data is often more accessible due to the limitation of expensive measuring equipment. Data reliability can be estimated from both an intrinsic and contextual perspective, each favouring different data quality dimensions. However, a combined method using both intrinsic and contextual methods do not exist.

Limitations of existing research

As data reliability becomes essential, the topic becomes more widely researched. However, few studies have focussed on the reliability of condition-based maintenance data in the mining industry. Applying the restriction of only having single-source data available further limits the number of studies. Analysing data, especially at high resolution and frequency, requires substantial resources and costs. As a result, analysing data from an intrinsic perspective can reduce the time and resources required. Expert knowledge of equipment and their working is required to analyse data effectively in context, which is scarce and costly. Thus, most current studies that focus on the reliability of condition-based maintenance data in the mining industry focus on

the intrinsic reliability of data. From the literature review, few studies focus on contextual reliability, and no studies focus on combining intrinsic and contextual data-integrity methods.

Research question

Is it possible to create a simplified method that estimates the reliability of single-source condition-based maintenance data using the combination of intrinsic and contextual data-integrity methods?

Contribution of this study

Existing data quality dimensions were identified from the literature and categorised depending on whether they are used for intrinsic or contextual data integrity perspectives. For intrinsic reliability, equations were compiled from the data quality definitions and existing studies. The relations between different characteristics were investigated for contextual reliability. Representative equations were derived from existing literature and knowledge on how standard mining equipment operates. An equation was compiled to combine the intrinsic and contextual data-integrity methods. These equations form a method to estimate the reliability of single-source condition-based maintenance data.

1.3.2 Novel contribution 2

A software system design that estimates the integrity of single-source condition-based maintenance data using the combination of intrinsic and contextual data integrity methods

Problem statement

Data is becoming a vital asset for businesses due to advancements in technology and a drive to a data-driven world. For businesses to extract valuable knowledge and gain better insight, the data must be reliable as using unreliable data, and the knowledge extracted therefrom, can lead to serious negative financial implications. Various factors have placed the South African mining industry under severe pressure to remain profitable. Condition-based maintenance can improve maintenance schedules and reduce unnecessary downtime. However, as mentioned above, reliable data is

required for the derived maintenance recommendations to be accurate and efficient. Analysing the required large data streams in real-time is resource-intensive. Additionally, single-source data is often more widely accessible. Expert knowledge is required to analyse the data as mining machinery is complex and often forms part of more extensive inter-connected systems.

Limitations of existing research

From the literature review, it was found that few studies focus on single-source condition-based maintenance data reliability. From the studies identified, software systems are often recommended as they are more efficient at analysing large quantities of data and can be left unattended with repeatable accuracy. However, to improve the accuracy of the software systems, they are generally designed for a specific dataset or machine. This requires expert knowledge of the specific dataset or machine to calibrate the software system accurately. Obtaining this expert knowledge can be challenging and comes at a cost. Current studies do not investigate a simplified software design that would negate the need for in-depth expert knowledge.

Research question

Can a simplified software system be designed to estimate the reliability of single-source condition-based maintenance data and remove the need for expert knowledge by using equipment behaviour estimations that can be applied to various types of machines?

Contribution of this study

This study proposes a generic software system design to estimate the reliability of single-source condition-based maintenance data using the combination of intrinsic and contextual data integrity methods. The proposed design can be applied to any machine, regardless of type or industry, providing it meets the criteria for implementation. The design abstracts the implementation by suggesting technology types instead of specific technologies.

1.4 Dissertation layout

Chapter 1 introduced the key concepts that are relevant to this study. Following the background, a literature review identified several shortcomings. These shortcomings highlighted a need for the study and formalised a problem statement. From the problem statement, study objectives were defined, and the novel contributions of this study were discussed.

Chapter 2 discusses the design of the method and software system as identified in the study objectives and novel contributions. In this chapter, existing methods from literature and new methods are discussed. How they address the identified need will be explored.

Chapter 3 elaborates on how the software system designed in Chapter 2 was implemented on various case studies. This chapter discusses the results obtained from the case study implementations and analyses the system performance.

Chapter 4 concludes with a summary of the study and highlights how and where the objectives were addressed along with recommendations for future studies.

Chapter 2

Design of proposed method and software

2.1 Introduction

Chapter 2 defines the methodology used for this thesis and is divided into six sections. Section 2.2 lists the scope and restrictions applicable to this study and establishes the conditions for this study. Section 2.3 discusses the methodology used for this study and illustrates where the study objects identified in Section 1.2.4 will be addressed. Section 2.4 consists of the investigation into the literature to identify existing data quality methods that can fulfil the study objects and suggests new methods that can potentially fill the gaps.

Section 2.5 proposes a method to estimate the integrity of single-source condition-based maintenance data. The proposed method is then applied to a software system design that can be used to determine the integrity of single-source condition-monitoring data. Section 2.6 concludes the chapter and summarises the main points.

2.2 Study restrictions

In Section 1.2.3, three main shortcomings were identified, namely:

- a lack of literature for investigating the integrity of single-source condition-based maintenance data,
- no recorded method of using the combination of intrinsic and contextual data quality methods to determine the overall data integrity, and
- systems that estimate data integrity are scenario-specific and are not implementable for a range of components.

This chapter will address these shortcomings; however, some restrictions will be applied to define an achievable scope. Single-source condition-based maintenance data typically found in the mining industry will be used when considering data. This limits the data structure and range of possible values. Data access for case studies and results are also limited to the mining industry. Secondly, when considering components, the four components identified in Section 1.1.3 will be used, namely pumps, fans, compressors, and fridge plants. Thirdly, only the four commonly measured characteristics identified in Section 1.1.3 will be considered: running status, electrical current drawn, temperature, and vibration.

For the aim of this study, the organisational structure of mines will be simplified and represented by a generic structure. Figure 1 illustrates this generic organisational

structure and the terminology used throughout this study and how they relate to each other.

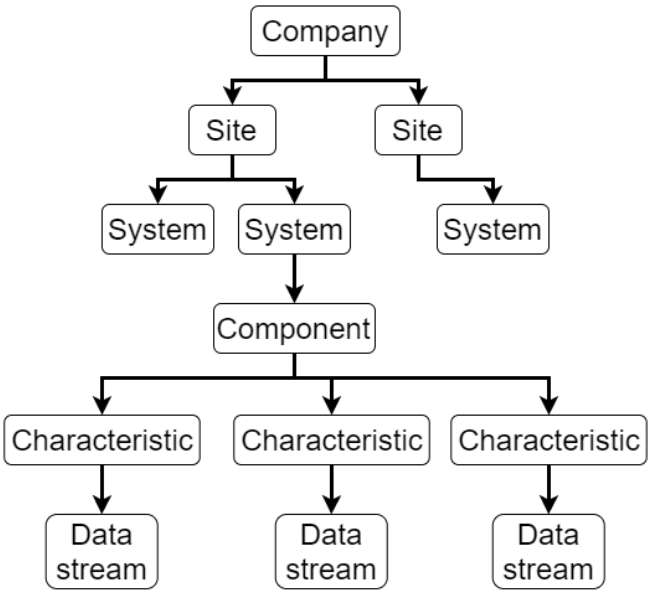

Figure 1: Simplified organisational diagram

Starting from the top and moving down, the structure refers to the following:

- A company is the highest level of an organisation.
- A site represents a mine that belongs to the company. Each company can have multiple sites, which can be located throughout the world.
- Every site comprises multiple systems, each with its own focus. In the case of a mine, these systems include ventilation, cooling, pumping and refrigeration.
- Components refer to the equipment or machines used within the system. Systems can have multiple components of the same type linked to them. For example, a cooling system can have multiple fridge plants or chillers.
- Every component has multiple characteristics monitored by sensors as part of condition monitoring. The data generated for these characteristics will be used when maintenance schedules are calculated using condition-based maintenance.
- Every characteristic being measured produces a data stream. Due to this study focussing on single-source data, each characteristic is assumed only to have one data stream linked to it.

This generic structure ensures that the methods developed in this study can be applied to any mine that uses a similar structure. In certain circumstances, these methods can also be applied to organisations outside the mining industry that follow a similar organisational structure, such as the vehicle and food manufacturing industries.

2.3 Methodology

To address the objectives identified in Section 1.2.4, Figure 2 was created by expanding on the methodology used by de Meyer [22]. The methodology described in Figure 2 identifies four main sections, namely investigate, design, verification, and implementation. The investigation section comprises research into and adaptions of existing methods to estimate data integrity and are discussed in Section 2.4.

The design section entails the design of a software system that will use the methods identified in the investigation section to estimate the integrity of data and is discussed in Section 2.5. The verification section will focus on the calibration and testing of the software system to ensure that it is functioning as expected and will be discussed in Section 3.2. Lastly, the implementation section will elaborate on how the system is implemented and the results obtained. The implementation section will be discussed in Chapter 3.

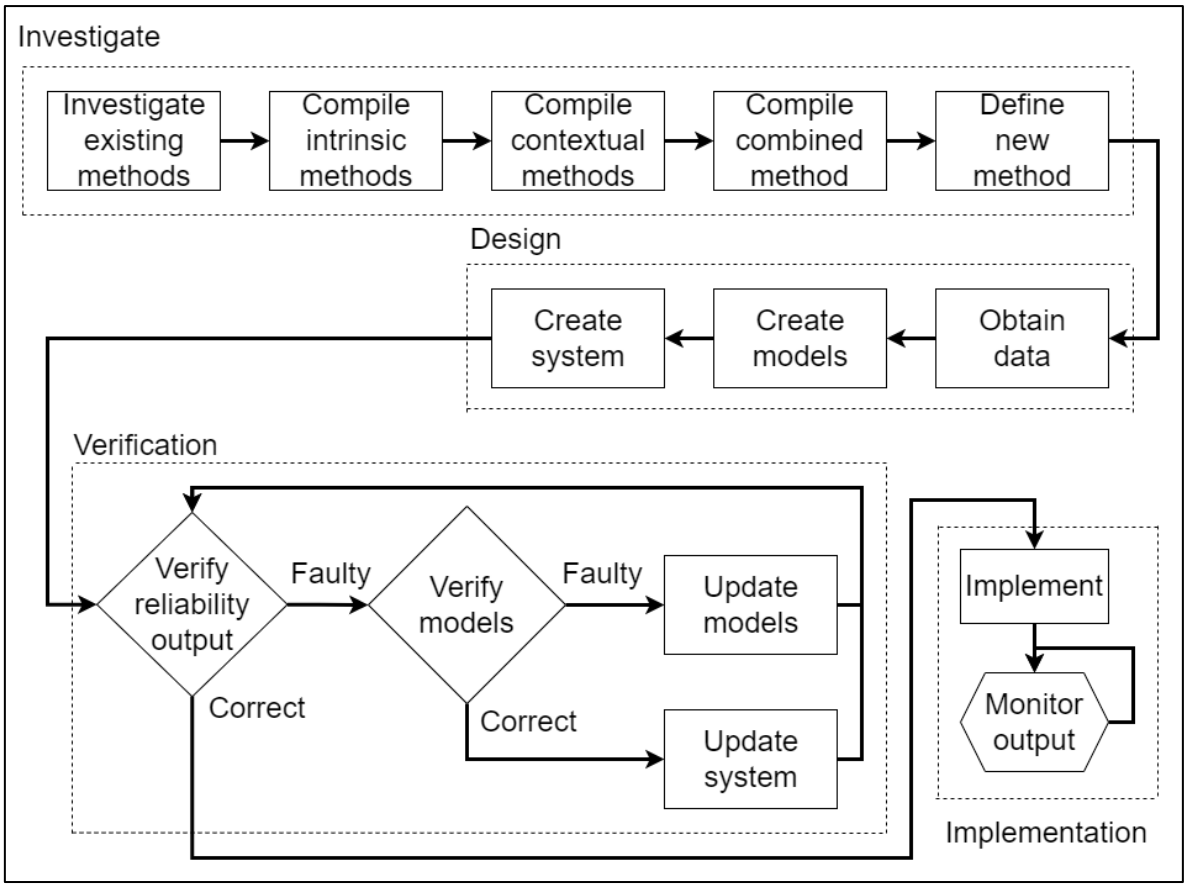

Figure 2: Methodology diagram

2.4 Investigation

This section investigates and adapts existing methods implemented in literature for intrinsic, contextual, and combined data integrity. Throughout the section, equations are introduced with some equations defining variables to represent upper- and lower limits. The limit variables are variably defined, and the value can differ depending on operational conditions, component type and component condition. This is done to abstract hard limits and convey that the values in the equations should be bound by realistic values, allowing the equations to be more generic and applicable to different component types.

2.4.1 Intrinsic integrity

The following data quality dimensions were selected from Section 1.1.4 to calculate the intrinsic integrity:

- accuracy,
- availability,
- completeness, and
- validity.

A presence attribute can be evaluated using Equation 1, which describes how many data points are present for a specified dataset.

$$P = \begin{cases} 1, if\ N_A \neq N_E \\ 0, otherwise \end{cases} \quad \text{Equation 1}$$

where

- P is the presence attribute score,
- N_A is the number of data points available for a range, and
- N_E is the number of data points expected for the range as dictated by the data source and data resolution.

Equation 1 can be used as described by Equation 2 to calculate the completeness dimension,

$$I_C = P \quad \text{Equation 2}$$

where

- I_C is the intrinsic completeness score, and
- P is the presence attribute.

Hanging data are consecutive data points with the exact same value where the values are expected to differ and represented by the hanging attribute described by

Equation 3. Hanging data is not applicable to all data sources as some data is expected to have consecutive repeating values, such as in the case of Boolean data, for example running status data.

$$H = \begin{cases} 1, if\ x_i = x_{i-1} = x_{i-2} = x_{i-3} = x_{i-4} \\ 0, otherwise \end{cases} \quad \text{Equation 3}$$

where

- H is the hanging data attribute,
- x is the value of a data point, and
- i is the position of the data point.

Simply put, if the current value and the previous four values are the same, then the values are hanging. The availability dimension can be calculated by the combination of Equations 1 and 3 as described by Equation 4.

$$I_{Av} = \begin{cases} 1, if\ P = 1\ or\ H = 1 \\ 0, otherwise \end{cases} \quad \text{Equation 4}$$

where

- I_{Av} is the intrinsic availability score,
- P is the presence attribute, and
- H is the hanging data attribute.

To calculate the accuracy dimension, contextual information regarding each data stream is needed with regards to itself. This contextual data includes information such as what the data stream is and what the values can be. This ensures that the data stream can be accurately evaluated in isolation and achieved using Equation 5.

$$I_{Ac} = \begin{cases} 0, if\ L_{lower} \leq V_{current} \leq L_{upper} \\ 1, otherwise \end{cases} \quad \text{Equation 5}$$

where

- I_{Ac} is the accuracy score for a data point,
- L_{lower} is the lower limit of possible values a specific data stream can reach,
- $V_{current}$ is the value of the measured data point at any given point in time, and
- L_{upper} is the upper limit of possible values the data stream can reach.

Similarly to the accuracy dimension, the validity dimension uses contextual information with regards to itself to calculate whether a data point is valid and is achieved by using Equation 6.

$$I_V = \begin{cases} 0, if\ L_{lower} \leq V_{current} \leq L_{upper} \\ 1, otherwise \end{cases}$$ Equation 6

where

- I_V is the validity score for a data point,
- L_{lower} is the lower limit of possible values a specific data stream can reach,
- $V_{current}$ is the value of the measured data point at any given point in time, and
- L_{upper} is the upper limit of possible values the data stream can reach.

Equations 5 and 6 correlate closely to one another. The validity dimension is described as whether data is within acceptable parameters, as illustrated by Equation 6. The accuracy dimension is described as whether the data is correct and reliable. For data with no additional information apart from limits that define which values are possible, Equation 5 represents a best-case estimation of the accuracy dimension. If the accuracy dimension were considered from a contextual perspective, additional information would be available to expand on Equation 5.

Each of the data quality dimensions can achieve a score of zero or one. To calculate the total intrinsic integrity for a data stream, the sum of each dimensional score is compared to the maximum achievable score as described by Equation 7.

$$R_I = \begin{cases} 0, \ \ if\ I_C = 1\ or\ I_{Av} = 1\ or\ I_{Ac} = 1\ or\ I_V = 1 \\ 1, otherwise \end{cases}$$ Equation 7

where

- R_I is the total intrinsic reliability score,
- I_C is the completeness score,
- I_{Av} is the availability score,
- I_{Ac} is the accuracy score, and
- I_V is the validity score.

Ultimately, if any individual dimension is identified as unreliable, the data point is considered unreliable from an intrinsic perspective and receives *zero* value. Conversely, if all dimensions indicate reliability, the data point is seen as reliable from an intrinsic perspective and receives value *one*.

2.4.2 Contextual integrity

The following data quality dimensions were identified in Section 1.1.4 to calculate contextual integrity:

- believability,
- integrity, and
- reliability.

Context should be added to the characteristics to estimate these dimensions, and the relations between the different characteristics should be established to determine the context of each characteristic. As mentioned earlier, each component has four unique characteristics measured and will be used for this study. These characteristics relate to the other characteristics when evaluating them from a contextual point of view, as illustrated in Figure 3.

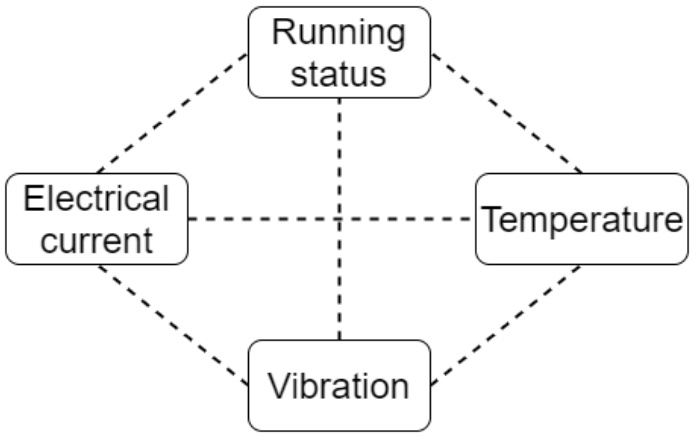

Figure 3: Illustration of relations between measured characteristics

Figure 3 shows how each characteristic is linked to the other three. These relationships may be understood by evaluating each characteristic during both component operation and component out-of-operation states. When a component is running, it draws an electrical current, so it is expected that both the temperature and the vibration will increase to a specific steady value. Conversely, when a component is not running, it cannot draw an electrical current, so it will start to cool down and will vibrate less.

By evaluating components for both in-operation and out-of-operation conditions, relationships between characteristics can be described with more context. This additional information allows the expansion of equations that describe the relationships and aids in minimising situations that are not catered for by the

equations, ultimately increasing the robustness of the equations. Additionally, by considering the operational state of the component, false-positives on three characteristics (electrical current, temperature and vibration) can be avoided by identifying the problem in another characteristic (running status).

The equations presented below aim to simplistically represent the working of components to reduce the expert knowledge needed to understand how each component works. The relations between the characteristics, with varying severity of characteristic value changes, should remain regardless of component operation conditions resulting in the equations remaining applicable for both transient-state and steady-state conditions.

Throughout this section, equations are presented to simplistically represent the relations between the characteristics of components. By combining these equations, components can be simplistically represented and reduce the expert knowledge needed to understand their working, making them applicable to a wider range of components at the cost of reduced accuracy in representing each component's specific operation.

Equations 8 and 9 describe the relationship between the electrical current drawn and the component running status. The value of the component status should be a Boolean value to more precisely indicate when a component is on (1) and when the component is off (0). To this end, any applicable measurement can be used to indicate running status if it is given in Boolean form, for example the on or off signals of a component motor drive or circuit breaker status.

$$E_{current} > 0, when\ R_{status} = 1 \qquad \text{Equation 8}$$

$$E_{current} = 0, when\ R_{status} = 0 \qquad \text{Equation 9}$$

where

- $E_{current}$ is the electrical current drawn, and
- R_{status} is the running status of the component.

Equations 10 and 11 describe the relationship between temperature and the component running status.

$$T_{ambient} \leq T_{current} = t_{on} \times T_{gain} + T_{ambient} \leq T_{limit}, when\ R_{status} = 1$$ Equation 10

$$T_{ambient} \leq T_{current} = t_{off} \times T_{loss} + T_{ambient}, when\ R_{status} = 0$$ Equation 11

where

- $T_{ambient}$ is the ambient temperature at the component,
- $T_{current}$ is the current temperature of the component,
- t_{on} is the duration for which the component has been running,
- T_{gain} is the rate at which the temperature of the component increases while in operation,
- t_{off} is the time the component has been switched off,
- T_{loss} is the rate at which the component cools down,
- T_{limit} is the safety limit of the component where it is switched off, and
- R_{status} is the running status of the component.

The values of T_{gain} and T_{loss} are calculated for each component individually using either historical data or data collected during sensor installation. To estimate T_{gain}, the rate at which the temperature of a component increases from an “off” state to an “on” state should be calculated. Conversely, to estimate T_{loss}, the rate at which the temperature of a component decreases from an “on” state to an “off” state should be calculated.

Equations 12 and 13 describe the relationship between vibration and the component running status.

$$V_{environment} < V_{current} \leq V_{limit}, when\ R_{status} = 1$$ Equation 12

$$V_{environment} \geq V_{current}, when\ R_{status} = 0$$ Equation 13

where

- $V_{environment}$ is the vibration experienced by the component caused by external factors,
- $V_{current}$ is the current vibration measured on the component,
- V_{limit} is the safety limit of the component where it is switched off, and
- R_{status} is the running status of the component.

The value of $V_{environment}$ should be estimated during sensor installation and/or calibration. Prior to connecting the vibration sensor to the component, it should be calibrated to zero. Once connected to the component, the value measured by the

sensor during the “off” state of the component can be used as an estimation of the environmental vibration.

Applying Newton’s law for the conservation of energy to the component, it can be assumed that, when the component draws electrical current, it will convert some of that energy into heat and kinetic energy, resulting in a temperature rise and an increase in vibration. Conversely, when a component stops drawing electrical current, it will eventually cool down and vibrate less. Equations 14 and 15 describe the relationship between temperature and electrical current drawn.

$$T_{ambient} \leq T_{current} = t_{energy\ cons} \times T_{gain} + T_{ambient}, when\ E_{current} > 0 \quad \text{Equation 14}$$

$$T_{ambient} \leq T_{current} = t_{last\ consumed} \times T_{loss} + T_{ambient}, when\ E_{current} = 0 \quad \text{Equation 15}$$

where

- $T_{ambient}$ is the ambient temperature at the component,
- $T_{current}$ is the current temperature of the component,
- $t_{energy\ cons}$ is the duration for which the component has been drawing electrical current,
- T_{gain} is the rate at which the temperature of the component increases when in operation,
- $t_{last\ consumed}$ is the time since the component last drew electrical current,
- T_{loss} is the rate at which the component cools down when not in operation, and
- $E_{current}$ is the electrical current drawn by the component.

Equations 16 and 17 describe the relationship between vibration and electrical current drawn.

$$V_{environment} < V_{current} \leq V_{limit}, when\ E_{current} > 0 \quad \text{Equation 16}$$

$$V_{environment} \geq V_{current}, when\ E_{current} = 0 \quad \text{Equation 17}$$

where

- $V_{environment}$ is the vibration experienced by the component caused by external factors
- $V_{current}$ is the current vibration of the component,
- V_{limit} is the safety limit where the component will be shut off, and
- $E_{current}$ is the electrical current drawn by the component.

Evaluating Equations 16 and 17, a strong correlation between vibration and electrical current drawn is observed. When a component is in operation, i.e., drawing electrical current, it is expected to experience vibration. Conversely, when a component is not in operation, only the vibration in the environment should be experienced. From this relationship, it can be assumed that similar relationships will exist between temperature and electrical current and between temperature and vibration, resulting in Equations 18 and 19.

$$T_{ambient} \leq T_{current} = t_{vib\ exp} \times T_{gain} + T_{ambient}, when\ V_{current} > V_{environment}$$ Equation 18

$$T_{ambient} \leq T_{current} = t_{last\ vib} \times T_{loss} + T_{ambient}, when\ V_{current} = V_{environment}$$ Equation 19

where

- $T_{ambient}$ is the ambient temperature at the component,
- $T_{current}$ is the current temperature of the component,
- $t_{vib\ exp}$ is the duration for which the component has been experiencing vibration,
- T_{gain} is the rate at which the temperature of the component increases when in operation,
- $t_{last\ vib}$ is the time since the component last experienced vibration,
- T_{loss} is the rate at which the component cools down when not in operation,
- $V_{current}$ is the vibration experienced by the component, and
- $V_{environment}$ is the vibration in the environment.

Using Equations 8 – 19, it can be seen that characteristics influence one another, which is essential to consider when evaluating the reliability of the data from a contextual viewpoint.

Using the relations described in Equations 8 – 19, each data stream can be compared to the data of the other three characteristics. A truth table can be compiled, as illustrated in Table 7.

Table 7: Truth table used for contextual data reliability calculations

		Characteristic evaluated			
		Running	**Current**	**Temperature**	**Vibration**
Characteristic Perspective	**Running**	-	True/False	True/False	True/False
	Current	True/False	-	True/False	True/False
	Temperature	True/False	True/False	-	True/False
	Vibration	True/False	True/False	True/False	-

The truth table compares the reliability of each characteristic from the view of a different characteristic. From this table, the reliability of the data point is calculated using Equation 20:

$$R_c = \frac{S_{received}}{S_{given}} \geq 0.6, where\ S_{received} \geq 2\ and\ S_{given} \geq 2\ and\ S_{given} \neq 0$$ Equation 20

where

- R_c is the contextual reliability of the data point,
- $S_{received}$ is the number of high-reliability scores (T) for the characteristic awarded by the other three characteristics, listed in the characteristic column, and
- S_{given} is the number of high-reliability scores (T) given to the other three characteristics by this characteristic, listed in the characteristic row.

If $R_c \geq 0.6$, then the value will be rounded up to one and is classified as reliable. Otherwise, it will be rounded down to zero and be seen as unreliable. For a data point to be seen as reliable, it should consider other data points reliable and be considered reliable by other data points. Considering the reliability of a data point from both perspectives, only data points that fit into the component's context should be classified as reliable.

To this end, the ratio between the scores received and given should be greater than 60%. This percentage was chosen to eliminate the majority of:

- low-performing data points from both perspectives, and
- high-performing data points from a single perspective.

Additionally, the given and received scores should be at least two to reduce the number of falsely classified high-reliability data points. The data points that are seen as reliable are illustrated in Table 8, where each cell indicates the ratio of scores given and received as described in Equation 20.

Table 8: Contextual integrity acceptable score ratio

		Score received			
		0	**1**	**2**	**3**
Score given	**0**	×	×	×	×
	1	×	×	×	×
	2	×	×	✓	✓
	3	×	×	✓	✓

2.4.3 Combined integrity

The combined reliability of each data point is the sum of its intrinsic and contextual reliability scores, Equations 7 and 20, respectively, as illustrated by Equation 21:

$$0 \leq R_T = \frac{(R_I + R_C)}{2} \leq 1 \qquad \text{Equation 21}$$

where

- R_T is the combined reliability of a data point,
- R_I is the intrinsic reliability, and
- R_C is the contextual reliability.

For a data point to be considered reliable, R_T should have a value of one. If the value is less than one, it will be rounded to zero and considered unreliable.

2.5 Design

A software system was designed to verify the method discussed in the previous section.

2.5.1 Data collection

Condition-based maintenance data is recorded at the mine by means of a data collection process. A simplified data collection process is illustrated in Figure 4.

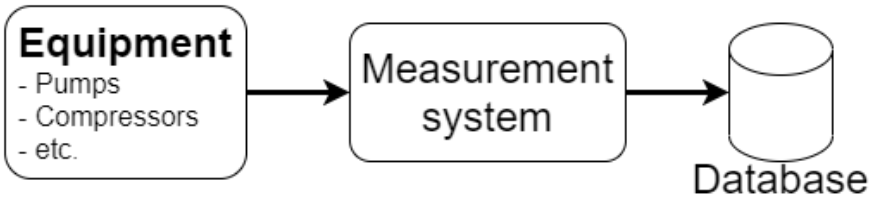

Figure 4: Diagram illustrating the data collection process for condition monitoring

Each component in the organisational structure is monitored as part of a monitoring system, such as a supervisory control and data acquisition (SCADA) system. This monitoring system captures the characteristic values of the component and sends them to a database. Once the data is in the database, it can be used by authorised systems and personnel.

2.5.2 Data format

Measurements for the different characteristics are saved to the database in a set format as determined by the data collection process. This study will assume that the measured characteristic data will be saved into a NoSQL[6] database using a tag-value document structure. A tag document represents the data stream for a measured characteristic for a specific component, e.g., a data stream for measuring the temperature for Component A on Site B can have a tag named “SiteB_ComponentA_Temp”.

A value document represents the measured data point for a specific characteristic for a specific component at a certain point in time. Each tag document can have multiple value documents; however, a value document can only be linked to one tag document. Both tag and value documents make use of the BSON[7] format. The tag document structure and the value document structure are discussed in Table 9 and Table 10, respectively.

[6] Non-relational database
[7] Binary JSON

Table 9: Breakdown of tag document attributes

Attribute	Description	Data type
_id	Unique identifier for the entry.	ObjectId
Name	User-friendly name.	String
Interval	Describes the data resolution for this tag.	JSON[8] object
Unit	Unit in which the measurements are taken.	String
Source	Source from which this tag receives data.	String

Table 10: Breakdown of value document attributes

Attribute	Description	Data type
_id	Unique identifier for the entry.	ObjectId
Timestamp	Timestamp for when this measurement was taken.	DateTime
Value	Numeric value captured by the monitoring system.	Double
TagID	Tag ID to which this value belongs.	ObjectId

Depending on the monitoring system and the installed sensors on a mine, different measurements can be taken and monitored. For this study, the focus will be placed on the four main characteristics identified in Section 1.1.3. Each characteristic will have its own data stream, namely a tag document and multiple value documents.

For the system to work with data streams in context on a component level, the component configuration must be accessible. An SQL[9] database will be used as this is configuration data and represents the relations between the different characteristics.

Six tables will be required to store the data. Three tables are required to store the company, site, and system information. There are tables to link the tags in the NoSQL database to the SQL database, configure components, and link tags as characteristics to components. A table layout is illustrated in Figure 5 and discussed in Table 11.

[8] JSON = JavaScript Object Notation
[9] Relational database

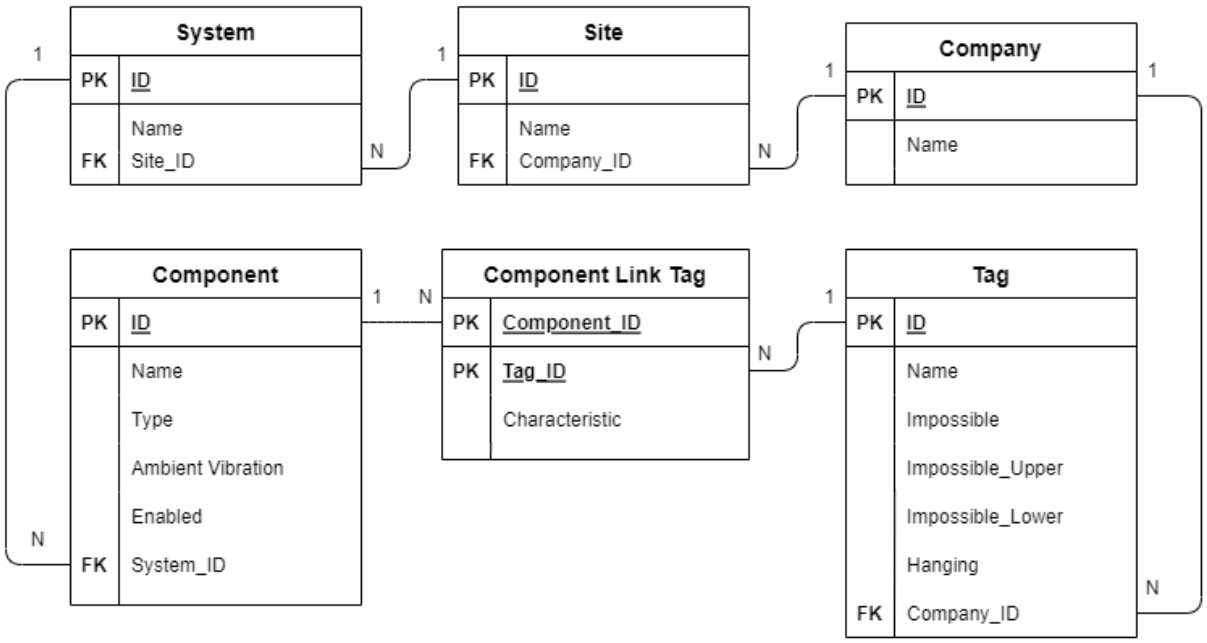

Figure 5: SQL database design

Table 11: Breakdown of SQL database design components

Table	Description
Company	Describes the company to which all the information belongs.
Site	Describes the sites linked to a specific company.
System	Describes the systems linked to a specific site.
Component	Represents the component to be analysed.
Tag	Adds additional context to the values measured for a specific characteristic.
Component Link Tag	Links data streams as characteristics to components.

2.5.3 Models

A simplistic model will be used to generically represent components to decouple the system from only working with specific component types. This ensures that the system is not restricted to simply working with specific components if the operation of the component can be described with Equations 8 – 19 in Section 2.4.2. The layout for the generic component model used by the system is discussed in Table 12 and mimics the layout of the relational database tables.

Table 12: Software system model layout

Attribute	Description	Data type
ID	Unique identifier for the component.	Unsigned integer
Name	Name of the component.	String
Type	Component type.	Enum
Ambient Vibration	Ambient vibration measured for the component.	Double
Running	Tag linked as a running-status characteristic data stream.	Characteristic class
Current	Tag linked as an electrical current characteristic data stream.	Characteristic class
Temperature	Tag linked as temperature characteristic data stream.	Characteristic class
Vibration	Tag linked as vibration characteristic data stream.	Characteristic class
SystemID	Reference to specific system.	Unsigned integer
Enabled	Indicator of whether the component should be analysed.	Boolean

Each of the linked characteristic data streams are represented by a custom class named “Characteristic class”, which is represented by the “Tag” table in Figure 5 and described in Table 13. Using this generic component model, the system can apply the method designed in Section 2.4 provided the model is calibrated correctly.

Table 13: Software system model custom class layout

Attribute	Description	Data type
ID	Unique identifier for the tag and reference to the tag in the non-relational database.	String
Hanging	Indicator of whether the data stream should be checked for hanging values.	Boolean
Impossible	Indicator of whether the data stream should be checked for impossible values.	Boolean
Impossible_Upper	Upper limit that the data stream may never exceed.	Double
Impossible_Lower	Lower limit that the data stream may never exceed.	Double
Company_ID	Reference to the company.	Unsigned integer

2.5.4 System

An overview of the proposed software system is displayed in Figure 6. The system will consist of three main components, each serving a different purpose, as discussed in Table 14. The SQL and NoSQL component designs are discussed in Section 2.5.2 above.

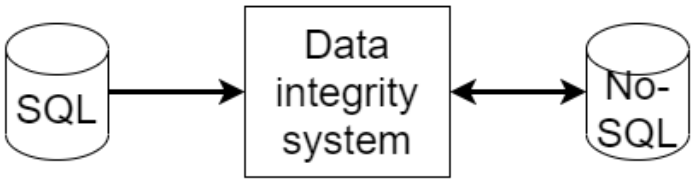

Figure 6: Overview of the proposed software system

Table 14: The main components in the proposed software system

Component	Type	Purpose
SQL	Relational database	Stores configuration data
Data Integrity System	Application	Calculates data integrity
NoSQL	Non-relational database	Store characteristic data and integrity analysis results

The data integrity system uses the method proposed in Section 2.4 to determine the data integrity, shown schematically in Figure 7. The system comprises three steps: calculating intrinsic integrity, contextual integrity, and combined integrity.

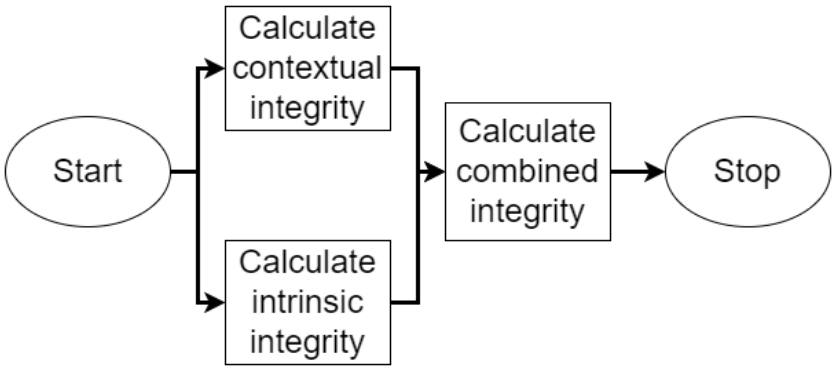

Figure 7: Diagram illustrating the overview of the data integrity system

Calculating intrinsic integrity

A sequence of steps was developed to calculate the data integrity for an intrinsic approach using the methods identified in Section 2.4.1. The steps are shown schematically in Figure 8 and described in Table 15.

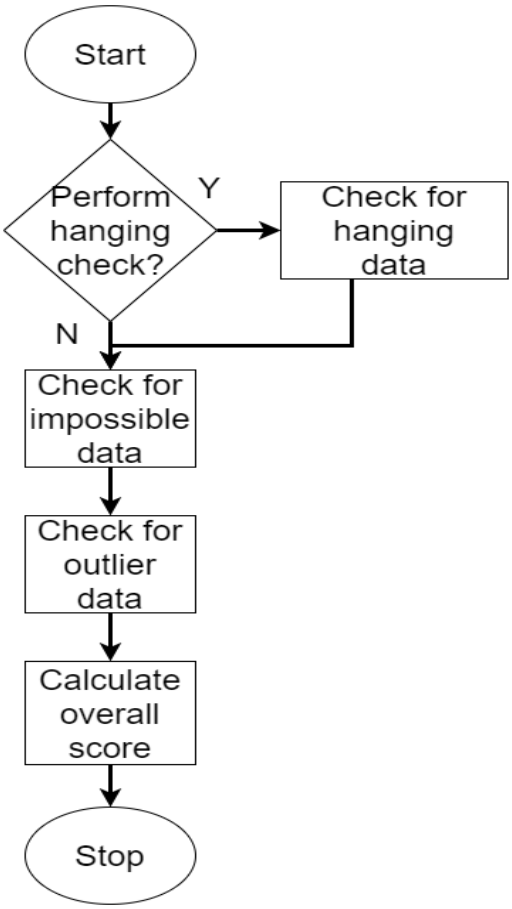

Figure 8: Diagram of system flow for intrinsic integrity calculation

Table 15: Description of the intrinsic data quality steps

Stage	Description	Equations used	Dimensions
Perform hanging check	Indicates whether the system should evaluate a data stream for hanging data		-
Check for hanging data	Check whether a data stream has repeating values	2	Accuracy, Validity
Check for impossible data	Check whether a data stream exceeds certain limits	4, 5	Accuracy, Validity
Check for outlier data	Check whether data points conform to the norm	4, 5	Accuracy, Validity
Calculate overall score	Evaluates the number of data points and calculates the overall score	1, 3	Completeness, Accessibility

Finally, the results from each calculation are used, and a final score is calculated using Equation 7. Whilst the system calculates the intrinsic integrity, it also performs calculations to determine contextual integrity.

Calculating contextual integrity

The process followed to calculate the contextual integrity is illustrated in Figure 9 and makes use of the methods identified in Section 2.4.2. Figure 9 illustrates how the system calculates the contextual integrity for single-source data. Firstly, the system retrieves the component information from the relational database, as discussed in Figure 5.

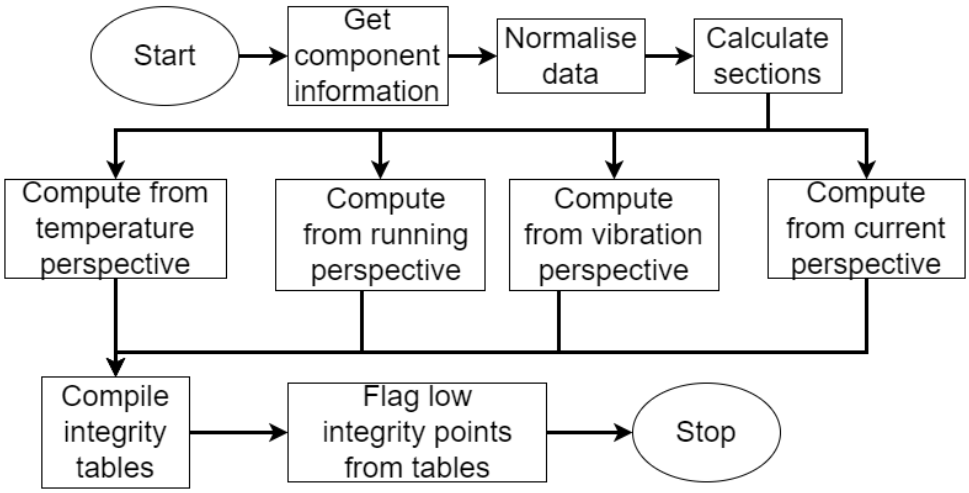

Figure 9: System flow for calculating contextual integrity

Next, the system uses the component information to retrieve each characteristic data stream linked and normalises the data. By normalising the data, the system can divide the data streams into sections with increasing, decreasing, or flat gradients, as illustrated in Figure 10.

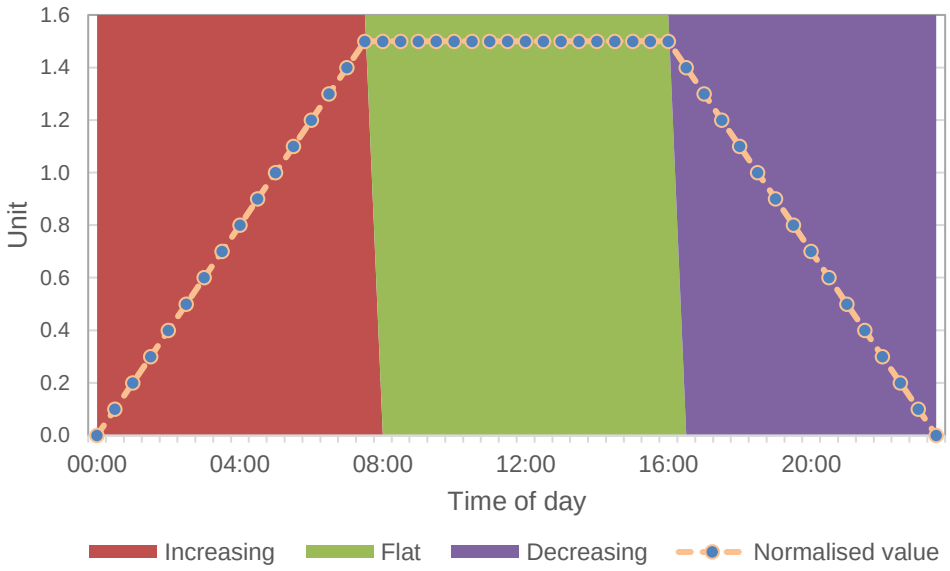

Figure 10: Illustration of normalised value sections

Each section is then evaluated from each characteristic perspective using Equations 8 – 19 to check for contradictions to relations described in the equations. Once all the sections have been analysed, the results are stored in a truth table as described by Table 7. Using this table, low-integrity data points are identified and flagged using Equation 20.

Each of the four characteristics has a different relation to the other characteristics; thus, calculating the integrity of a data point from each characteristic perspective needs a different approach for each data point – a characteristic approach. Each characteristic perspective is calculated in the same general way, namely calculating the sections, then evaluating them one at a time.

Figure 11 illustrates how characteristic data streams are analysed and divided into different data sections, as discussed in Table 16.

Table 16: Characteristic section analysis

Section	Description
Flat	There is almost negligible variation between data points and the group of data points can be represented by a flat line.
Positive/Increasing	There is a positive variation between data points and the group of data points can be represented by a line with a positive/increasing gradient.
Negative/Decreasing	There is a negative variation between data points and the group of data points can be represented by a line with a negative/decreasing gradient.

After dividing the data into sections, each section is evaluated. Due to the differences between characteristics, each section for each characteristic is evaluated differently. Each characteristic section evaluation is described in more detail, with flow diagrams, in its' corresponding section.

Thus, for each of the characteristics, the generic flow diagram illustrated in Figure 11 is used in combination with the characteristic-specific evaluation flow diagrams, linked by letter indicator, as discussed in the subsequent sections. For example, when analysing running status data, Figure 11 is used for the evaluation process, Figure 12 is used to evaluate flat sections (A), Figure 13 is used to evaluate positive sections (B) and Figure 14 is used to evaluate negative sections (C).

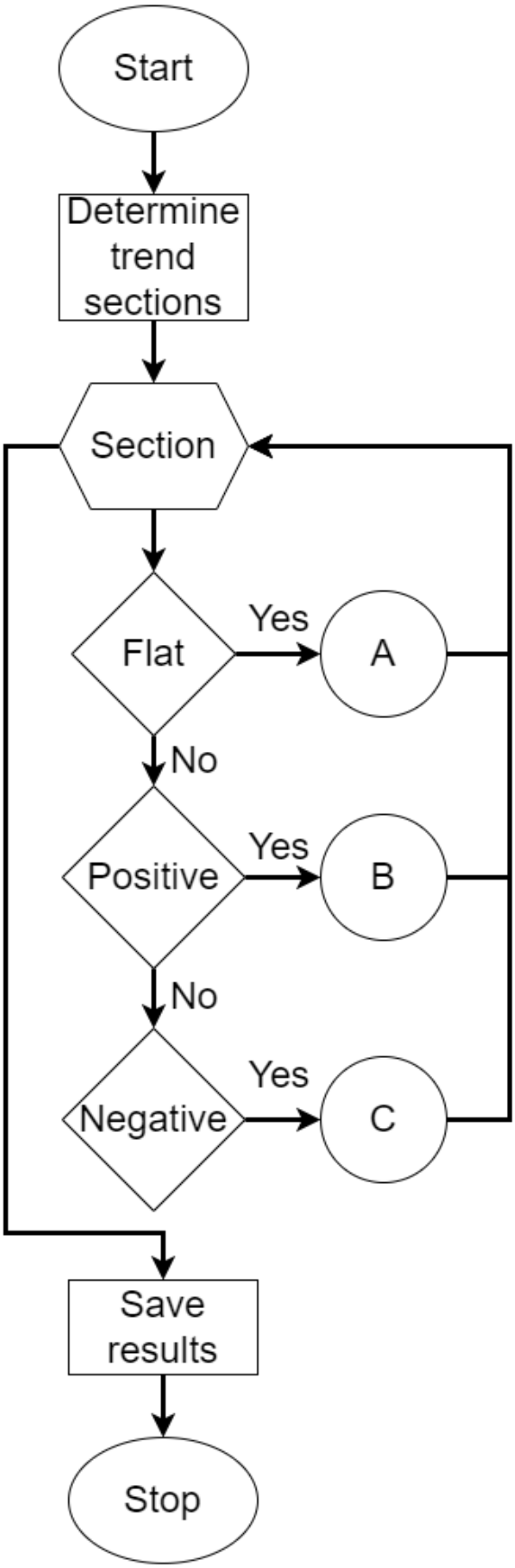

Figure 11: Overview of characteristic contextual integrity calculations

Running status

The running status indicates whether a component is running or not. From the running status perspective, there is one main section, namely the flat section, as switching a component from the *on* status to the *off* status and vice-versa is nearly instantaneous, creating only short-lived positive or negative sections. The behaviour of each characteristic from the perspective of the running status was derived from Equations 8-13. Figure 12 illustrates how characteristics are flagged during flat sections of the running status, with Table 17 describing the expected behaviour of each characteristic to be classified as high integrity. The classification of characteristics during positive and negative sections are illustrated in Figure 13 and Figure 14, respectively. The expected behaviour of characteristics to be classified as high integrity are described in Table 18.

Table 17: Running status perspective flat section evaluation

Running status	Characteristic	Expected behaviour	Equation
On	Electrical current	Positive and greater than zero.	8
	Temperature	Positive and greater than ambient temperature.	10
	Vibration	Positive and greater than environmental vibration.	12
Off	Electrical current	Zero.	9
	Temperature	Positive and decreasing towards ambient temperature.	11
	Vibration	Positive and decreasing towards environmental vibration.	13

Table 18: Running status perspective positive and negative section evaluation

Section	Characteristic	Expected behaviour	Equation
Positive	Electrical current	Positive and increasing.	8, 9
	Temperature	Positive and increasing.	10, 11
	Vibration	Positive and increasing.	12, 13
Negative	Electrical current	Positive and decreasing.	8,9
	Temperature	Positive and decreasing.	10,11
	Vibration	Positive and decreasing.	12,13

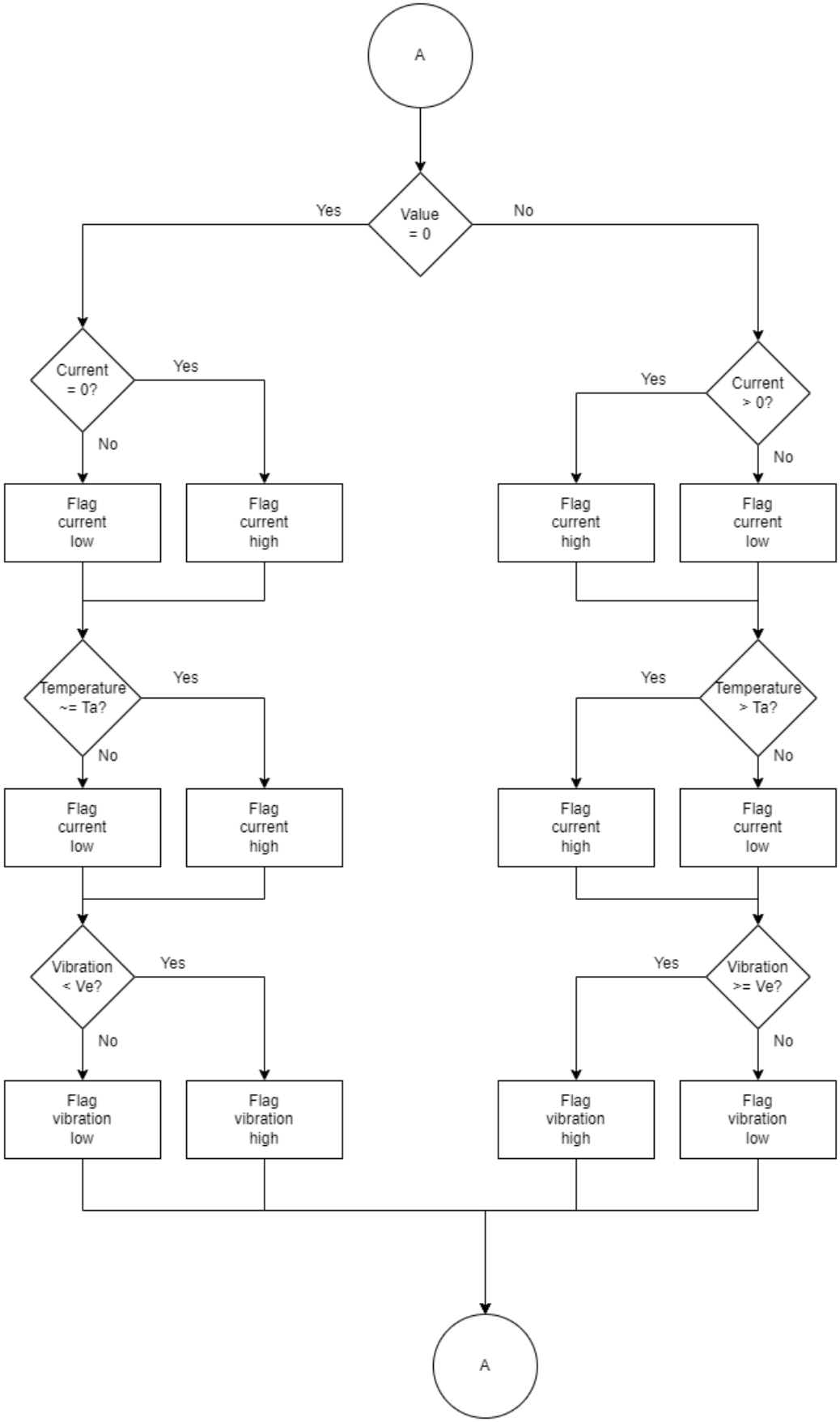

Figure 12: Contextual integrity flow of flat sections for the running status

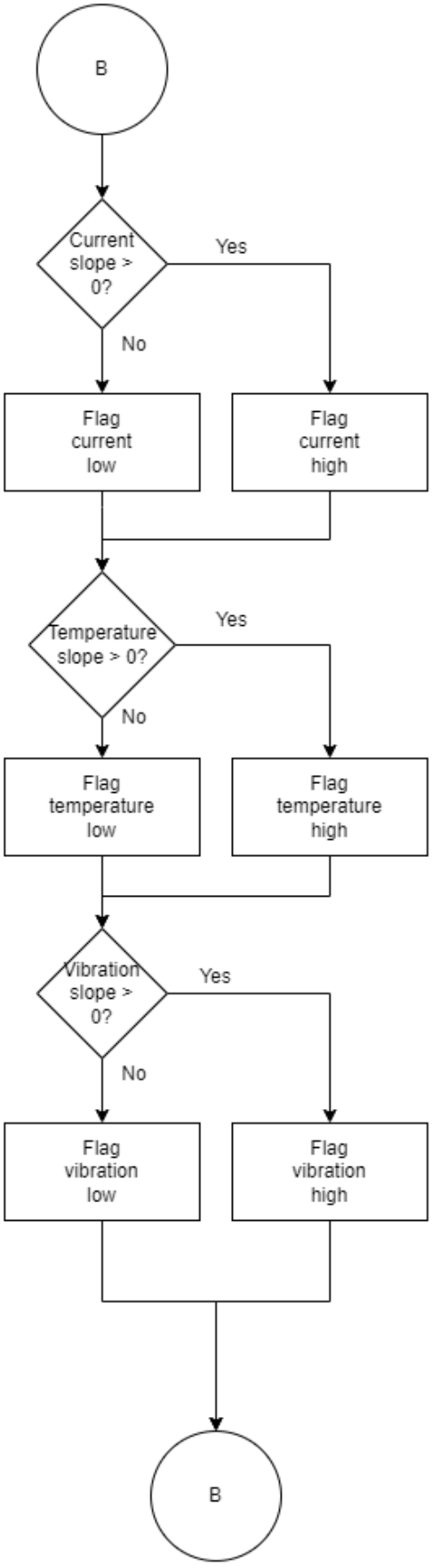

Figure 13: Contextual integrity flow of positive sections for the running status

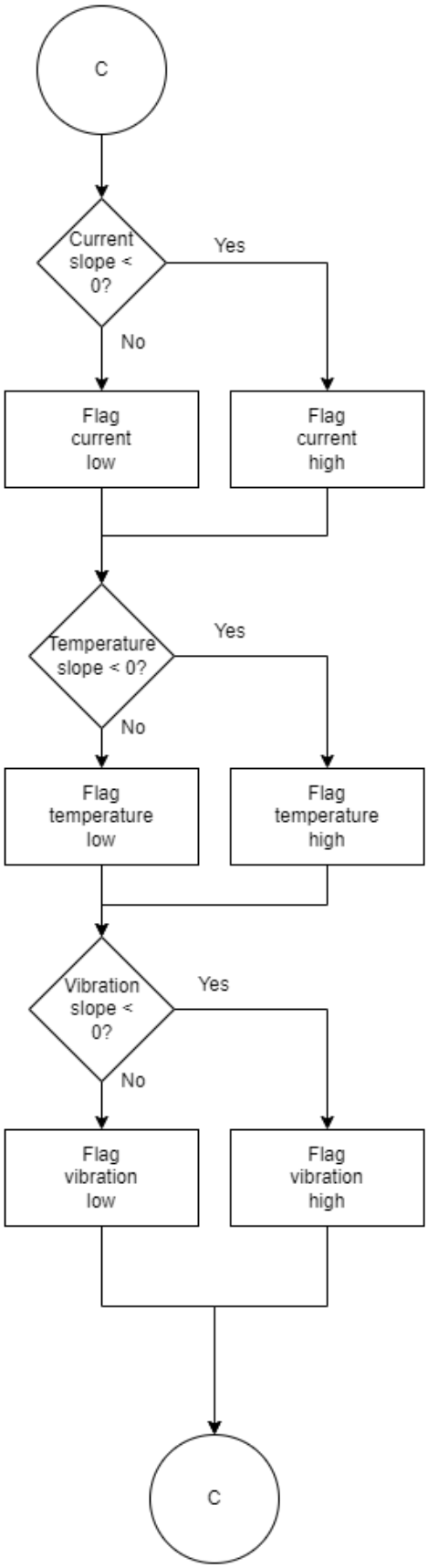

Figure 14: Contextual integrity flow of negative sections for the running status

Electrical current

Electrical current can indicate whether a component is in use or not, indicated by a positive, non-zero, or zero, value respectively. During the operation of a component, the electrical current drawn can differ depending on various factors, such as load. Thus, it is expected that the electrical current characteristic will be comprised of various flat, positive and negative sections. The behaviour of characteristics from the perspective of the electrical current is derived from Equations 8-9 for the running status, Equations 14-15 for temperature and Equations 16-17 for vibration. From the electrical current perspective, flat section calculations are illustrated in Figure 15, positive section calculations in Figure 16 and negative section calculations in Figure 17. Stages from the figures above are described in Table 19, detailing how characteristics are classified as *high integrity*.

Table 19: Electrical current perspective characteristic evaluations

Section	Characteristic	Expected behaviour	Equation
Flat	Running status	Zero if electrical current is zero and one if electrical current is greater than zero.	8,9
	Temperature	Near static value	14,15
	Vibration	Near static value	16,17
Positive	Running status	One if the electrical current slope is below a defined threshold and electrical current un-normalised value does not start at zero. Otherwise, it increases from zero to one.	8,9
	Temperature	Increasing if electrical current is increasing by more than a defined threshold, stable otherwise.	14,15
	Vibration	Increasing.	16,17
Negative	Running status	Value unity the electrical current slope is above a defined threshold, and electrical current does not go to zero. Otherwise, it should go from one to zero.	8,9
	Temperature	Decreasing if electrical current decreases faster than a defined threshold, stable otherwise.	14,15
	Vibration	Decreasing.	16,17

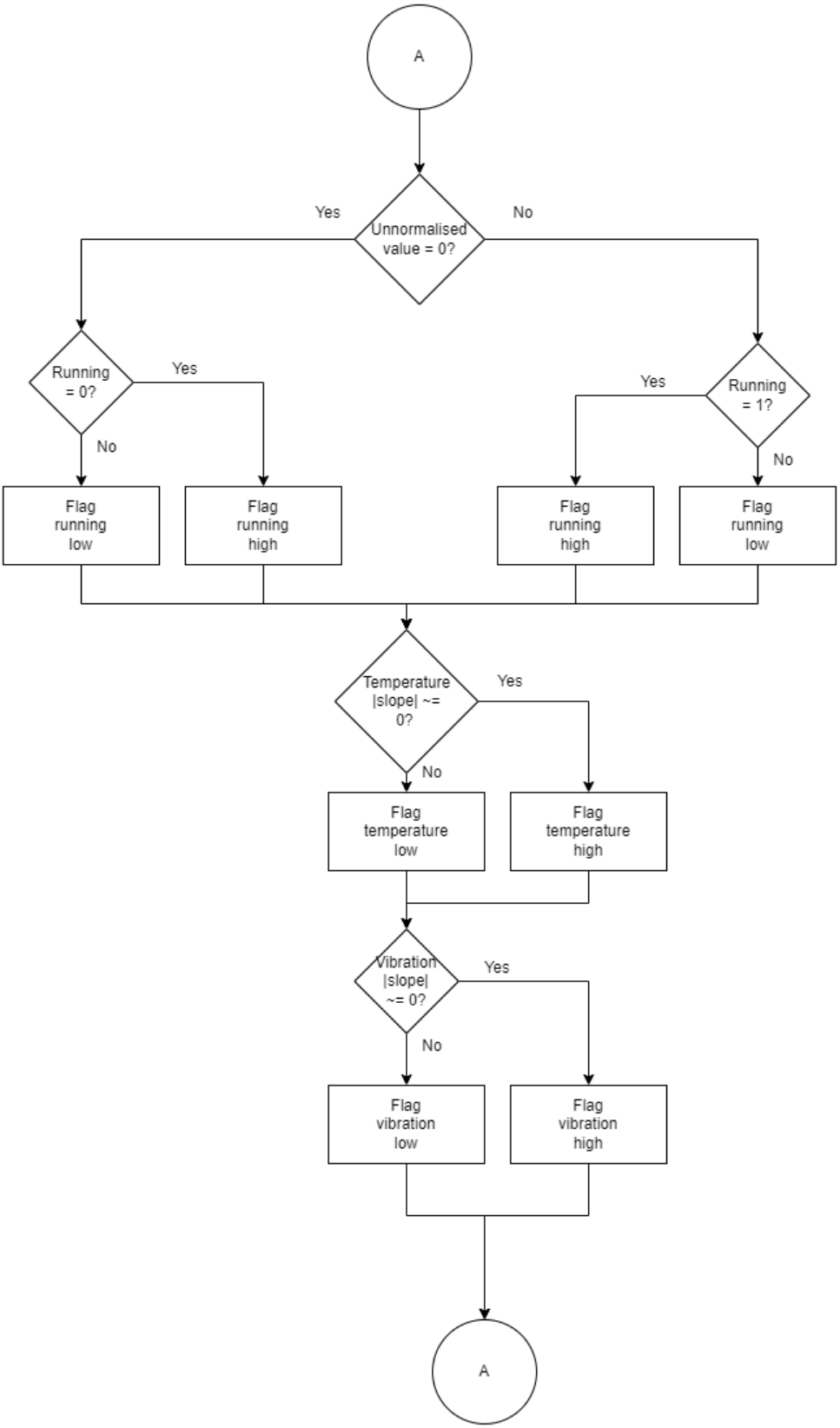

Figure 15: Contextual integrity flow of flat sections for electrical current

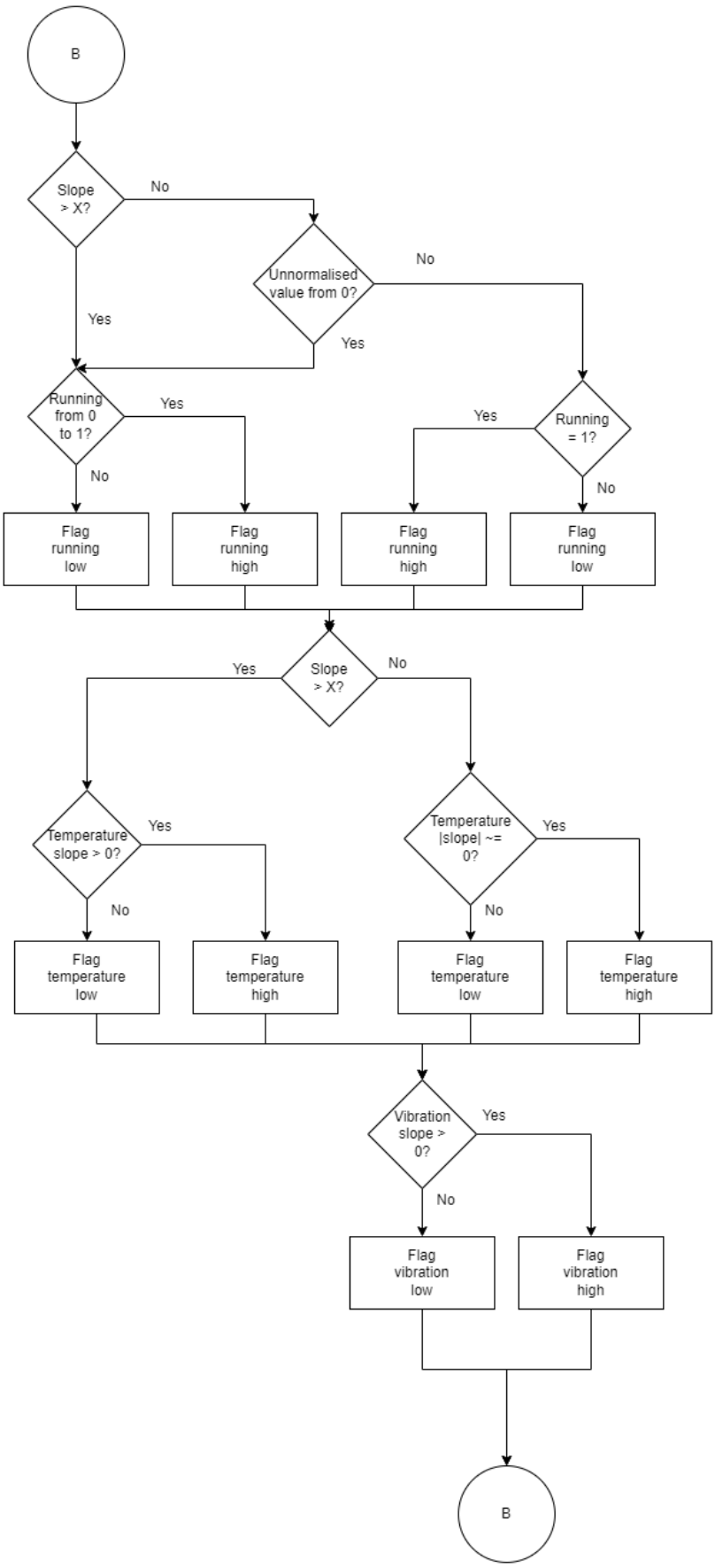

Figure 16: Contextual integrity flow of positive sections for electrical current

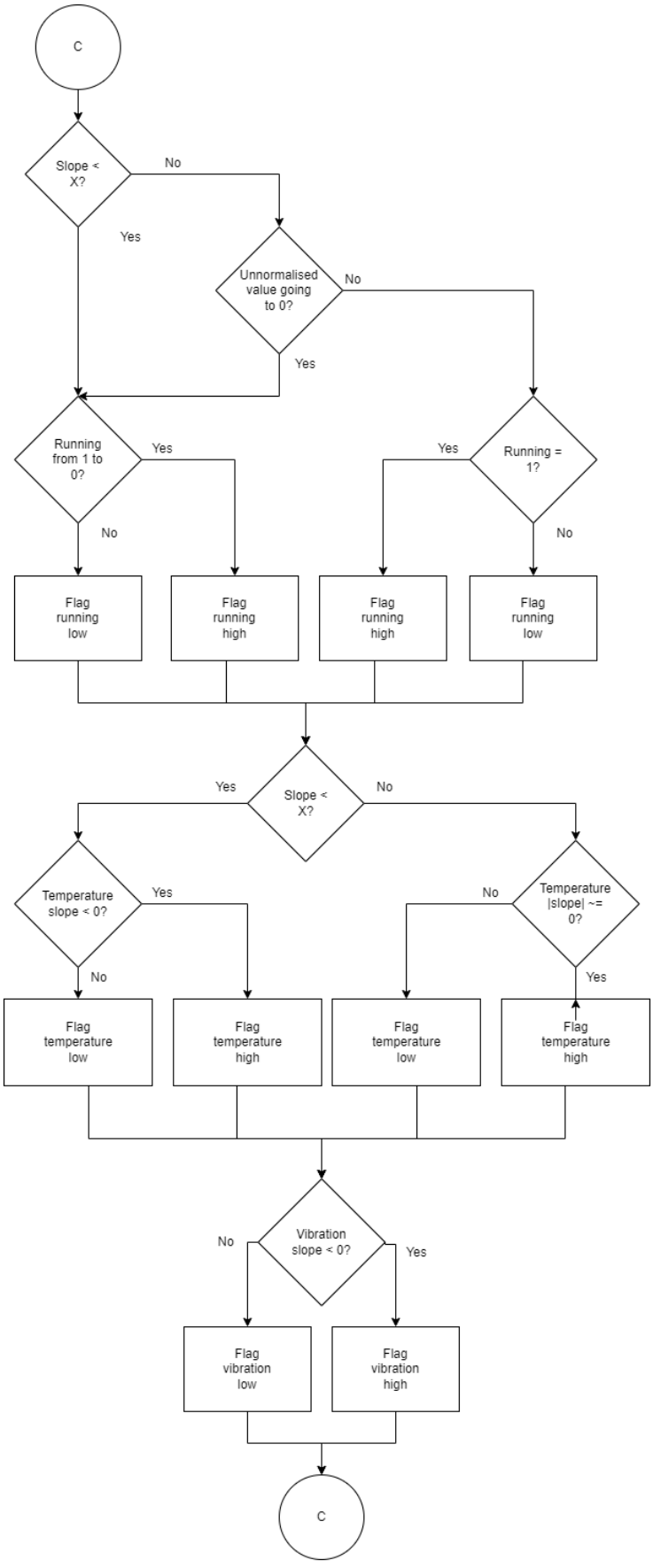

Figure 17: Contextual integrity flow of negative sections for electrical current

Temperature

Temperature is the most volatile characteristic and fluctuates regularly, resulting in an abundance of the three section types. Like the electrical current, the temperature is influenced by internal and external factors. However, unlike electrical current, the temperature does not go down to zero when a component is switched off.

To simplify the identification process from the temperature perspective, a threshold temperature is assumed to act as an indicator of when the component is switched off. This threshold temperature can vary depending on the component and the conditions in which it operates. Normally, an ambient temperature of 25˚C would be used, however, this study focusses on the mining industry in which ambient temperatures are generally higher. Thus, for the purpose of this study, a threshold temperature of 30˚C is assumed.

Categorising characteristics from the temperature perspective uses Equations 10 and 11 for running status, Equations 14 and 15 for electrical current and Equations 18 and 19 for vibration. Figure 18 illustrates how flat sections are evaluated, and reliable characteristic behaviour is discussed in Table 20 for this section. Positive and negative section evaluations are illustrated in Figure 19 and Figure 20, respectively. The expected reliable behaviour for characteristics in these sections is discussed in Table 21.

Table 20: Temperature perspective flat section characteristic evaluation

Temperature	Characteristic	Expected behaviour	Equation
Above 30˚C	Running status	One.	10,11
	Electrical current	Normalised value is one.	14,15
	Vibration	Greater than environmental vibration.	18,19
Below 30˚C	Running status	Zero.	10,11
	Electrical current	Zero.	14,15
	Vibration	Less than environmental vibration	18,19

Table 21: Temperature perspective characteristic evaluation for positive and negative sections

Section	Characteristic	Expected behaviour	Equation
Positive	Running status	One or the previous value had to be one.	10,11
	Electrical current	Positive and increasing.	14,15
	Vibration	Positive and increasing.	18,19
Negative	Running status	Value unity if the temperature decreases with less than a defined threshold and the un-normalised temperature is above a defined threshold, zero otherwise.	10,11
	Electrical current	Positive and decreasing.	14,15
	Vibration	Positive and decreasing.	18,19

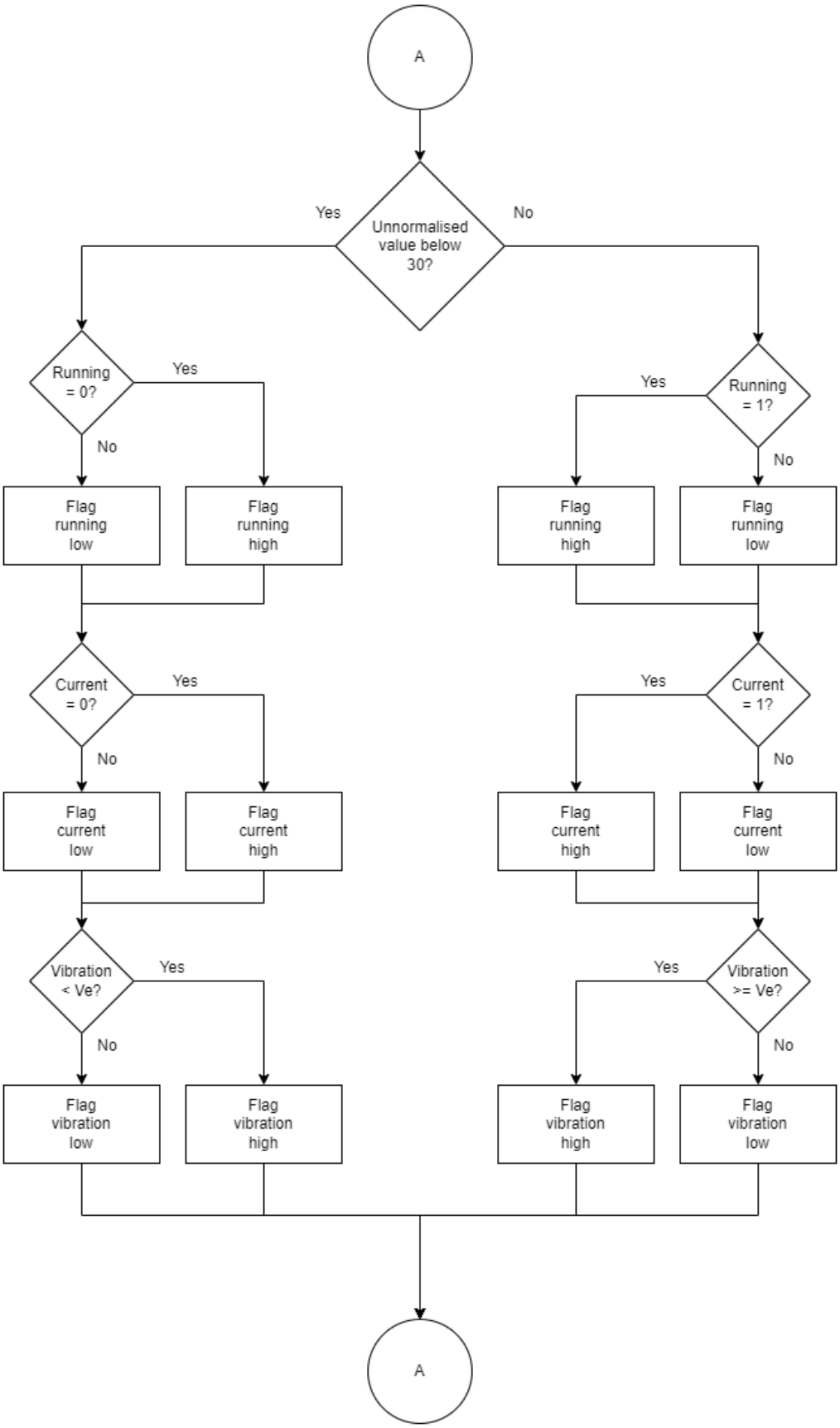

Figure 18: Contextual integrity flow of flat sections for temperature

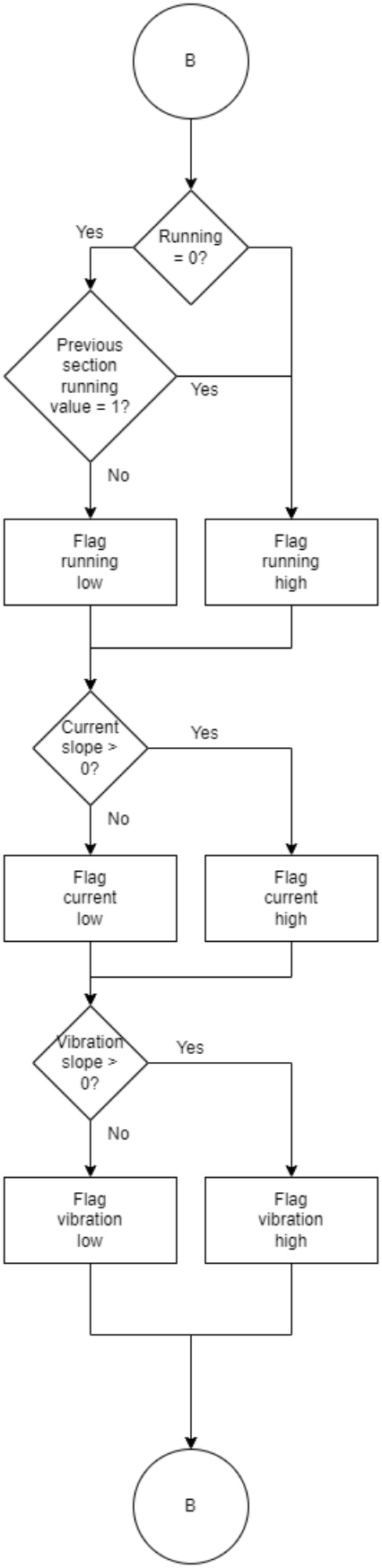

Figure 19: Contextual integrity flow of positive sections for temperature

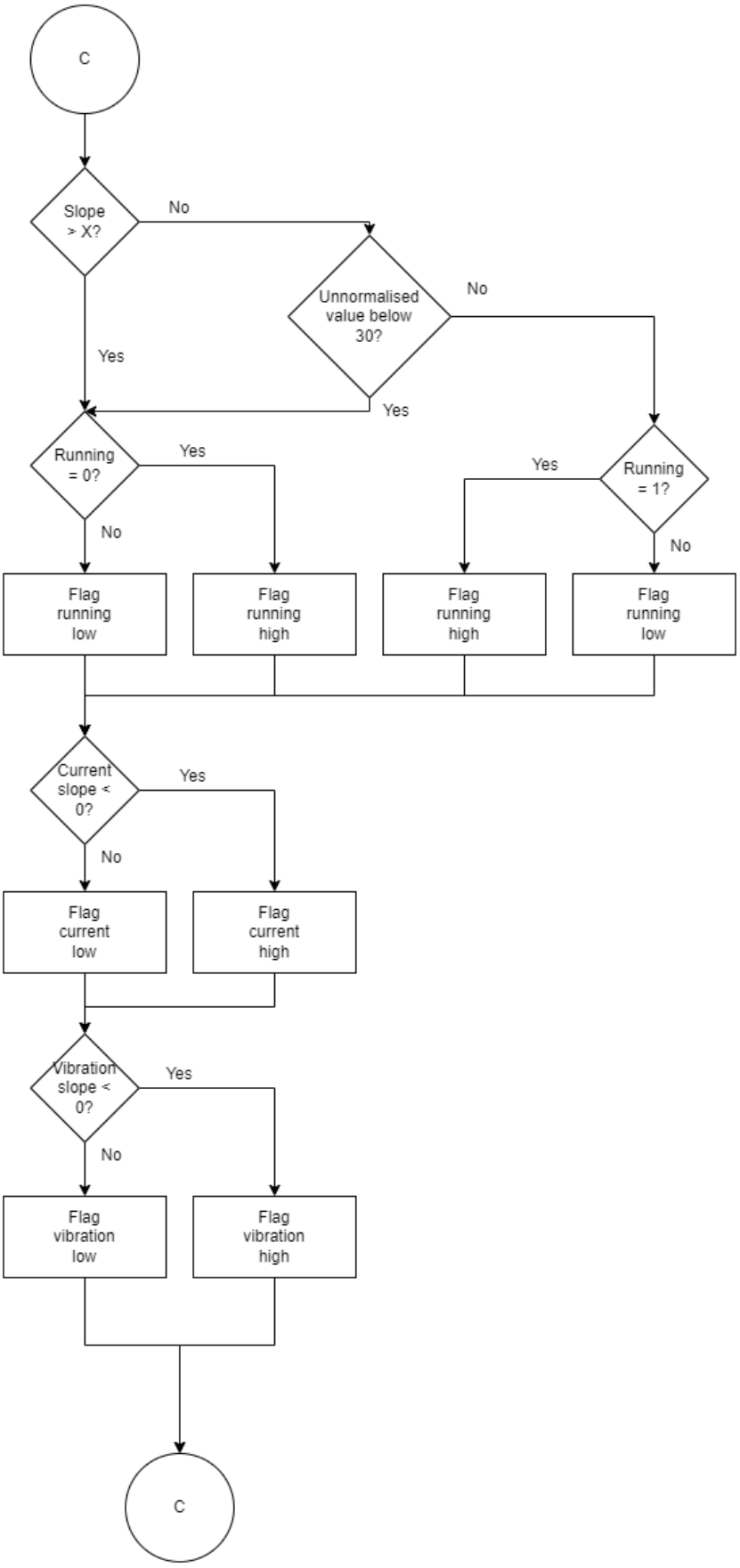

Figure 20: Contextual integrity flow of negative sections for temperature

Vibration

Vibration can be used as an indicator of the running status of a component as a component in the *on* state will produce vibrational energy. However, due to the influence of other components in the vicinity, the measurement equipment can read the environmental vibration. This is taken into account and used to adjust the baseline measurements for each component to ensure accurate analysis of the different characteristics. Equations 12 and 13 were used to evaluate the reliability from the vibration perspective for the running status characteristic. Equations 16 and 17 were used for electrical current, and Equations 18 and 19 were used for temperature. The evaluation processes for flat sections are illustrated in Figure 21, positive sections in Figure 22 and negative sections in Figure 23. Each characteristic's expected high integrity behaviour for each section is described in Table 22.

Table 22: Vibration perspective characteristic evaluation

Section	Characteristic	Expected behaviour	Equation
Flat	Running status	One if un-normalised vibration is greater than environmental vibration, zero otherwise.	12,13
	Electrical current	Near static value	16,17
	Temperature	Near static value	18,19
Positive	Running status	One if the vibration slope is below a defined threshold and the un-normalised value does not start at zero. Increasing from zero to one otherwise.	12,13
	Electrical current	Increasing.	16,17
	Temperature	If the vibration slope is below the threshold, the temperature should be stable. Otherwise, the temperature should be increasing.	18,19
Negative	Running status	One if vibration slope is above a defined threshold and vibration does not go to environmental vibration. Otherwise, it should go from one to zero.	12,13
	Electrical current	Decreasing.	16,17
	Temperature	If the vibration slope is above the threshold, the temperature should be stable. Otherwise, the temperature should be decreasing.	18,19

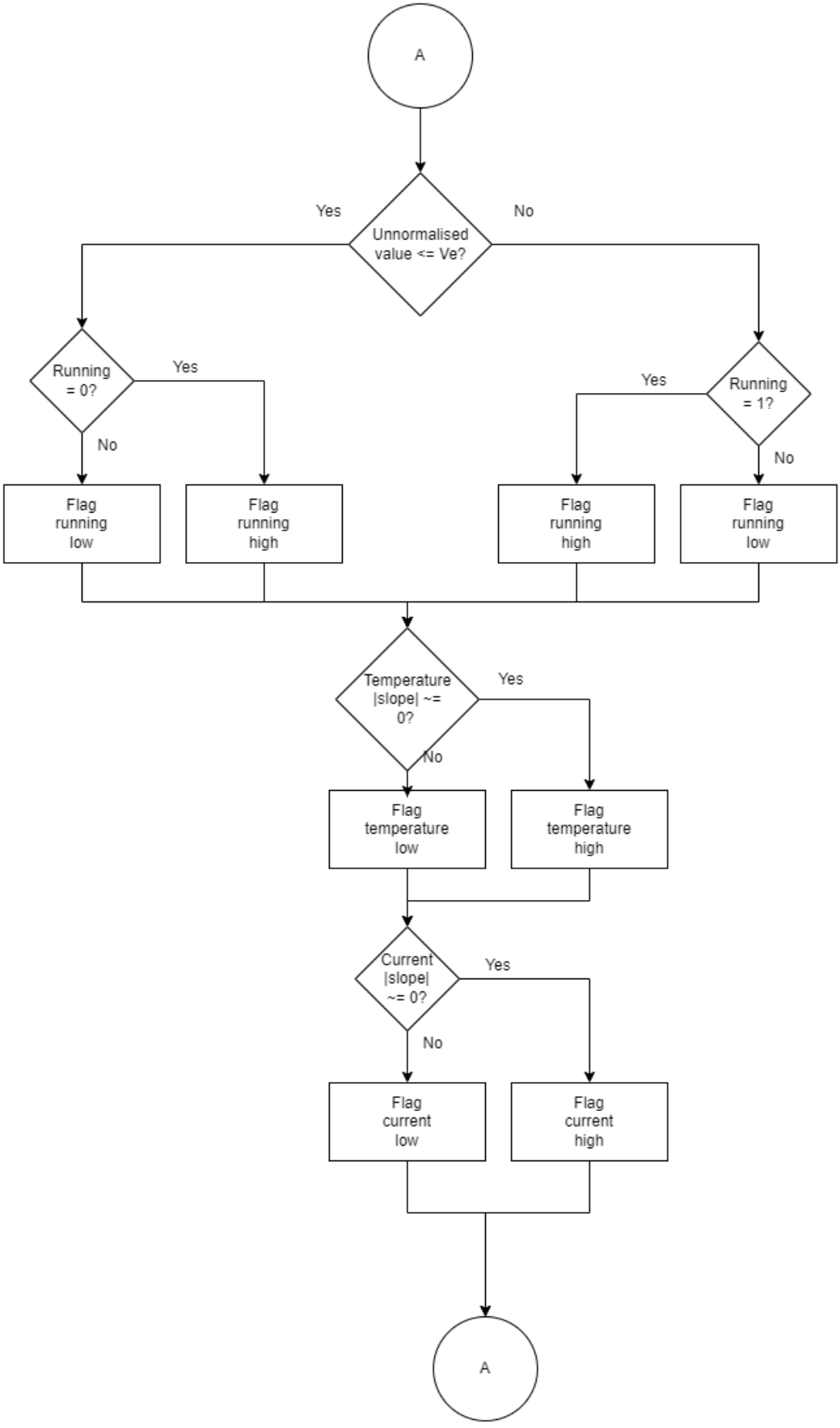

Figure 21: Contextual integrity flow of flat sections for vibration

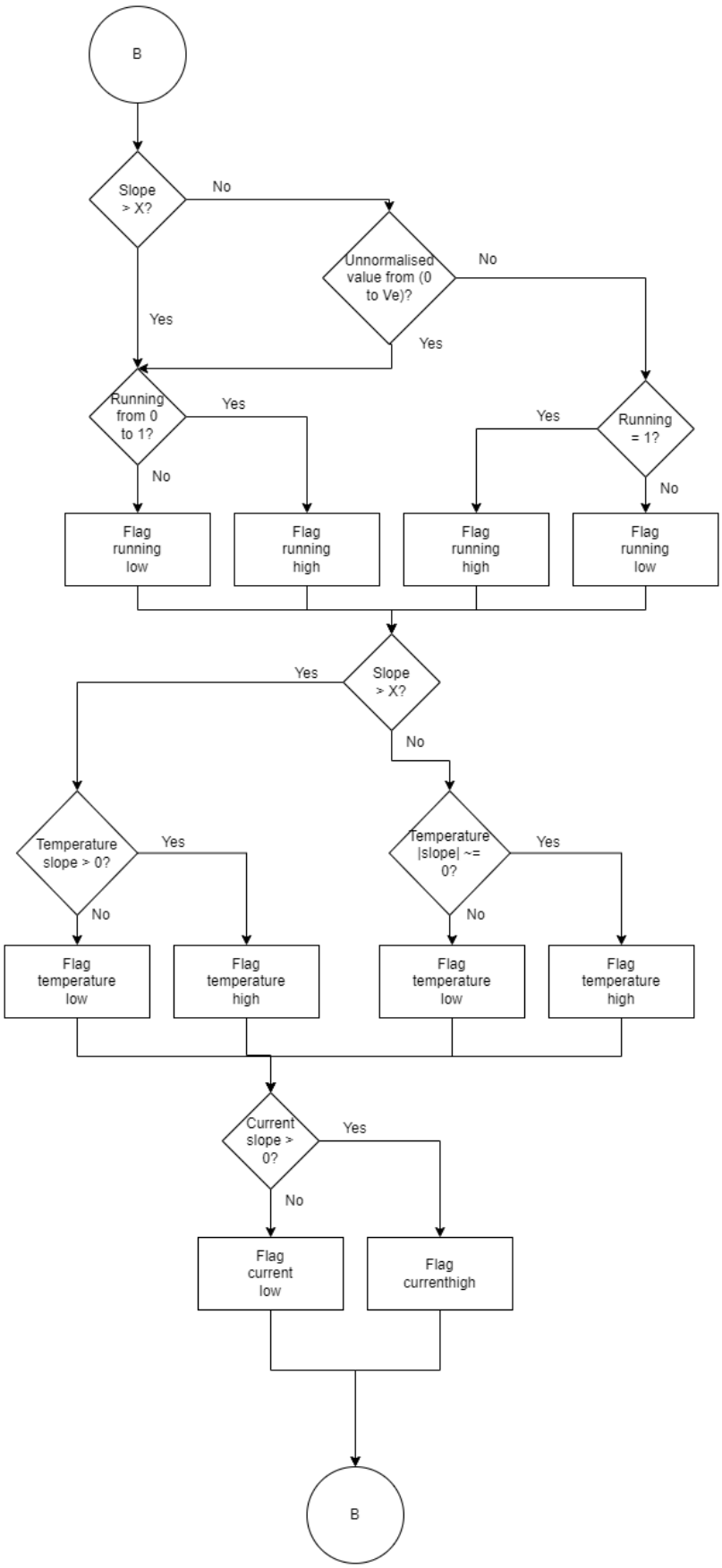

Figure 22: Contextual integrity flow of positive sections for vibration

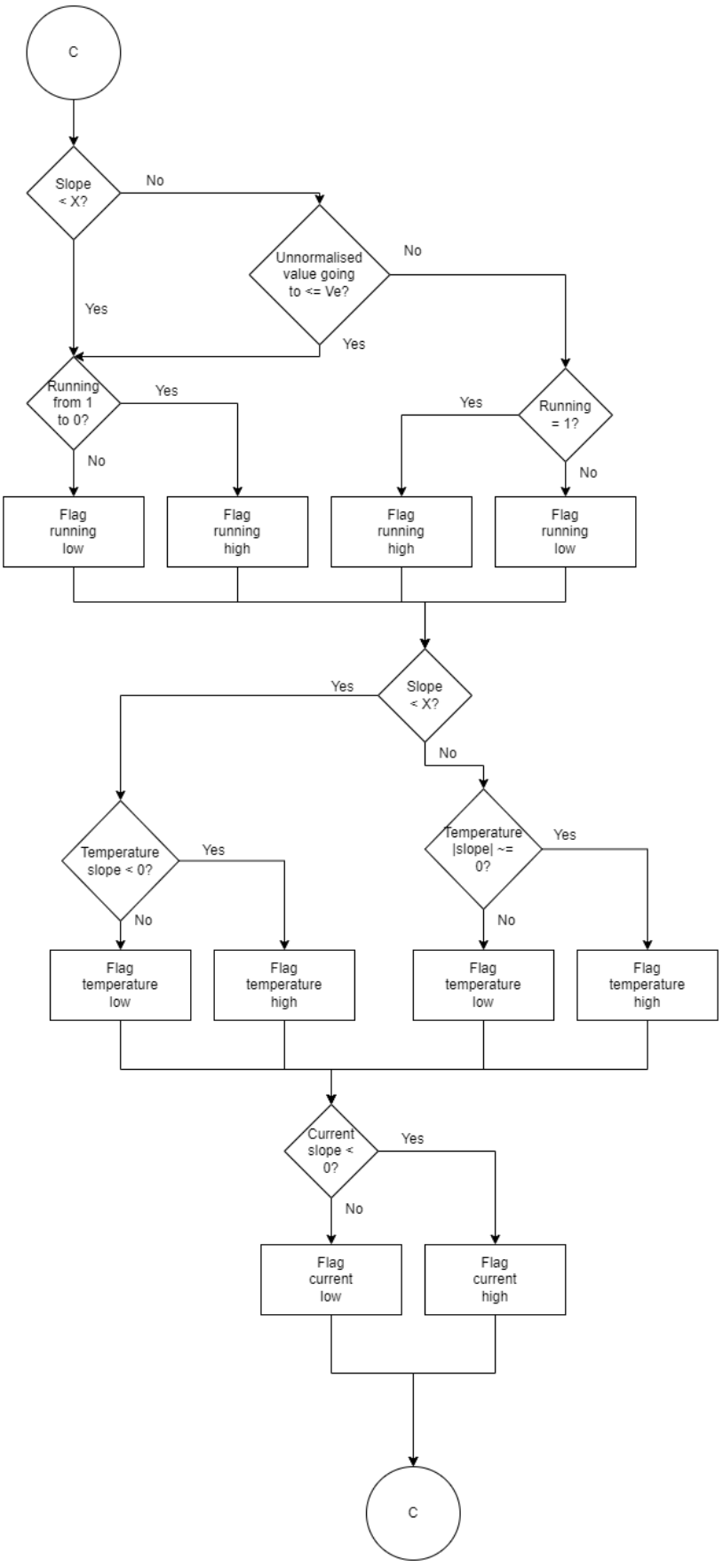

Figure 23: Contextual integrity flow of negative sections for vibration

After the data points have been evaluated, the results are combined into a truth table as represented by Table 7. By applying Equation 20, a contextual integrity score is calculated for each data point.

The system then calculates an overall score for each data point, using Equation 21, using the intrinsic and contextual integrity scores, saving the results to the database for future use as shown in Figure 7.

2.6 Summary

Section 2.1 provided the layout of the chapter. The scope of the study was defined in Section 2.2, with a few restrictions being highlighted. Section 2.3 introduced the methodology for this study. In Section 2.4, methods were investigated to determine the integrity of single-source condition-based maintenance data from both an intrinsic and contextual perspective. Section 2.5 proposed a software system design to calculate the integrity of single-source condition-based maintenance data by implementing the method proposed in Section 2.4.

Chapter 3

Results

3.1 Introduction

In this chapter, the system design from Chapter 2 was implemented on various case studies to validate whether it addresses the need identified in Chapter 1.

Section 3.2 describes the software system specifics and verification. The software system functions are verified to be performing as intended by using three trial datasets with known characteristics. These datasets include two clean datasets and one dataset with errors – termed the *erroneous* dataset. The system analyses of these trial datasets are manually compared to the known features (various categories and quantities of erroneous data points).

Multiple case studies were used to showcase the versatility of the system. Section 3.3 introduces the case studies and elaborates on how they were chosen. Section 3.4 discusses the results obtained from the case studies from various angles. Section 3.5 gives an overview of the system performance, and Section 3.6 concludes the chapter.

3.2 System specifics and verification

Using the software system design described in Chapter 2, a software system was created using the technologies described in Table 23. However, before implementing the system on case studies, it was first verified to ensure that it is correctly calibrated and produces accurate results.

Table 23: Implemented software system specifics

System component	Technology used
NoSQL database	MongoDB
SQL database	MySQL
Software language	C#

To verify the software system, three control datasets were evaluated by the system for which the integrity of each data point was known. After the system analysed the datasets, a manual inspection was performed to quantify the system's accuracy. Two clean datasets and an erroneous dataset was used for verification. Each of these datasets was manually inspected for low-integrity data and selected for a single component. Each dataset contains 48 data points per linked characteristic for a total

of 192 data points per set. The detailed results for each dataset can be found in *Appendix B: Verification results.*

3.2.1 Clean dataset

Using a clean dataset identifies whether the proposed system overcompensates or is uncalibrated. This is evident by incorrectly flagging any data point as unreliable when the entire dataset only includes reliable data points. The normalised data used are displayed in Figure 24, with the summarised results in Figure 25.

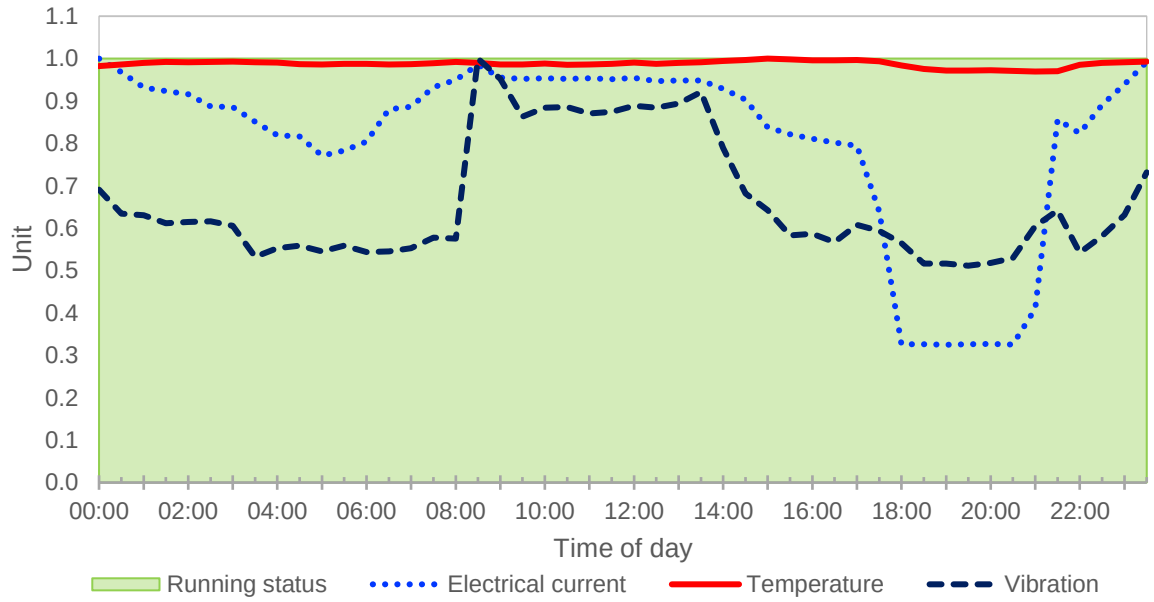

Figure 24: Normalised values of an always-running clean dataset

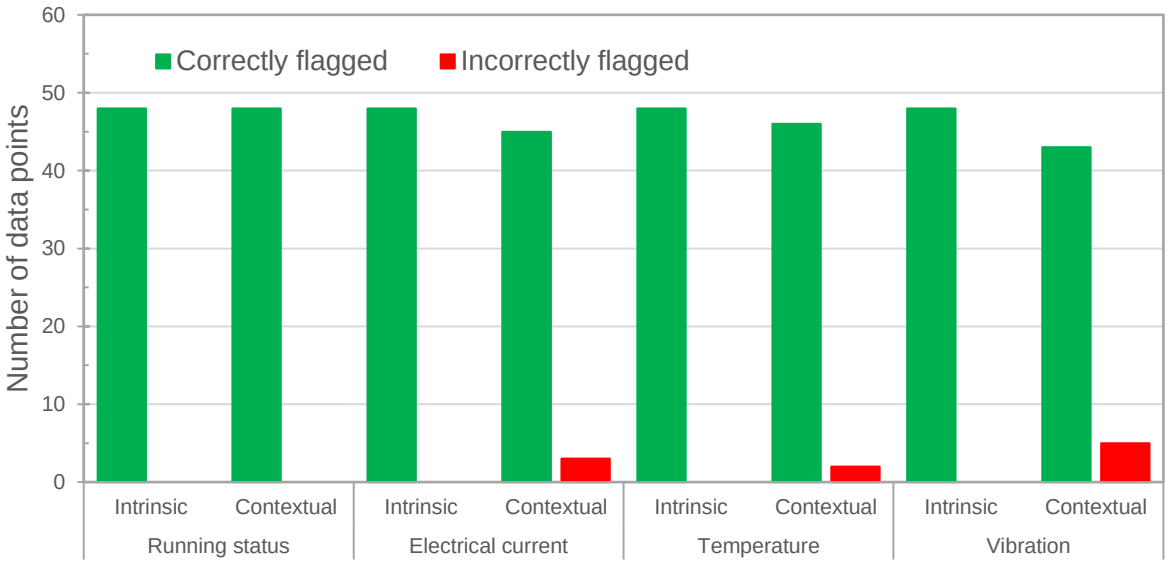

Figure 25: Clean dataset verification summary

Ultimately, the system was reasonably accurate with its assessment of the clean dataset, correctly identifying 94.8% of data points as high-integrity data points. It was concluded that the system performed well overall despite the ten incorrectly classified data points.

To ensure that the system could accurately classify data points while a component alternates between the *on* and *off* states, a second clean dataset was selected in which the component switches from the *off* state to the *on* state and back again. The normalised values of the data streams are displayed in Figure 26, with the summarised results from the individual streams displayed in Figure 27.

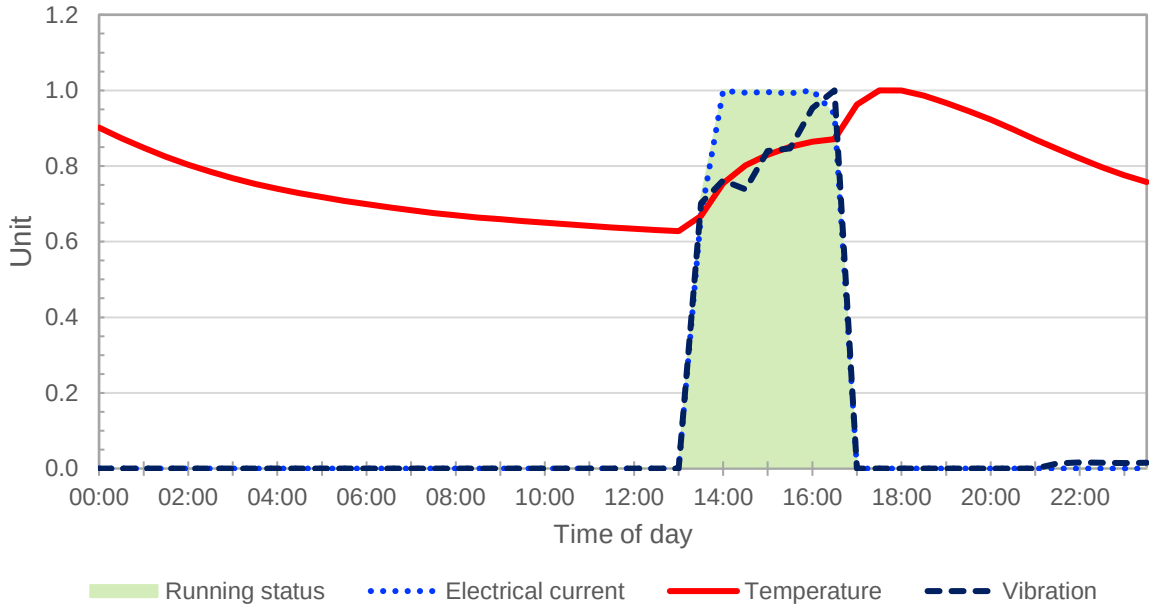

Figure 26: Normalised values of a state-switching clean dataset

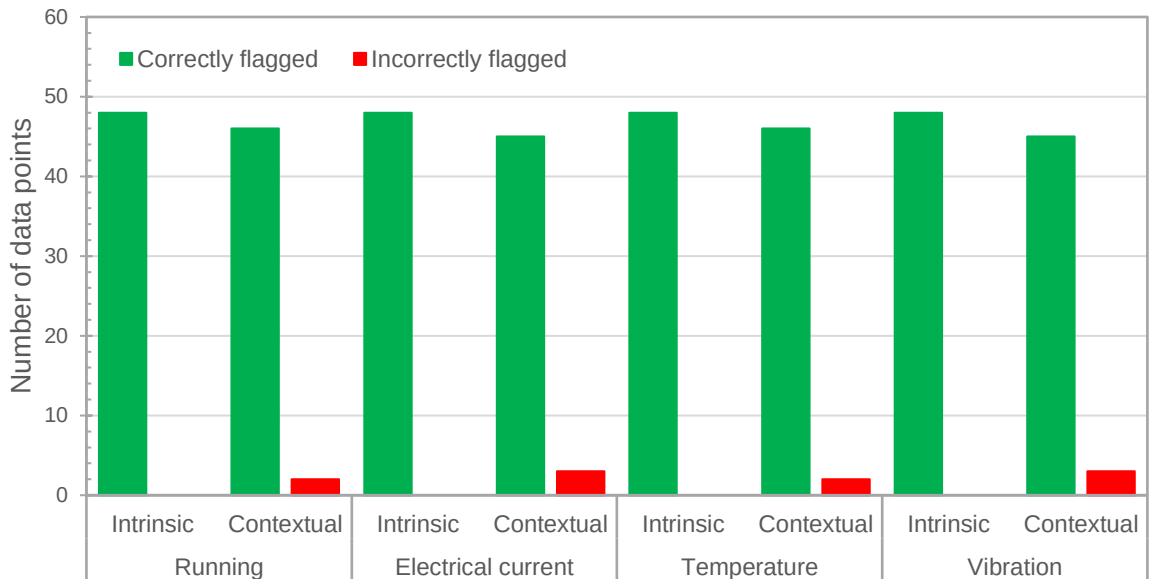

Figure 27: State-switching clean dataset verification summary

Ten data points in total were incorrectly identified as unreliable data points. Two observations were made from this dataset. Firstly, the incorrectly flagged data points correlated between the running status and temperature datasets as they correlated between the electrical current and vibration datasets. Secondly, it was observed that temperature does not follow the same trends as vibration and electrical current when the component switches from the on to the off state. Whereas vibration and electrical current closely follow the running status, the temperature does the opposite and increases when the component switches off. This can be attributed to Newton's law of energy conservation, as the existing energy in the system is converted into heat energy when the component is switched off.

3.2.2 Erroneous dataset

Using an erroneous dataset identifies whether the proposed system is ready to be applied to case studies, as the dataset represents what the system is likely to encounter. The normalised data streams for the erroneous dataset are displayed in Figure 28, with the summarised results displayed in Figure 29.

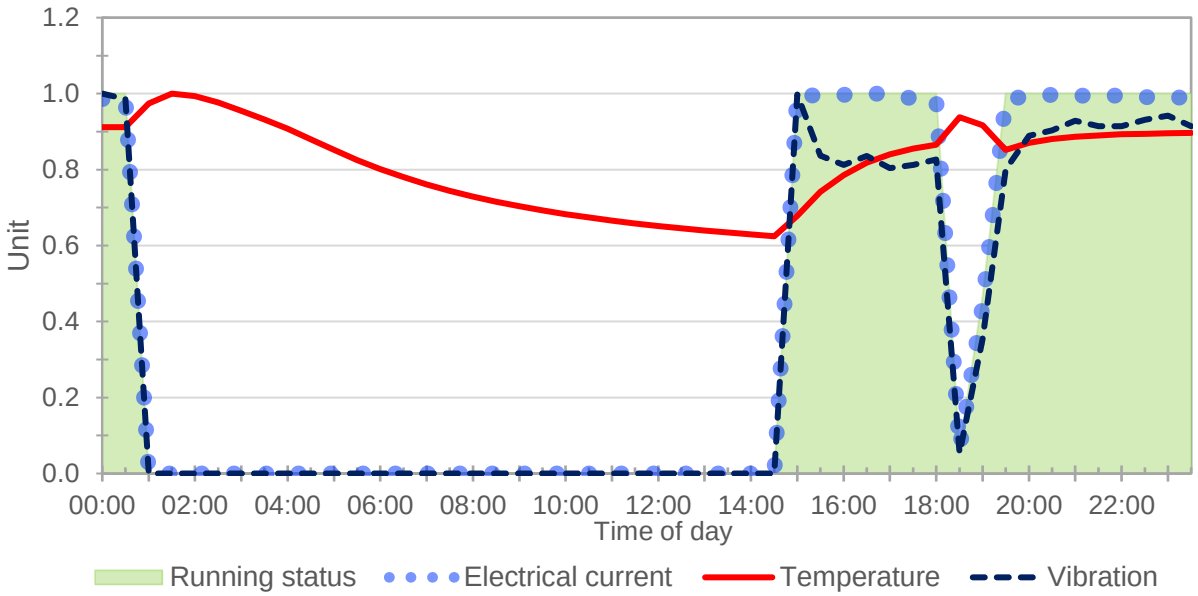

Figure 28: Normalised values of the erroneous dataset

Figure 29 illustrates the summarised results of the erroneous dataset verification. The results show that the system incorrectly identifies more unreliable data points from a contextual perspective than an intrinsic perspective. Despite this, the system could still correctly classify approximately 94 % of the data points, regardless of the reliability.

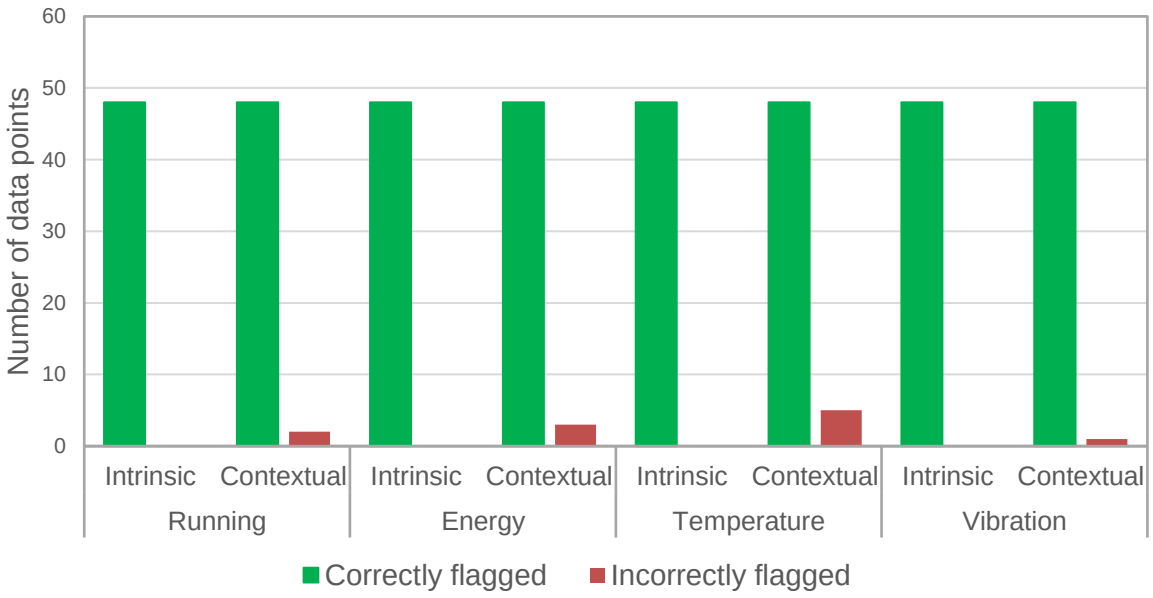

Figure 29: Erroneous dataset verification results

With the system accurately classifying data points for both a clean and an erroneous dataset, it was deemed ready for case studies implementation.

3.3 Criteria for implementation

The proposed system was implemented on case studies of twenty pieces of equipment (components) to ensure the genericism of the design and identify any design biases.

A mining company was identified that had implemented condition-based maintenance on their equipment and used a condition monitoring system making use of SCADA to capture data. The company had limited measuring equipment installed, resulting in only single-source data being available, spread across eleven sites within the mine. Five components with the longest history of erroneous data measurements were chosen for each of the four main equipment types identified in Chapter 1. These twenty components are listed in Table 24, categorised according to component type and site.

Table 24: Case study components by type and site

Component name	Component type	Site name
Component A	Pump	Site A
Component B	Pump	Site B
Component C	Pump	Site C
Component D	Pump	Site D
Component E	Pump	Site E
Component F	Fan	Site D
Component G	Fan	Site F
Component H	Fan	Site G
Component I	Fan	Site H
Component J	Fan	Site I
Component K	Compressor	Site C
Component L	Compressor	Site D
Component M	Compressor	Site F
Component N	Compressor	Site J
Component O	Compressor	Site K
Component P	Fridge plant	Site C
Component Q	Fridge plant	Site D
Component R	Fridge plant	Site G
Component S	Fridge plant	Site I
Component T	Fridge plant	Site K

3.4 Case studies

Data records for each component were evaluated for the entirety of 2020, January to December, resulting in approximately 1.26 million data points. The results of the system were manually inspected for correctness and are presented in various formats to provide insights into the system behaviours.

3.4.1 Components

Each component was evaluated in isolation to ensure that they did not interfere with one another. The results for each component are presented in three complementary formats.

Firstly, the percentage data reliability for a component is displayed as a pie chart, with data categorised as *reliable*, or flagged exclusively by *contextual* methods, flagged exclusively by *intrinsic* methods, or *shared* data points, flagged by both intrinsic and contextual methods (Example: Figure 30).

Secondly, a bar chart analyses the number of data points flagged as reliable, unreliable exclusively by *contextual* methods, unreliable exclusively by *intrinsic* methods, and unreliable by *shared* (both *intrinsic and contextual*) methods per characteristic – vibration, running status, electrical current and temperature (Example: Figure 31).

Thirdly, interesting observations made by the system are highlighted if applicable.

A selection of components that contained interesting observations is presented in this section. A comprehensive presentation of the results for all components can be found in *Appendix C: Case study results*.

Component B - Pump

Component B was identified as a component with the most unreliable data, highlighting a problem in the measurement equipment used to monitor the component. Figure 30 displays the reliability statistics for Component B. The contextual methods flagged roughly 40% of data points as unreliable.

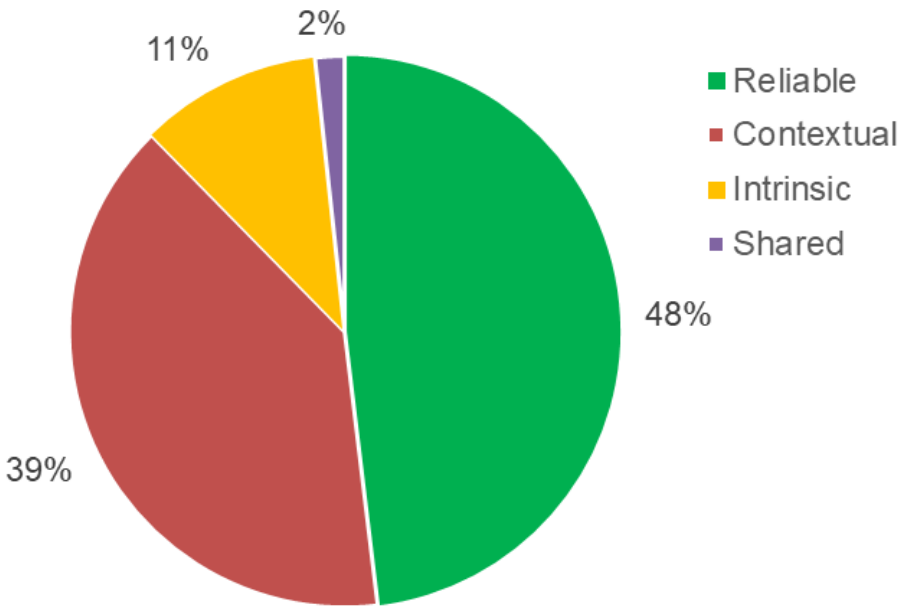

Figure 30: Component B data reliability for January to December 2020

Figure 31 displays the spread of unreliable data across the different characteristics. The temperature characteristic was the only data stream with more than 60% reliable data points. The vibration characteristic was heavily flagged by the intrinsic methods, accounting for more than 40% of the data points in the data stream.

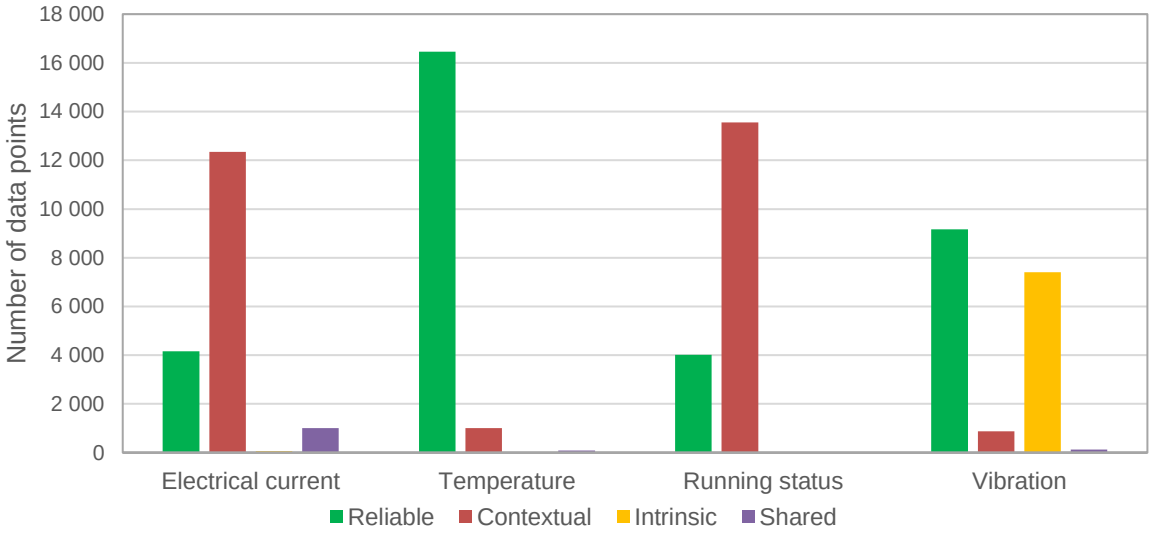

Figure 31: Analysis of component B reliability by characteristic

Figure 32 is a diurnal plot of the four characteristics measured on the 4th of January 2020. The component is switched off, as both the running status and electrical current drawn are zero. Considering the minimal variation in the temperature values, further claims are made that the component is indeed in the *off* state. A corresponding vibration value of between zero and the environmental vibration is expected. For this

component, the environmental vibration was calibrated at zero. The figure illustrates that the vibration measurements never go below 0.19, indicating that the vibration sensor was not correctly calibrated, and as a result, has a floating value.

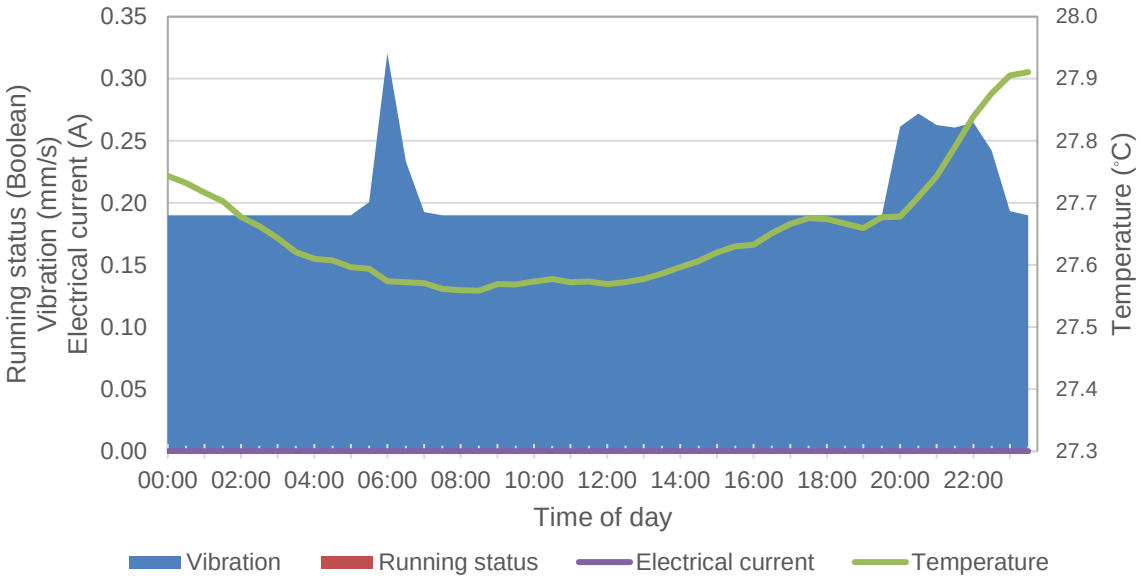

Figure 32: Uncalibrated vibration sensor for Component B (4 January 2020)

During the year, the state of the sensors deteriorated, reflected in Figure 31 above by the large fraction of unreliable data flagged by the contextual methods. The errant behaviour for Component B at the year-end is illustrated in a diurnal plot for 31 December 2020 (Figure 33), showing the conflicting operational states between the different characteristics. From a temperature perspective, the component was off between 00:00 and 03:00, switched on at 03:00, switched off between 05:00 and 06:00 and remained off for the rest of the day. From an electrical current perspective, the component was in operation for the entire day, with reduced loads/strain for 03:00–06:30 and 19:30–21:00. From a running perspective, the component was only switched on between 03:30 and 06:30. The vibration reading implies that the component was never in operation as it read a constant (small) negative value.

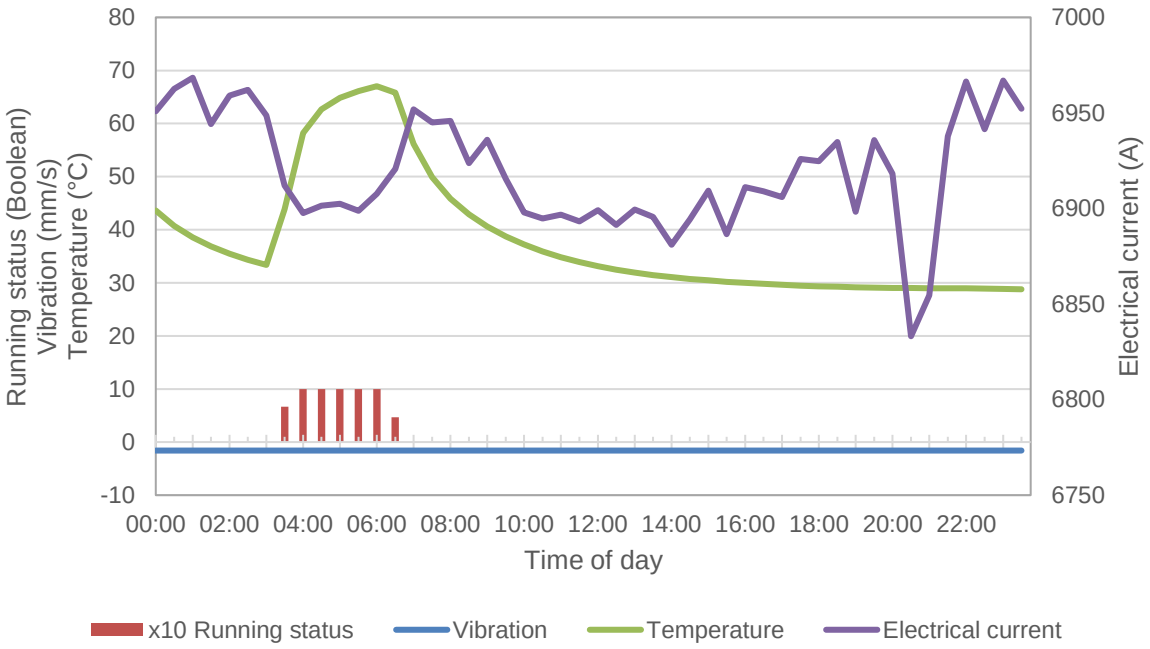

Figure 33: Severe sensor issues identified for Component B (31 December 2020)

From the values, it is clear that the vibration sensor is uncalibrated or incorrectly connected, as it should not record negative values. Regarding the other three characteristics, the temperature and running status seem to align and to be the inverse of the electrical current. It is unclear what the actual operational state of the component was for the day. The safest outcome would be to dismiss all data streams as being unreliable. Component B supports the need identified for this study in Chapter 2, as both the intrinsic and contextual methods identified different data as being unreliable.

Component D - Pump

Component D contained an even split between data points flagged as unreliable by the intrinsic and contextual methods. Figure 34 might suggest that the measurement equipment used for Component D is not correctly calibrated. This is suggested as there is an even split between unreliable data identified by intrinsic and contextual methods without being flagged by both.

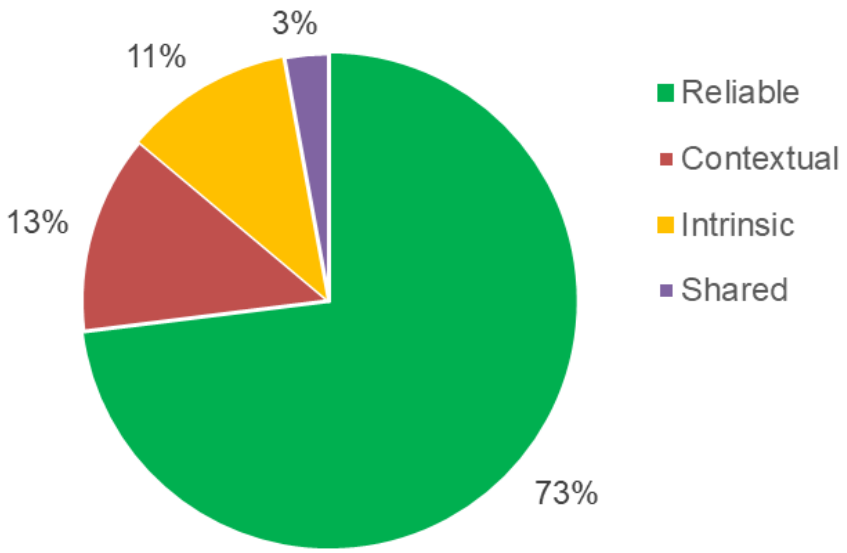

Figure 34: Component D data reliability

Figure 35 illustrates how the unreliable data identified by the contextual methods are distributed between the four characteristics. It can be seen that the majority of unreliable data identified by the intrinsic methods are concentrated to a single characteristic - vibration.

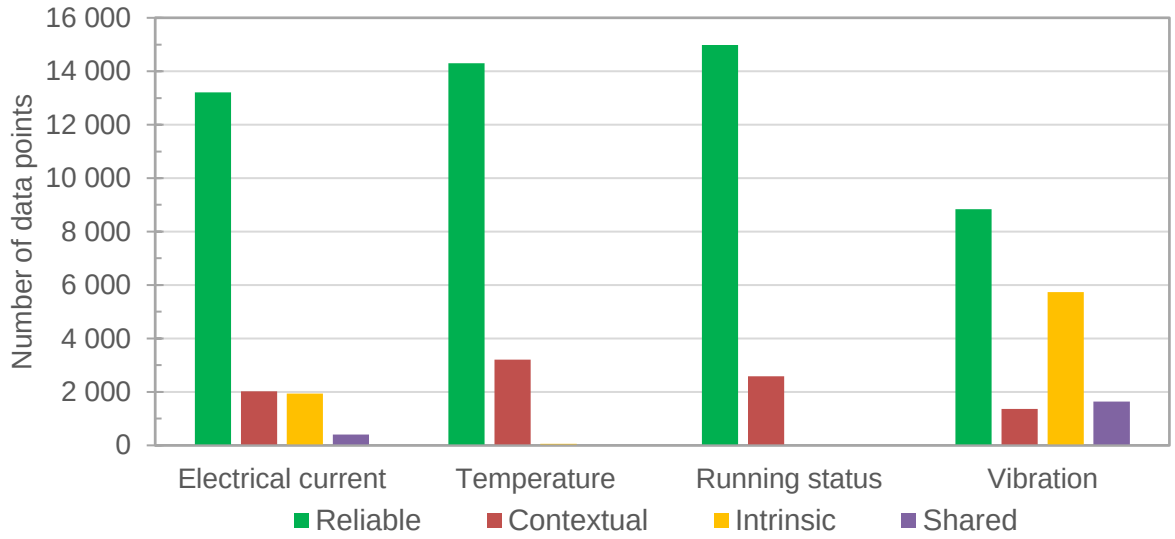

Figure 35: Analysis of component D reliability by characteristic

Upon further investigation, it was found that the vibration measurement equipment was not correctly calibrated. In Figure 36, it can be seen that the component was switched off for most of the day. The measurements during the off stages, although hard to see, are negative values. Negative values for the vibration characteristic for this component are impossibilities and were thus flagged as unreliable, regardless of how small the negative values are.

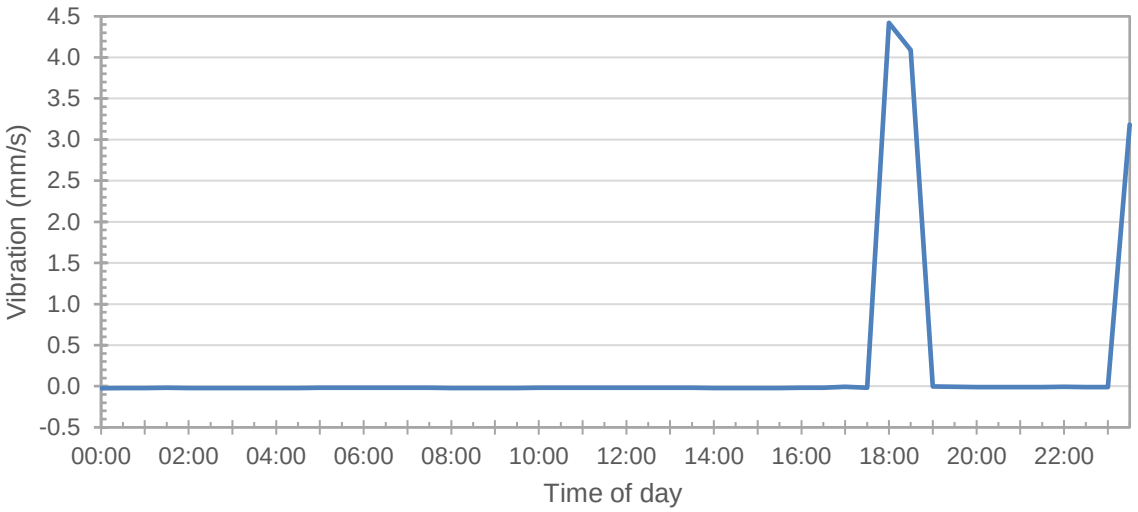

Figure 36: Uncalibrated vibration sensor for Component D

Component H - Fan

Component H had a large amount of unreliable data flagged by the contextual methods, with 20% of the available data being flagged in this manner, shown in Figure 37. In total, 34% of available data is classified as unreliable by the different methods.

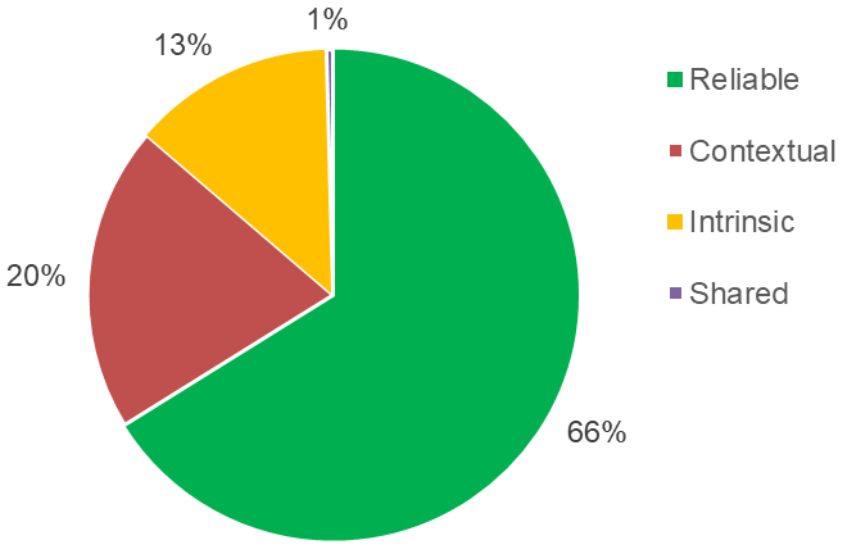

Figure 37: Component H data reliability

Figure 38 shows how the distribution of unreliable data is spread across the different characteristics. Unreliable data is found in the measurements of all four characteristics. However, the temperature characteristic contains more than 30% unreliable data identified by the intrinsic methods. Similarly, the running characteristic contains more than 40% of unreliable data identified by the contextual methods.

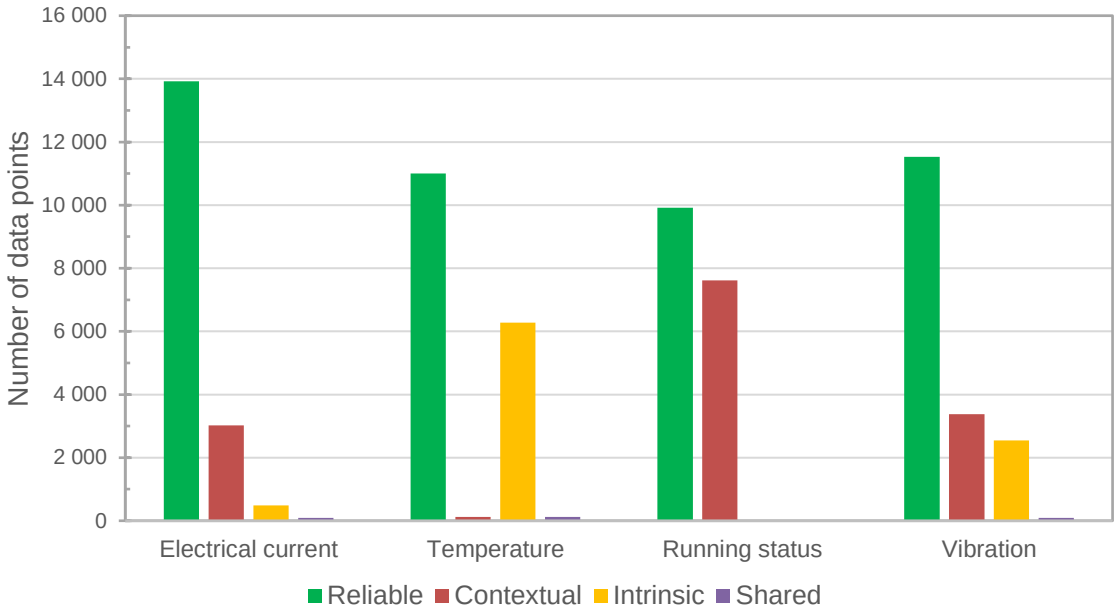

Figure 38: Analysis of component H reliability by characteristic

Figure 39 displays the characteristic profiles for Component H for 13 April 2020. The temperature reading stayed constant with one excursion at 19:00. Furthermore, despite the electrical current reading zero and the vibration reading below the environmental vibration, the running status indicates that the component was in the *on* state. The temperature reading should not be constant. The environmental and machine conditions are variable, influencing the component temperature to induce observable changes. This constant value repeats across several datasets, resulting in the intrinsic methods flagging the data points as unreliable. As for the running status, from the electrical current and vibration perspectives, it seems that the running status values might be erroneously inverted.

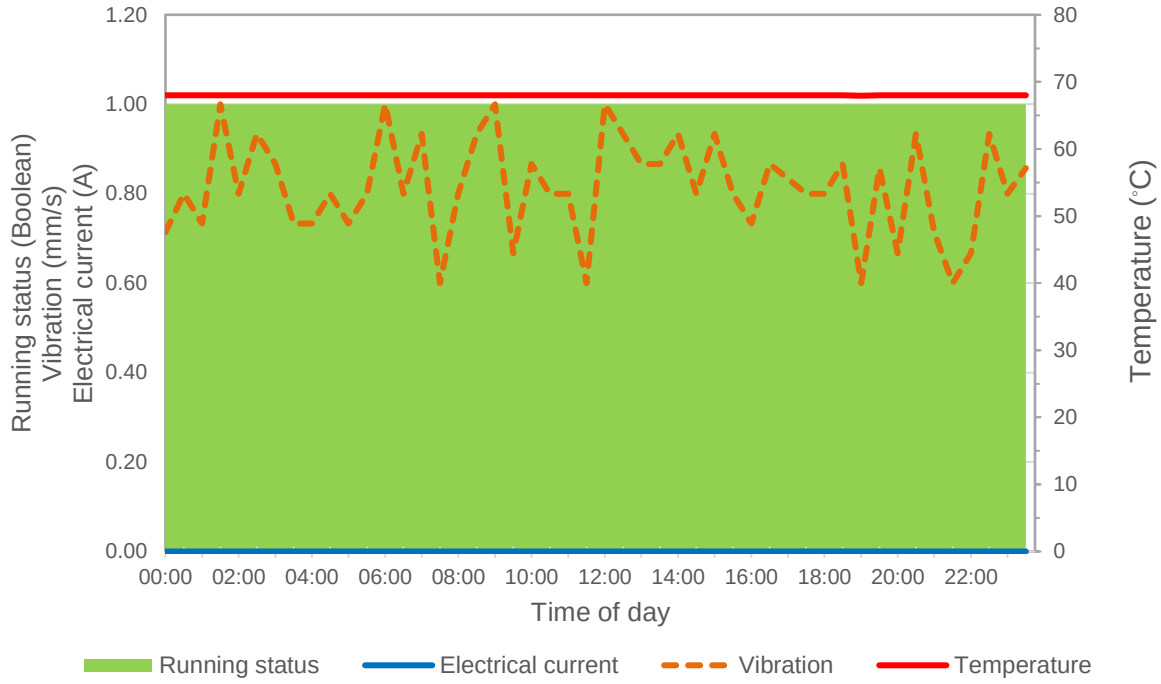

Figure 39: Component H characteristic profile (13 April 2020)

Component P - Fridge Plant

Component P contains 68% unreliable data (Figure 40). Both the intrinsic and contextual methods flagged many data points as unreliable.

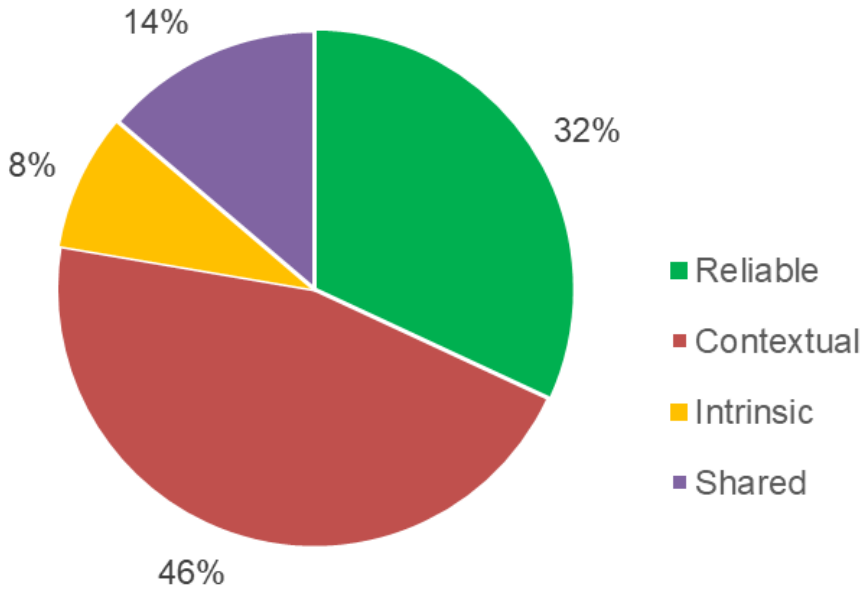

Figure 40: Component P data reliability

Figure 41 illustrates that unreliable data is distributed across all four characteristics.

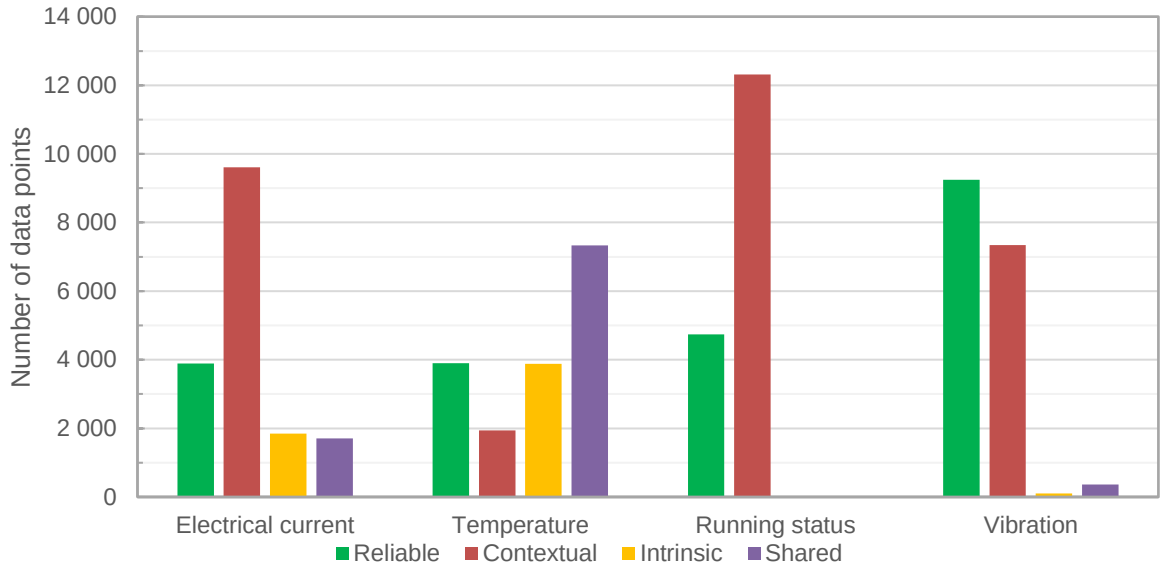

Figure 41: Analysis of component P reliability by characteristic

To validate the results displayed in Figure 41, Figure 42 displays the characteristic diurnal profiles for 25 September 2020. The running status indicates that the component was in operation for the entire day, excluding 17:30 to 21:00. Looking at the other three characteristics, it is implied that the component was actually in the *off* state. Despite the other characteristics contradicting the running characteristic, the changes in the running status do not reflect in any of the other characteristics. This brings the reliability of the running status measurements into question.

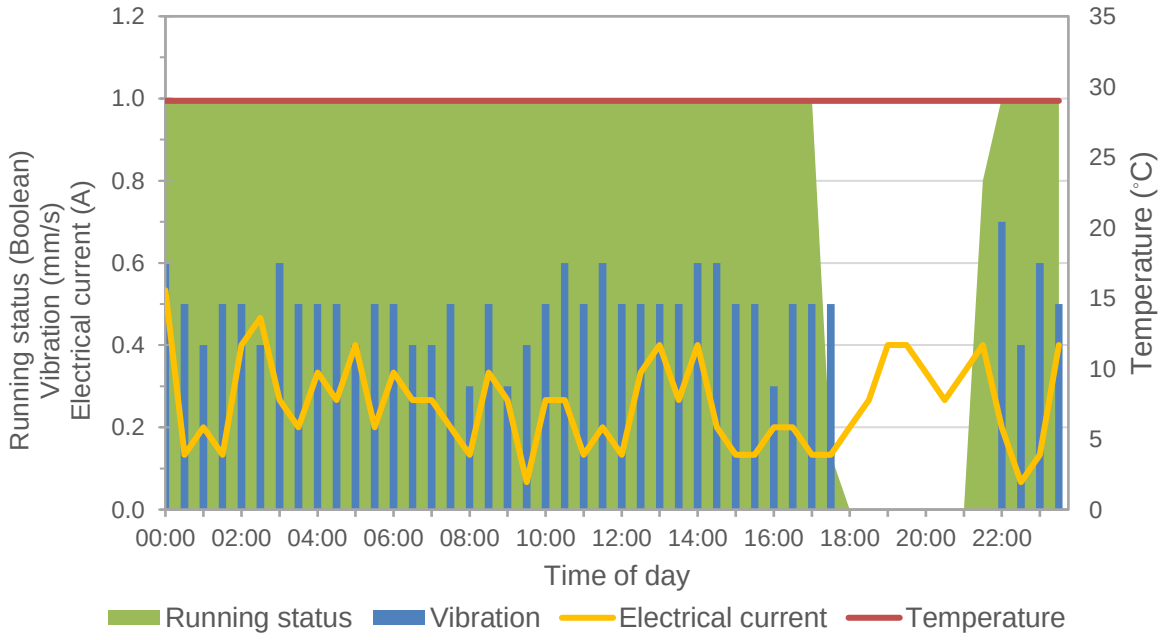

Figure 42: Diurnal characteristic profiles for Component P (25 September 2020)

Evaluating the electrical current drawn, it is noted that the value is not in the expected region of Component P when it is in the *on* state. This implies that the component is in the *off* state. However, the values are not zero as they should be for the *off* state. The implication is that the measuring equipment was not calibrated correctly.

The temperature profile flat-lines on a constant value, which is unexpected for a temperature profile. This brings the reliability of the measurements into question. It could be that the sensor was disconnected from the component and was recording a floating value.

Component Q - Fridge Plant

Figure 43 shows an analysis of data-point reliability by characteristic for Component Q which contains 35% unreliable data.

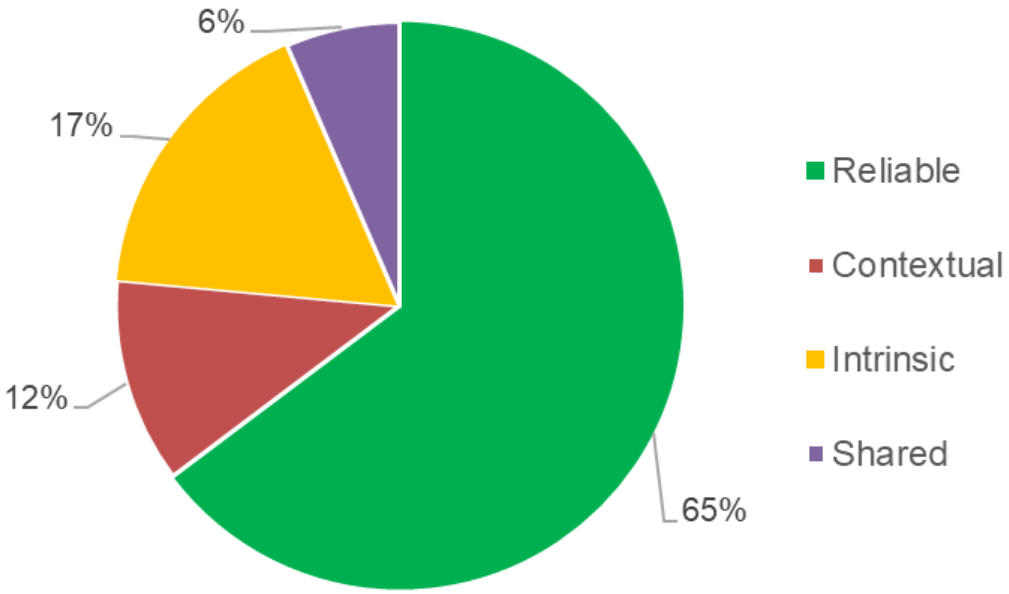

Figure 43: Component Q data reliability

Figure 44 indicates a large number of unreliable data points for the electrical current characteristic as classified by the intrinsic methods. Upon investigation, it was found that the electrical current data stream contains a lot of static, negative values. This would imply that the measurement equipment might have disconnected from the component on multiple occasions and was not correctly calibrated.

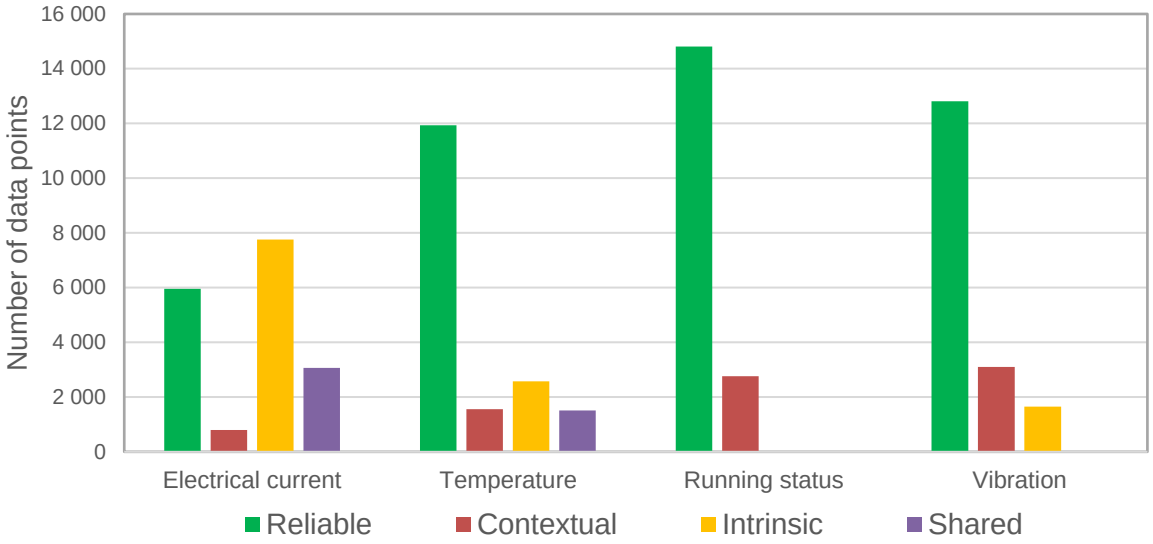

Figure 44: Analysis of component Q reliability by characteristic

Component R – Fridge Plant

The data reliability of the data streams linked to Component R is displayed in Figure 45.

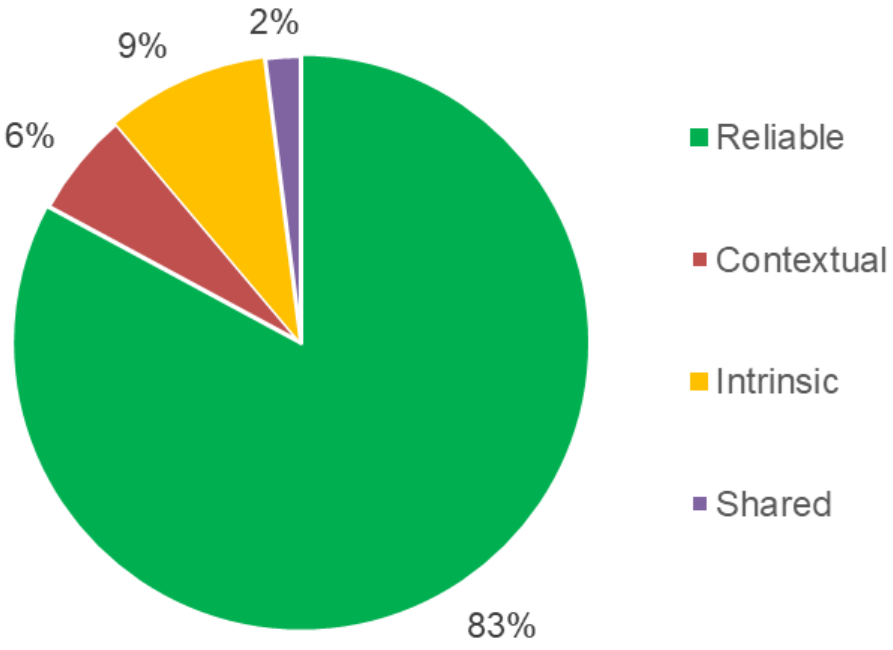

Figure 45: Component R data reliability

Figure 46 illustrates that the electrical current characteristic contains many unreliable data as identified by the intrinsic methods. During a further investigation into the results, it was found that the measurements exceeded the upper limit of possibility defined for the component. Due to the number of occurrences, it was concluded that the upper limit for the component was not correctly configured. As a result, data points

were flagged as unreliable according to the component specifications used that might be reliable when using calibrated specifications for the component.

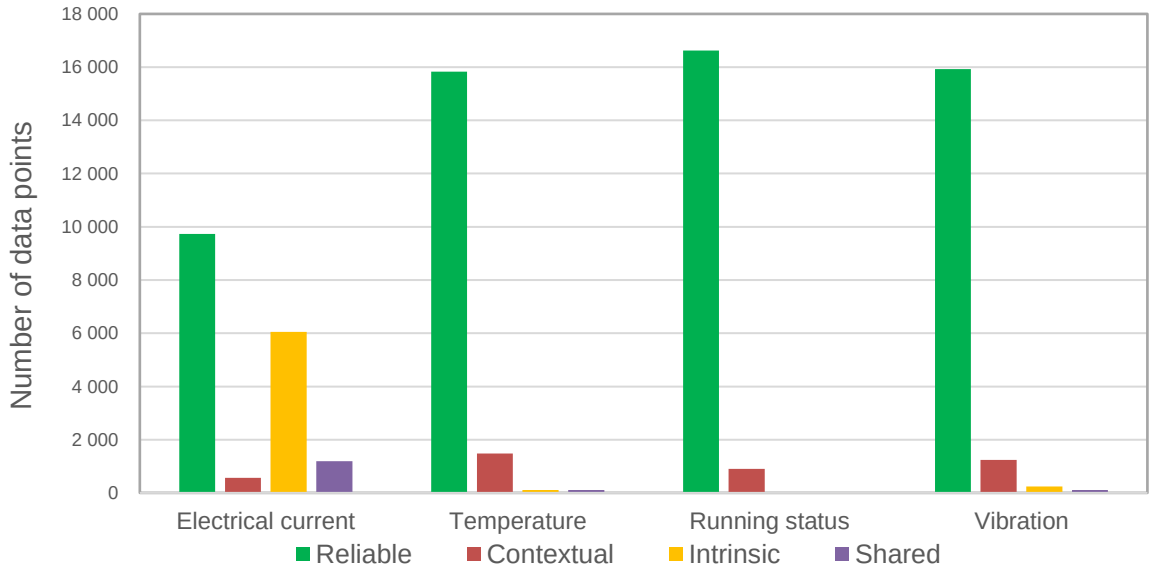

Figure 46: Analysis of component R reliability by characteristic

3.4.2 Component types

After the components were analysed individually, the results were clustered on a component level to gain insight into data reliability issues for the different component types.

The results for the five compressors were combined to create Figure 47, in which it can be seen that compressors generally do not experience excessive levels of unreliable data. The contextual methods identified a majority of the unreliable data points.

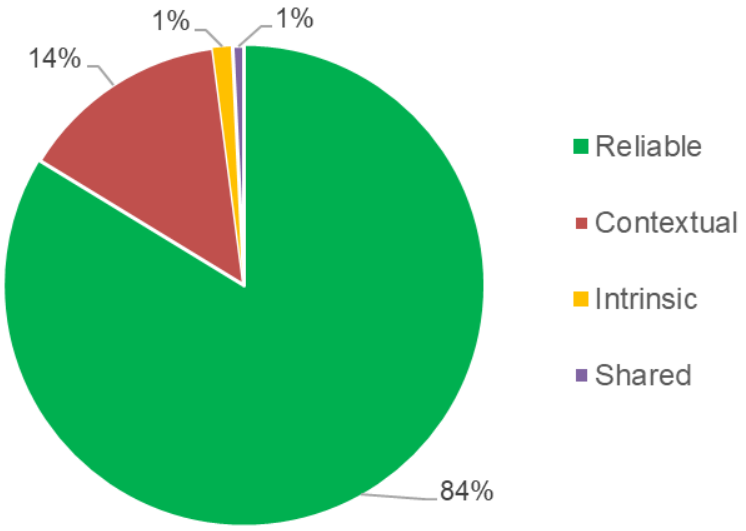

Figure 47: Compressor combined data reliability

The reliabilities of the five fans were combined to create Figure 48. It can be seen that fans had the fewest unreliable data problems, with 11% of available data unreliable. The data points flagged as unreliable were identified by both the intrinsic and contextual methods.

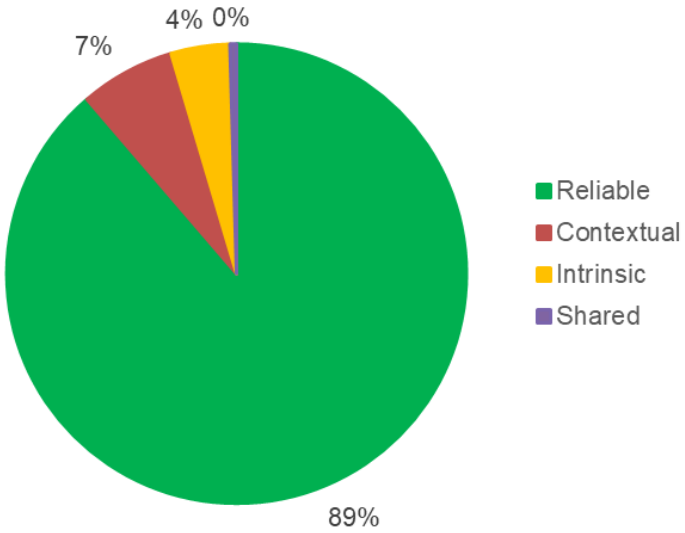

Figure 48: Fan combined data reliability

The five fridge plant case studies combined had the most unreliable data identified by the system, with only 66% of data seen as reliable (Figure 49). Despite both the intrinsic and contextual methods identifying unreliable data, the contextual methods identified a significantly higher fraction.

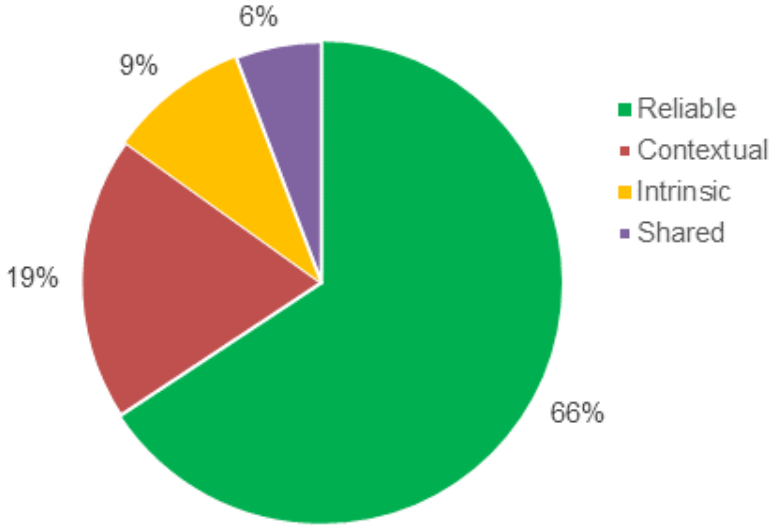

Figure 49: Fridge plant combined data reliability

The five pump case studies were combined to generate Figure 50. Like the fridge plant cases, the pump results show contextual methods contributed a larger fraction of unreliable data identification.

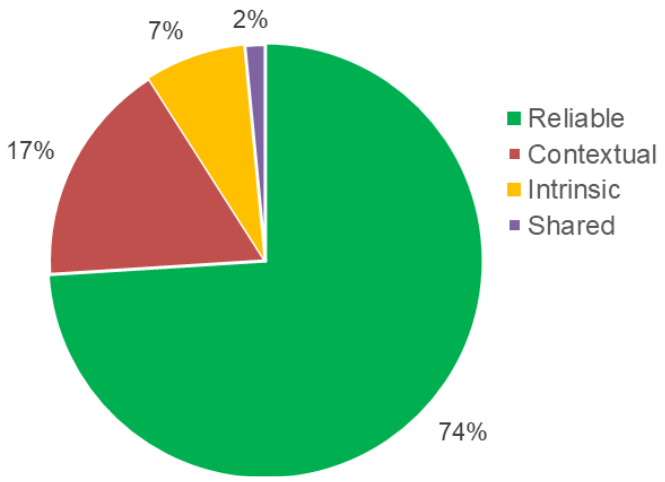

Figure 50: Pump reliability results percentage breakdown

3.4.3 Sites

The results of the twenty case studies were summarised to gain insight into which sites had the most issues with data reliability (Table 25). The number of reliable and unreliable data points is given for each site and the percentage of reliable data. The unreliable data points are categorised according to the identification method, from contextual, intrinsic or combined.

Table 25 indicates that Site H had the best overall data reliability, and Site B had the weakest. The reliability of Site I is flagged as unrepresentative, as the site did not have any data for the first ten months of the year.

Table 25: Summarised data reliability by site (number of data points)

Site	Reliable	Flagged as unreliable			Reliability (%)
		Contextual	Intrinsic	Shared	
A	62 926	7 358	44	24	89.4
B	33 792	27 796	7 471	1 217	48.1
C	133 838	44 223	15 228	11 317	65.4
D	221 958	29 988	22 392	6 842	78.9
E	55 046	11 883	1 634	1 089	79.0
F	117 612	20 274	238	924	84.6
G	104 493	18 329	15 718	1 716	74.5
H	68 929	1 425	11	20	97.9
I	9 029	1 984	304	763	74.7♣
J	61 294	3 441	4 156	497	88.3
K	120 639	15 280	2 069	1 323	86.6

♣ The reliability of Site I is unrepresentative as the site has data for only the last two months of the year.

3.4.4 Characteristics

To determine which characteristic had the most reliability issues, the results for all 20 case studies were tabulated. Table 26 indicates that all four characteristics did well regarding reliability from a characteristic perspective. However, similar to the site results, all reliability percentages are lower because of extensive missing data. Regardless of this, temperature data streams were ultimately the most reliable and electrical current data streams the least.

Table 26: Summarised data reliability by characteristic (number of data points)

Characteristic	Reliable	Flagged as unreliable			Reliability (%)
		Contextual	Intrinsic	Shared	
Electrical current	230 117	52 672	23 919	10 098	72.6
Running	255 024	61 089	0	6	80.7
Temperature	250 946	30 799	0	11 141	85.7
Vibration	253 469	37 421	21 431	4 487	80.0

3.4.5 Data integrity methods

The numbers of low-reliability points identified by the intrinsic, contextual or combined methods are summarised in Table 27. This summary indicates that contextual methods outperformed intrinsic methods in identifying low-integrity points. However, the combination of methods outperformed the individual methods significantly, identifying 192% more unreliable data points than the intrinsic methods alone and 33% more than the contextual methods.

Table 27: Reliability methods results summarised

Method	Number of low-reliability points identified
Intrinsic	94 997
Contextual	207 713
Combined	276 978

3.4.6 Overview

In summary, the overall reliability results for the case studies are displayed in Figure 51. The system identified 22% of the data points as low integrity or unreliable. Of this 22%, 69% were identified by the contextual methods and the remainder by the intrinsic methods, as displayed in Figure 52.

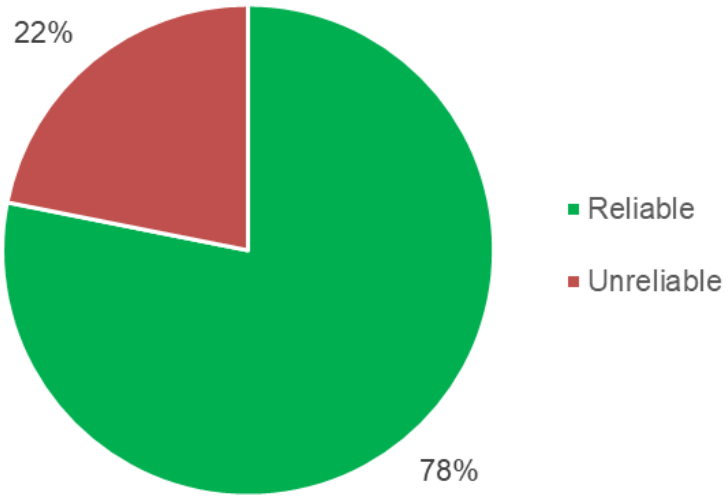

Figure 51: Case study summarised reliability results percentage breakdown

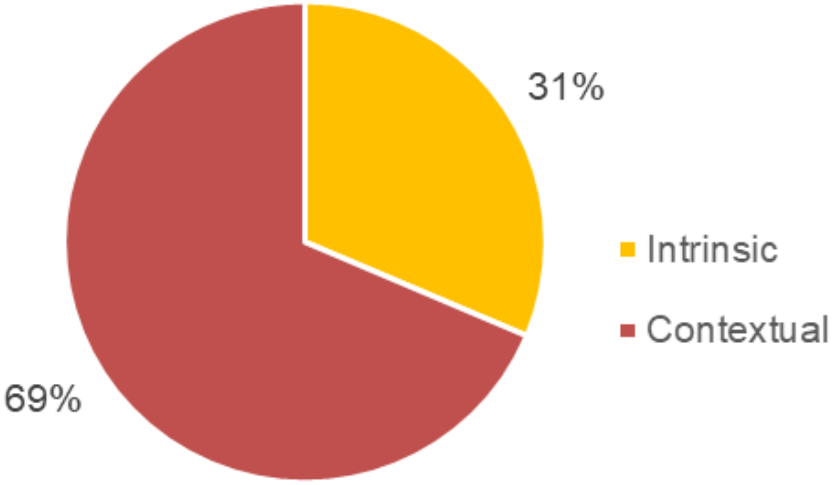

Figure 52: Summarised method reliability results percentage breakdown

3.5 Discussion

The system analysed a total of 1 266 534 data points and identified 276 978 data points as unreliable. This figure does not take the missing data experienced on Site I into consideration. If this missing data was also considered, the total unreliable data for the mine would realistically exceed 25%. This indicates that the case study mine has a severe issue with unreliable data. From the literature discussed in Section 1.1.1,

it was concluded that unreliable data can have severe financial consequences for companies. Having more than a quarter of the business data flagged as unreliable should raise serious concerns for the case study company. Along with the high amount of unreliable data identified, the system also highlighted several issues.

Firstly, missing data is a common occurrence. Despite Site I being the only explicitly-marked site in Table 25 with missing data, all the data streams had varying amounts of missing data. Missing data can be attributed to various factors such as broken/faulty sensors, unstable networks used for data transfer, and downtime on control systems or databases. Ultimately, for the system to analyse data streams more accurately, they need to be complete as required by Equation 18 and Table 8. A supplementary system should be introduced to monitor missing data and alert relevant users to investigate the problem as quickly as possible to prevent negative impacts on the system.

Secondly, the system is susceptible to user error. During the case study for Component G, a configuration error occurred in which the incorrect data stream was linked to a component. Although the system successfully identified the user error using the contextual methods, it could only do so because it was a large error. Smaller user errors might go unnoticed without manual inspection of the data and a critical analysis of the system's results, as illustrated by case study Component R where one of the limits was incorrectly configured. Over time, additional functionality can be introduced to help identify user errors during component configuration. Continuous inspection of the configuration is advised as the components operate in harsh and ever-changing conditions, which could impact the limits for the component

Thirdly, most of the unreliable data points identified by the system can be traced back to uncalibrated sensors. A common sensor-related issue is an incorrect zero-point. These errors are impossible for the system to detect if the faulty readings fall within limits configured for the component and never exceed them. This highlights the need for regular calibration of the sensors, as the measured data are highly likely to be incorrect, which could ultimately negatively impact the decisions made thereon.

Fourthly, the system accuracy is reliant on configuration. For the system to analyse the different components and their characteristic data streams, they must be configured in the database. As part of the configuration, the variable limits in the contextual equations must be specified. If these limits are incorrectly specified, the

system can flag data as unreliable during viable operating conditions in which certain characteristics have abnormal values. The system will not indicate that these values might be incorrectly flagged as it uses Boolean logic and manual inspection or a secondary system that analyses the results of this system will be required to identify these edge-cases after which recalibration of the limits for the component will be required in the database.

Finally, system accuracy varies depending on the data resolution available. If the data resolution differs between data streams linked to the same component, data points on mismatching timestamps will always be flagged as unreliable data. To mitigate this, data sources should preferably use the same resolution per component. For this study, most of the case studies only had access to 30-minute resolution data; however, nearing the end of this study, the measured data resolution for Component A was changed from 30-minute resolution data to two-minute resolution data. To investigate the difference in system accuracy correlating to data resolution, the system was run on both the 30-minute and two-minute resolution data over the same period as discussed in *Appendix C:C-21: Component A data resolution*. The results are summarised in Table 28 and indicate that the system's accuracy can be significantly improved by using higher-resolution data.

Table 28: Summarised results for data resolution difference case study

Resolution	Reliable	Contextual	Intrinsic	Shared	Reliability (%)
2-minute	2774	106	0	0	96.3
30-minute	172	20	0	0	89.6

3.6 Summary

Section 3.1 gave an overview of the chapter and detailed what the chapter layout is. Section 3.2 described how the proposed software system design was implemented on a new software system. The section continues to detail how the system was verified using three sample data sets with known integrity. Section 3.3 detailed how case studies were identified and briefly discussed the breakdown of the chosen case studies. Section 3.4 discussed the results obtained from the twenty implemented case studies and ended by summarising the system performance from various angles. Section 3.5 concluded the chapter with a discussion of the system.

Chapter 4

Conclusion

4.1 Discussion

As data generation and availability increase, more knowledge is extracted from data and used for decision making. As highlighted from the literature, using unreliable data can lead to ill-informed decisions and negative consequences; therefore, a stronger emphasis should be placed on ensuring high data integrity.

Implementing data-driven decision making with high-integrity data can add valuable business insights and increase profitability. In the mining industry, profitability is a constant concern due to various factors identified in the literature; thus, unnecessary expenditure must be reduced to remain profitable.

Ensuring that the equipment is in a healthy operational state and minimising downtime are vital for generating income. Implementing an efficient, condition-based maintenance strategy using high-integrity data helps achieve these goals. To this end, shortcomings in literature were identified when only single-source data was available.

Chapter 1 gave some background into the need for data integrity and its increasing importance in the mining sector. A literature review was conducted, and a need was identified to investigate the integrity of single-source condition-based maintenance data using the combination of intrinsic and contextual methods. This study proposed two novel contributions, and study objectives were identified. These novel contributions can help identify low-integrity data in the future, helping reduce the impact of unreliable data on decision making.

Chapter 2 established the scope and listed some restrictions and assumptions that would be followed. The methodology was outlined in Section 2.3. Section 2.4 investigated methods that can be used to estimate the data integrity from both an intrinsic and contextual perspective, completing the first novel contribution of this study. Section 2.5 described the design of a software system to estimate data integrity using the methods discussed in Section 2.4, producing the second novel contribution.

Chapter 3 started by creating a software system using the proposed software system design, verifying it with three test datasets and then discussing the criteria for implementing the system on case studies. Twenty case studies across eleven sites were identified for implementation due to their history of data-related issues.

Section 3.5 discusses the system's performance during the case studies and highlights some system shortcomings.

The novel contributions of this study made it possible to identify low-integrity data and delivered a way to identify the sources of low-integrity data. Using these contributions, 22% of the data points were identified as unreliable, reducing the number of ill-informed decisions that would have been made with them. Analysing the results further, the novel contributions identified which components, and more importantly, which sites had the most problems correctly capturing data. Despite the number of sites having low-integrity data being small, they have a large overall impact on the company.

The case studies used in this study were chosen because of their known history of data-related problems. Had the system been implemented on different case studies, the results could have looked significantly different. In the case where less unreliable data was present, the value of the system would be seen as less significant as it would verify that the case study company did not have a severe problem with unreliable data. However, had the system identify the same number of, or more, unreliable data points on components that were thought to be reliable, the value of the system would have increased and add to research showing that companies overestimate the quality of their data [22], [28], as discussed in Section 1.1.1.

With a shift towards a more data-driven world, having high-integrity data will become increasingly important and valuable. As a result, the contributions of this study are a vital first step in the decision-making process. Ultimately, to address the need for this study, the objectives identified in Chapter 1 must be met. All study objectives were met, as summarised in Table 29.

Table 29: Breakdown of implemented study objectives

Contribution	Objective	Addressed	Section
Method for estimating the integrity of single-source condition-based maintenance data	Investigate intrinsic data quality methods	✓	2.4.1
	Investigate contextual data quality methods	✓	2.4.2
	Combined intrinsic and contextual data quality method	✓	2.4.3
Software system design for estimating the integrity of single-source condition-based maintenance data	Implement newly created method into a software system design	✓	2.5
	Create a software system using the design	✓	3.2
	Verify software system	✓	3.2
	Implement software system	✓	3.4

4.2 Recommendations for future work

4.2.1 Expand intrinsic methods

The novel method proposed should be expanded in future to include more intrinsic methods. The proposed method implemented three simple methods to verify intrinsic integrity: hanging data, outliers, and limited exceptions. Although these methods identified unreliable data points effectively throughout the case studies, expanding on this list could help identify edge-case data points as unreliable.

4.2.2 Improved characteristic behaviour

For this study, data streams were placed into context with other data streams on a component level. This enabled the detection of unreliable data streams and data points in context to the component; however, adding additional context to the data streams themselves could also improve unreliable data identification. Using the temperature characteristic as an example, the current system checks whether the temperature behaves as expected in relation to the electrical current drawn, vibration, and running status of the component. Future iterations could include the capability to evaluate whether the temperature behaves as expected, such as whether it increases or decreases unnaturally.

4.2.3 Refine characteristic relations

The equations derived in this study are simplistic representations of the relations between the different characteristics. Future studies could investigate and expand on the available equations that better describe the relations.

4.2.4 Expand supported characteristics

This study focused on the four commonly measured characteristics identified in the literature. Depending on the availability of measuring equipment, these characteristics might not be measured on specific components. In these situations, alternative characteristics should be used to determine the integrity of the measured data. By identifying possible substitutes for each characteristic, the method could be more generic and widely implementable.

4.2.5 Implement error-reduction interface

As mentioned in Section 3.5, the system is susceptible to user errors. To reduce the number of user errors, an intelligent configuration interface should be implemented to help the user configure their components. This interface should highlight potential problems with configurations, such as suspicious limit values for a data stream depending on the characteristic.

4.2.6 Information distribution

This study implemented a software system on multiple case studies, and the results were manually inspected to extract knowledge. A recommended improvement on this study would be implementing an information extraction and distribution system that automatically extracts the information from the analysis results and makes it readily available to the users. An online monitoring platform or automated reporting system would enable easy access to the information and faster reaction times to data-related problems. A notification system could also be implemented to notify relevant personnel of data-related problems identified by the system. This notification system could be implemented in various ways, such as emails, SMSs and in-application notifications on mobile devices.

4.2.7 Correction platform

The system proposed in the study analysed data and identified unreliable data. Future studies could use the results of this system to correct the unreliable data points.

4.2.8 Event-driven system

The system design in this study triggered fault events through manual off-line data analysis. However, as condition monitoring needs to invoke rapid responses in severe conditions, measured data should be written to the database and processed as close to real-time as possible. The proposed system can be adapted to use an event-driven architecture to analyse data as it comes into the database rather than on a schedule.

4.2.9 Application purpose refinement

Data is becoming a valuable commodity and ensuring the integrity thereof should be a priority. As a result, the system can be used as a base and expanded or modified using different methods to make it suite a specific need better. To make the system more component specific and increase the accuracy of the system, a Neural Network or statistical methods can be implemented. Implementing Fuzzy Logic would increase the range of output values, allowing for better classification of data depending on the purpose of the system.

– — O — –

References

[1] N. Laranjeiro, S. N. Soydemir, and J. Bernardino, 'A survey on data quality: classifying poor data', *2015 IEEE 21st Pacific Rim International Symposium on Dependable Computing, (PRDC)*, pp. 179–188, 2015, doi: 10.1109/PRDC.2015.41.

[2] M. Obitko and V. Jirkovský, 'Big data semantics in industry 4.0', In: V. Mařík, A. Schirrmann, D. Trentesaux, P. Vrba (Eds) *Industrial Applications of Holonic and Multi-Agent Systems*. HoloMAS 2015. *Lecture Notes in Computer Science*, vol. 9266, pp. 217–229, Springer, Cham. https://doi.org/10.1007/978-3-319-22867-9_19.

[3] J. Patel, 'An effective and scalable data modeling for enterprise big data platform', *2019 IEEE International Conference on Big Data (Big Data)*, 2019, pp. 2691–2697, doi: 10.1109/BigData47090.2019.9005614.

[4] P. Zhang, F. Xiong, J. Gao, and J. Wang, 'Data quality in big data processing: Issues, solutions and open problems', *2017 IEEE SmartWorld Ubiquitous Intelligence and Computing, Advanced and Trusted Computed, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 2017, pp. 1–7, 2017, doi: 10.1109/UIC-ATC.2017.8397554.

[5] D. Li, Y. Gong, G. Tang, and Q. Huang, 'Research and design of mineral resource management system based on big data and GIS technology', *2020 5th IEEE International Conference on Big Data Analytics (ICBDA)* 2020, pp. 52–56, doi: 10.1109/ICBDA49040.2020.9101268.

[6] A. Goosen, 'A system to, quantify industrial data quality', MEng (Computer and Electronic Engineering) thesis, North-West University, Potchefstroom Campus, 2018.

[7] K. Park, M. C. Nguyen, and H. Won, 'Web-based collaborative big data analytics on big data as a service platform', *International Conference on Advanced Communication Technology (ICACT)*, 2015, pp. 564–567, doi: 10.1109/ICACT.2015.7224859.

[8] Y. Ishizuka, W. Chen, and I. Paik, 'Workflow transformation for real-time big data processing', *2016 IEEE International Congress on Big Data (BigData Congress),* 2016, pp. 315–318, doi: 10.1109/BigDataCongress.2016.47.

[9] L. Joubert and L. Louw, 'Towards an internet-of-things framework for assisting quality-controlled-logistics decision making within the fresh produce supply chain', *29th Annual SAIIE Conference (SAIIE29): Steering the 4th Industrial Revolution*, Stellenbosch, pp. 261-274, 2018, https://conferences.sun.ac.za/index.php/saiie29/saiie29/paper/viewFile/3583/516.

[10] J. Lu, Z. Yan, J. Han and G. Zhang, "Data-Driven Decision-Making (D3M): Framework, Methodology, and Directions," in IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 3, no. 4, pp. 286-296, Aug. 2019, doi: 10.1109/TETCI.2019.2915813.

[11] T. Zhuang, M. Ren, X. Gao, M. Dong, W. Huang and C. Zhang, "Insulation Condition Monitoring in Distribution Power Grid via IoT-Based Sensing Network," in IEEE Transactions on Power Delivery, vol. 34, no. 4, pp. 1706-1714, Aug. 2019, doi: 10.1109/TPWRD.2019.2918289.

[12] V. Grover, R. H. L. Chiang, T. P. Liang, and D. Zhang, 'Creating strategic business value from big data analytics: A research framework', *J. Manag. Inf. Syst.*, vol. 35, no. 2, pp. 388–423, 2018.

[13] E. Brynjolfsson and K. McElheran, 'The rapid adoption of data-driven decision-making', *Am. Econ. Rev.*, vol. 106, no. 5, pp. 133–139, 2016.

[14] C. M. Caffrey, J. Flak, I. Marttila, N. Pesonen, and P. Pursula, 'Development of a reader device for fully passive wireless sensors', *2017 IEEE SENSORS*, 2017, pp. 1-3, doi: 10.1109/ICSENS.2017.8234409.

[15] A. Zaslavsky, C. Perera, and D. Georgakopoulos, 'Sensing as a service and big data', *International Conference on Advances in Cloud Computing (ACC)*, 2012, Bangalore, India. *ArXiv* abs/1301.0159 (2013), 8 pp.

[16] S. Sakr and A. Elgammal, 'Towards a comprehensive data analytics framework for smart healthcare services', *Big Data Res.*, vol. 4, June 2016, pp. 44–58.

[17] C-c Qi, 'Big data management in the mining industry', *Int. J. Miner. Metall. Mater.*, vol. 27, no. 2, pp. 131–139, 2020.

[18] S. Yin and O. Kaynak, 'Big data for modern industry: challenges and trends [Point of View]', *Proceedings of the IEEE*, vol. 103, no. 2, pp. 143–146, 2015, doi: 10.1109/JPROC.2015.2388958.

[19] L. Coetzee, W. Booysen, and J. F. van Rensburg, 'A systems approach to sustain RSA Section 12L tax incentive projects', *29th Annual SAIIE Conference (SAIIE29): Steering the 4th Industrial Revolution*, Stellenbosch, pp. 169–184, 2018.

[20] R. H. Hariri, E. M. Fredericks, and K. M. Bowers, 'Uncertainty in big data analytics: survey, opportunities, and challenges', *J. Big Data*, vol. 6, 44, 2019, https://doi.org/10.1186/s40537-019-0206-3.

[21] S. W. van Heerden, J. Herman, and J. C. Vosloo, 'The need for information system data maintenance', *28th Annual Southern African Institute for Industrial Engineering Conference (SAIIE28)*, 2017, pp. 3370.1–3370.13, https://www.saiie.co.za/system/files/2021-11/SAIIE28%20Proceedings.pdf.

[22] J. N. de Meyer, 'Validating the integrity of single source condition monitoring data', MSc (Computer and Electronic Engineering) thesis, North-West University, Potchefstroom Campus, 2020.

[23] A. Oussous, F. Z. Benjelloun, A. Ait Lahcen, and S. Belfkih, 'Big data technologies: A survey', *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 30, no. 4, pp. 431–448, 2018.

[24] J. N. du Plessis, B. M. Friedenstein, and J. C. Vosloo, 'The need for effective application of data visualisations in management information systems', *28th Annual Southern African Institute for Industrial Engineering Conference (SAIIE28)*, 2017, pp. 3377.1–3377.13, https://www.saiie.co.za/system/files/2021-11/SAIIE28%20Proceedings.pdf.

[25] C. Adrian, R. Abdullah, R. Atan, and Y. Y. Jusoh, 'Expert review on big data analytics implementation model in data-driven decision-making', *2018 4th International Conference on Information Retrieval and Knowledge Management: Diving into Data Sciences(CAMP)* 2018, pp. 13–17.

[26] S. Maurya, V. Singh, N. K. Verma and C. K. Mechefske, "Condition-Based Monitoring in Variable Machine Running Conditions Using Low-Level Knowledge Transfer With DNN," in IEEE Transactions on Automation Science and Engineering, vol. 18, no. 4, pp. 1983-1997, Oct. 2021, doi: 10.1109/TASE.2020.3028151.

[27] C. Cichy and S. Rass, 'An overview of data quality frameworks', *IEEE Access*, vol. 7, no. 1, pp. 24634–24648, 2019, doi: 10.1109/ACCESS.2019.2899751.

[28] H. T. Moges, V. Van Vlasselaer, W. Lemahieu, and B. Baesens, 'Determining the use of data quality metadata (DQM) for decision making purposes and its impact on decision outcomes - An exploratory study', *Decis. Support Syst.*, vol. 83, pp. 32–46, 2016.

[29] B. Heinrich and M. Klier, 'Metric-based data quality assessment - Developing and evaluating a probability-based currency metric', *Decis. Support Syst.*, vol. 72, pp. 82–96, 2015.

[30] J. Rabcan, P. Rusnak, E. Zaitseva, D. Macekova, M. Kvassay, and I. Sotakova, 'Analysis of data reliability based on importance analysis', *2019 International Conference on Information and Digital Technologies (IDT)*, 2019, pp. 402-408, doi: 10.1109/DT.2019.8813668.

[31] A. Immonen, P. Pääkkönen, and E. Ovaska, 'Evaluating the quality of social media data in big data architecture', *IEEE Access*, vol. 3, pp. 2028–2043, 2015.

[32] H. Liu, F. Huang, H. Li, W. Liu, and T. Wang, 'A big data framework for electric power data quality assessment', *2017 14th Web Information Systems and Applications Conference (WISA)* 2017, pp. 289–292, doi: 10.1109/WISA.2017.29.

[33] I. M. Prinsloo, J. N. du Plessis, and J. C. Vosloo, 'Improving data management of a mobile data collection system by using web-services', *28th Annual Southern African Institute for Industrial Engineering Conference (SAIIE28)*, 2017, pp. 3394.1–3394-12, https://www.saiie.co.za/system/files/2021-11/SAIIE28%20Proceedings.pdf.

[34] S. van Jaarsveld, S. W. van Heerden, and J. F. van Rensburg, 'Development of a condition monitoring information system for deep level mines', *28th Annual Southern African Institute for Industrial Engineering Conference (SAIIE28)*, 2017, pp. 3422.1–3422.13, https://www.saiie.co.za/system/files/2021-11/SAIIE28%20Proceedings.pdf

[35] P. Goosen, M. J. Mathews, and J. C. Vosloo, 'Automated electricity bill analysis in South Africa', *South African J. Ind. Eng.*, vol. 148, pp. 148–162, 2017.

[36] J. A. Deysel, M. Kleingeld, and C. J. R. Kriel, 'DSM strategies to reduce electricity costs on platinum mines', *2015 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2015, pp. 89-96, doi: 10.1109/ICUE.2015.7280252.

[37] W. Conradie, M. Kleingeld, and F. G. Jansen Van Rensburg, 'Development of a model to predict cost savings of reconfigured mine water reticulation systems', *2018 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2018, pp. 1–6, https://ieeexplore.ieee.org/document/8636614.

[38] W. Hamer, 'Analysing electricity cost saving opportunities on South African gold processing plants', MEng (Mechanical Engineering) dissertation, North-West University, 2015.

[39] B. Pascoe, H. J. Groenewald, and M. Kleingeld, 'Improving mine compressed air network efficiency through demand and supply control', *2017 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2017, pp. 1-5, doi: 10.23919/ICUE.2017.8067992.

[40] J. A. Stols, C. Cilliers, and J. F. van Rensburg, 'Implementing a remote condition monitoring system for South African gold mines', *29th Annual SAIIE Conference (SAIIE29): Steering the 4th Industrial Revolution*, Stellenbosch, 2018, pp. 3760.1–3760.15, https://conferences.sun.ac.za/index.php/saiie29/saiie29/paper/viewFile/3760/546.

[41] S. van Jaarsveld, W. van Blerk, J. H. Marais, and P. Goosen, 'Continuous evaluation of operational risks on deep-level mine equipment', *29th Annual SAIIE Conference (SAIIE29): Steering the 4th Industrial Revolution*, Stellenbosch, 2018, pp. 3604.1–3604.12, https://conferences.sun.ac.za/index.php/saiie29/saiie29/paper/viewFile/3604/524.

[42] I. Zaman, M. R. Barzegaran and O. A. Mohammed, "Condition Monitoring of Electric Components Using 3-D Printed Multiple Magnetic Coil Antennas," in IEEE Transactions on Magnetics, vol. 53, no. 6, pp. 1-4, June 2017, Art no. 6201404, doi: 10.1109/TMAG.2017.2664725.

[43] M. Madhikermi, A. Buda, B. Dave, and K. Främling, 'Key data quality pitfalls for condition based maintenance', *2017 2nd International Conference on System Reliability and Safety, (ICSRS) 2017*, pp. 474–480, doi: 10.1109/ICSRS.2017.8272868.

[44] S. Turrin, S. Subbiah, G. Leone, and L. Cristaldi, 'An algorithm for data-driven prognostics based on statistical analysis of condition monitoring data on a fleet level', *2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, 2015, pp. 629-634, doi: 10.1109/I2MTC.2015.7151341.

[45] J.-H. Shin and H.-B. Jun, 'On condition based maintenance policy', *J. Comput. Des. Eng.*, vol. 2, no. 2, pp. 119–127, 2015.

[46] A. K. S. Jardine, D. Lin, and D. Banjevic, 'A review on machinery diagnostics and prognostics implementing condition-based maintenance', *Mech. Syst. Signal Process.*, vol. 20, no. 7, pp. 1483–1510, 2006.

[47] S. van Jaarsveld, 'Developing an integrated information system to assess the operational condition of deep level mine equipment', DPhil (Computer and Electronic Engineering), North-West University, Potchefstroom Campus, 2018.

[48] B. Chindondondo, L. Nyanga, A. Van der Merwe, T. Mupinga, and S. Mhlanga, 'Development of a condition based maintenance system for a sugar producing company', *SAIIE26 Proceedings*, 2014, pp. 1–14. http://hdl.handle.net/10019.1/102521

[49] X. Xu, Y. Lei, and X. Zhou, 'A LOF-based method for abnormal segment detection in machinery condition monitoring', *2018 Prognostics and System Health Management Conference (PHM-Chongqing)*, 2018, pp. 125-128, doi: 10.1109/PHM-Chongqing.2018.00027. i

[50] R. Ajith, A. Tewari, D. Gupta and S. Tallur, "Low-Cost Vibration Sensor for Condition-Based Monitoring Manufactured From Polyurethane Foam," in IEEE Sensors Letters, vol. 1, no. 6, pp. 1-4, Dec. 2017, Art no. 6001504, doi: 10.1109/LSENS.2017.2773652.

[51] S. van Jaarsveld, J. N. du Plessis, and R. Pelzer, 'A control system for the efficient operation of bulk air coolers on a mine', *2015 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2015, pp. 133-137, doi: 10.1109/ICUE.2015.7280259.

[52] A. J. Schutte, M. Kleingeld, and H. J. Groenewald, 'A holistic energy-cost evaluation of ice vs chilled water in mining', *2016 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2016, pp. 118-122, https://ieeexplore.ieee.org/document/7605625.

[53] D. C. Mazur, J. A. Kay, and K. D. Mazur, 'Advancements in vibration monitoring for the mining industry', *IEEE Transactions on Industry Applications,* 2015, *Trans. Ind. Appl.*, vol. 51, no. 5, pp. 4321–4328.

[54] H. J. Groenewald, M. Kleingeld, and G. J. Cloete, 'An autoregressive fault model for condition monitoring of electrical machines in deep-level mines', *2018 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2018, pp. 1-6.

[55] J. Engles, S. Van Jaarsveld, and S. W. Van Heerden, 'Cost effective control of a platinum mine cooling system using combined DSM strategies', *2016 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2016, pp. 128-132.

[56] B. G. G. Terblanche, J. F. Van Rensburg, and W. Biermann, 'Improving deep-level mining refrigeration through increasing pre-cooling efficiency', *2018 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2018, pp. 1-5.

[57] J. L. Buys, M. Kleingeld, and C. Cilliers, 'Optimising the refrigeration and cooling system of a platinum mine', *2015 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2015, pp. 36-43, doi: 10.1109/ICUE.2015.7280244.

[58] M. H. P. Van Niekerk, S. W. Van Heerden, and J. F. Van Rensburg, 'The implementation of a dynamic air compressor selector system in mines', *2015 International Conference on the Industrial and Commercial Use of Energy (ICUE),* 2015, pp. 129-132, doi: 10.1109/ICUE.2015.7280258.

[59] G. -Y. Kwon and Y. -J. Shin, "Condition Monitoring Technique of HTS Cable via Tangent Distance-Based Template Matching Coefficient," in IEEE Transactions on Applied Superconductivity, vol. 31, no. 5, pp. 1-5, Aug. 2021, Art no. 4800305, doi: 10.1109/TASC.2021.3057032.

[60] G. Wang, M. Nixon and M. Boudreaux, "Toward Cloud-Assisted Industrial IoT Platform for Large-Scale Continuous Condition Monitoring," in Proceedings of the IEEE, vol. 107, no. 6, pp. 1193-1205, June 2019, doi: 10.1109/JPROC.2019.2914021.

[61] W. Hamer, 'A practical approach to quantify RSA Section 12L EE tax incentives for large industry', MEng (Mechanical Engineering) dissertation, North-West University, 2016.

[62] S. Zhang, W. Yao, P. Sun, and Y. Zhang, 'A condition monitoring data cleaning method for power equipment based on correlation analysis and ensemble learning', *2020 IEEE International Conference on High Voltage Engineering and Application (ICHVE), 2020, pp. 1-4, doi: 10.1109/ICHVE49031.2020.9279409*.

[63] A. G. S. Gous, W. Booysen, and W. Hamer, 'Data quality evaluation for measurement and verification processes', *2016 International Conference on the Industrial and Commercial Use of Energy (ICUE)*, 2016, pp. 9-15.

[64] W. Hamer, W. Booysen, and E. H. Mathews, 'A data quality evaluation framework for industrial energy efficiency reporting', *Steering the 4th Industrial Revolution*, *29th SAIIE Conference,* 2018, 16 pp. https://conferences.sun.ac.za/index.php/saiie29/saiie29/paper/view/3587.

[65] A. Lazar, L. Jin, C. A. Spurlock, K. Wu, and A. Sim, 'Data quality challenges with missing values and mixed types in joint sequence analysis', *2017 IEEE International Conference on Big Data (Big Data) 2017*, pp. 2620-2627, doi: 10.1109/BigData.2017.8258222.

[66] J. Y. Song, Y. P. Guo, S. J. Zhang, and J. D. Hao, 'Data integrity assessment method based on data dependence', *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, *2018*, pp. 1–9, doi: 10.1109/CISP-BMEI.2018.8633205.

[67] A. Juneja and N. N. Das, 'Big data quality framework: pre-processing data in weather monitoring application', *International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Perspectives and Prospects (COMITCon),2019*. IEEE, pp. 559–563, doi: 10.1109/COMITCon.2019.8862267.

[68] M. A. Serhani, H. T. El Kassabi, I. Taleb, and A. Nujum, 'An hybrid approach to quality evaluation across big data value chain', *2016 IEEE International Congress on Big Data (BigData Congress)*, 2016, pp. 418-425, doi: 10.1109/BigDataCongress.2016.65.

[69] A. F. Haryadi, J. Hulstijn, A. Wahyudi, H. Van Der Voort, and M. Janssen, 'Antecedents of big data quality: An empirical examination in financial service organizations', *2016 IEEE International Conference on Big Data (Big Data)*, 2016, pp. 116-121, doi: 10.1109/BigData.2016.7840595.

[70] M. M. J. Ferney, L. Beltran Nicolas Estefan, and V. V. J. Alexander, 'Assessing data quality in open data: A case study', *2017 Congreso Internacional de Innovacion y Tendencias en Ingenieria (CONIITI)*, 2017, pp. 1-5, doi: 10.1109/CONIITI.2017.8273343.

[71] I. Taleb, H. T. E. Kassabi, M. A. Serhani, R. Dssouli, and C. Bouhaddioui, 'Big data quality: A quality dimensions evaluation', *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications,* 2016, pp. 759-765, doi: 10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.0122.

[72] I. Taleb, M. A. Serhani, and R. Dssouli, 'Big Data Quality: A Survey', *2018 IEEE International Congress on Big Data (BigData Congress)*, 2018, 8 pp, doi:10.1109/bigdatacongress.2018.00029.

[73] D. Lee, 'Big data quality assurance through data traceability: A case study of the National Standard Reference Data Program of Korea', *IEEE Access*, vol. 7, pp. 36294–36299, 2019.

[74] A. Z. Faroukhi, I. El Alaoui, Y. Gahi, and A. Amine, 'Big data value chain: A unified approach for integrated data quality and security', *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, 2020, pp. 1-8, doi: 10.1109/ICECOCS50124.2020.9314391.

[75] Tian Hongxun, Wang Honggang, Zhou Kun, Shi Mingtai, Li Haosong, Xu Zhongping *et al.*, 'Data quality assessment for on-line monitoring and measuring system of power quality based on big data and data provenance theory', *2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, 2018, pp. 248-252, doi: 10.1109/ICCCBDA.2018.8386521.

[76] H. H. Ahmed, 'Data quality assessment in the integration process of linked open data (LOD)', *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), 2017, pp. 1-6, doi: 10.1109/AICCSA.2017.178.*

[77] D. Rao, V. N. Gudivada, and V. V. Raghavan, 'Data quality issues in big data', *2015 IEEE International Conference on Big Data (Big Data), 2015, pp. 2654-2660, doi: 10.1109/BigData.2015.7364065.*

[78] S. D. Rahmawati and Y. Ruldeviyani, 'Data quality management strategy to improve the quality of worker's wage and income data: A case study in BPS-Statistics Indonesia, 2018', *2019 Fourth International Conference on Informatics and Computing (ICIC*), 2019, pp. 1-6, doi: 10.1109/ICIC47613.2019.8985803.

[79] S. Juddoo and C. George, 'Discovering most important data quality dimensions using latent semantic analysis', *2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD),* 2018, pp. 1-6, doi: 10.1109/ICABCD.2018.8465129.

[80] A. G. Labouseur and C. C. Matheus, 'Dynamic data quality for static blockchains', *019 IEEE 35th International Conference on Data Engineering Workshops (ICDEW)*, 2019, pp. 19-21, doi: 10.1109/ICDEW.2019.00-41.

[81] A. L. Nobles, K. Vilankar, H. Wu, and L. E. Barnes, 'Evaluation of data quality of multisite electronic health record data for secondary analysis', *2015 IEEE International Conference on Big Data (Big Data)*, 2015, pp. 2612-2620, doi: 10.1109/BigData.2015.7364060.

[82] K. Xiangwei, 'Evaluation of flight test data quality based on rough set theory', *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2020, pp. 1053-1057, doi: 10.1109/CISP-BMEI51763.2020.9263667.

[83] S. Kim, R. Pérez-Castillo, I. Caballero, J. Lee, C. Lee, D. Lee *et al.*, 'Extending data quality management for smart connected product operations', *IEEE Access*, vol. 7, pp. 144663–144678, 2019, doi: 10.1109/ACCESS.2019.2945124.

[84] K. Hee, 'Is data quality enough for a clinical decision?: Apply machine learning and avoid bias', *2017 IEEE International Conference on Big Data (Big Data)*, 2017, pp. 2612-2619, doi: 10.1109/BigData.2017.8258221.

[85] S. Juddoo, 'Overview of data quality challenges in the context of Big Data', *2015 International Conference on Computing, Communication and Security (ICCCS)*, 2015, pp. 1-9, doi: 10.1109/CCCS.2015.7374131.

[86] S. Loetpipatwanich and P. Vichitthamaros, 'Sakdas: A Python package for data profiling and data quality auditing', *2020 1st International Conference on Big Data Analytics and Practices (IBDAP),* 2020, pp. 1-4.

[87] Xiao HongJu, Wang Fei, Wang FenMei, Wang XiuZhen, 'Some key problems of data management in army data engineering based on big data', *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA),* 2017, pp. 149-152, doi: 10.1109/ICBDA.2017.8078796.

[88] N. Micic, D. Neagu, F. Campean, and E. H. Zadeh, 'Towards a data quality framework for heterogeneous data', *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2017, pp. 155-162, doi: 10.1109/iThings-GreenCom-CPSCom-SmartData.2017.28.

[89] Qian. Fu and J. M. Easton, 'Understanding data quality: Ensuring data quality by design in the rail industry', *2017 IEEE International Conference on Big Data (Big Data)*, 2017, pp. 3792–3799, doi: 10.1109/BigData.2017.8258380.

[90] J. Byabazaire, G. O'Hare, and D. Delaney, 'Using trust as a measure to derive data quality in data shared IoT deployments', *2020 29th International Conference on Computer Communications and Networks (ICCCN)*, 2020, pp. 1-9.

[91] O. Kwon, N. Lee, and B. Shin, 'Data quality management, data usage experience and acquisition intention of big data analytics', *Int. J. Inf. Manage.*, vol. 34, no. 3, pp. 387–394, 2014.

[92] F. Fox, V. R. Aggarwal, H. Whelton, and O. Johnson, 'A data quality framework for process mining of electronic health record data', *2018 IEEE International Conference on Healthcare Informatics (ICHI)*, 2018, pp. 12-21, doi: 10.1109/ICHI.2018.00009.

[93] Tao Dai, Hongpu Hu, Yanli Wan, Quan Chen and Yan Wang, 'A data quality management and control framework and model for health decision support'; *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, 2015, pp. 1792-1796, doi: 10.1109/FSKD.2015.7382218.

[94] G. Zhang, 'A data traceability method to improve data quality in a big data environment', *2020 IEEE Fifth International Conference on Data Science in Cyberspace (DSC)*, 2020, pp. 290-294, doi: 10.1109/DSC50466.2020.00051.

Appendices

Appendix A: Publication

An academic article was published in the South African Journal of Industrial Engineering (SAJIE) on 29 November 2021. The final accepted manuscript is appended below and also available online at https://doi.org/10.7166/32-3-2625.

South African Journal of Industrial Engineering November 2021 Vol 32(3) Special Edition, pp 173-184

ESTIMATING THE RELIABILITY OF CONDITION-BASED MAINTENANCE DATA USING CONTEXTUAL MACHINE-SPECIFIC CHARACTERISTICS

J.N. de Meyer[1]*, P. Goosen[1], J.F. van Rensburg[1], J.N. du Plessis[2] & J.H. van Laar[2]

ARTICLE INFO

Article details
Presented at the 32nd annual conference of the Southern African Institute for Industrial Engineering (SAIIE), held from 4-6 October 2021 in Muldersdrift, South Africa.

Available online 29 Nov 2021

Contact details
* Corresponding author
jdemeyer@rems2.com

Author affiliations
1 Centre for Research and Continued Engineering Development (CRCED), North-West University, South Africa

2 Department of Industrial Engineering, Stellenbosch University, South Africa

ORCID® identifiers
J.N. de Meyer
https://orcid.org/0000-0002-9044-4523

P. Goosen
https://orcid.org/0000-0002-5744-5268

J.F. van Rensburg
https://orcid.org/0000-0002-8246-3396

J.N. du Plessis
https://orcid.org/0000-0002-7080-726X

J.H. van Laar
https://orcid.org/0000-0003-0457-328X

DOI
http://dx.doi.org/10.7166/32-3-2625

ABSTRACT

In the mining industry, inter-connected machinery operates under harsh conditions 24 hours a day. Naturally, this degrades their state, and can lead to premature breakdowns and production losses. Condition-based maintenance (CBM) is a strategy that plans maintenance schedules depending on the condition of the equipment, and aims to improve decision-making processes. Data collected from machinery for CBM purposes must be reliable to avoid negative impacts on the maintenance strategy. Data reliability can be estimated by comparing multiple data streams; however, they are not always available, and can be expensive. This study aims to estimate the isolated and contextual reliability of single-source CBM data by applying multiple data analytics techniques. An application is designed to analyse current data on a machine level and to determine combined reliability. A case study implementation shows the difference in reliability classification accuracy between the isolated and contextual methods, highlighting the need for them to be combined.

OPSOMMING

In die mynbedryf word komplekse masjienstelsels in ongewenste omstandighede 24 ure per dag bedryf. Dit veroorsaak die agteruitgang van hul toestand en kan lei tot stelsels wat vroegtydig onklaar raak en daaropvolgende produksie verliese. Toestandsgebasseerde onderhoud (TGO) is 'n strategie wat onderhoudskedules beplan afhangende van die toestand van die masjien en beoog om besluitnemingsprosesse te verbeter. Data opgeneem van masjiene vir TGO doeleindes moet betroubaar wees om die negatiewe gevolge op onderhoudskedules te vermy. Data betroubaarheid kan geskat word deur verskeie databronne te vergelyk, maar menigte bronne is nie altyd beskikbaar nie en kan duur wees om te bekom. Hierdie studie poog om die geïsoleerde en kontekstuele betroubaarheid van enkelbron TGO data te skat deur gebruik te maak van verskeie data analise tegnieke. 'n Sagteware program word ontwerp om data te ontleed op 'n masjien vlak en die betroubaarheid daarvan te bepaal. 'n Gevallestudie wys die verskil in betroubaarheidsuitspraak akkurraatheid tussen geïsoleerde en kontekstuele metodes en lig die behoefte uit om die metodes te kombineer.

1 INTRODUCTION

In the mining industry, inter-connected machinery operates under harsh conditions 24 hours a day [1], [2]. These conditions degrade the state of the machinery, and can lead to premature breakdowns and production losses [2].

To avoid unnecessary downtime, intermittent maintenance is performed on the machines to keep them in an operational state [3]. However, inefficient maintenance strategies can also have negative consequences, such as unnecessary downtime and/or delayed maintenance [1]. To minimise the negative impact, maintenance schedules should be optimised to maximise the benefits [4].

Condition-based maintenance (CBM) is a strategy that plans maintenance schedules, depending on the condition of the equipment, and that aims to improve decision-making processes [5]. For CBM to be effective, operational data from the machine is required [5].

Data can be collected using condition monitoring, which involves adding sensors to equipment in order to measure the vital characteristics of the machine [2], [6]. The more reliable the measured data, the more confidence it instils in the decision [7]-[11].

Data reliability can be estimated in various ways [12]. A simple yet efficient method is to compare the data for a specific characteristic from multiple sources. Multiple data streams can be set up by installing more sensors or by making use of secondary data sources, such as third-party measurements or reports [13].

Unfortunately, these additional data streams are not always available, and can be expensive to set up; as a result, single-source data streams are mostly used. The reliability of single-source CBM data can be estimated in two ways: in isolation, or in context [14].

Isolated reliability implies that the data streams are evaluated individually, disregarding external influences [15]. This is useful for gauging whether an installed sensor is calibrated correctly.

By contrast, contextual reliability is calculated on a machine level by considering other data streams [15]. This provides a broader picture of the operational conditions at a machine level. One such example is a machine that is switched off but still incorrectly shows, from the data, that it is drawing electrical current. Contextual reliability can be used to gauge whether the data approved by isolated reliability makes sense in the bigger picture.

Despite the rise in research related to data reliability over the past few years, only limited research has focused on single-source CBM data reliability using contextual methods in the mining industry. This study aims to reduce the knowledge gap by estimating the isolated and contextual reliability of single-source CBM data by applying multiple data analytics techniques to mining systems.

2 RELATED WORK

The amount of CBM data available to companies has increased dramatically over the past few years [2]. With this increase in volume, companies are emphasising the reliability of the data [8]. Two studies that investigated the quality of CBM data were identified as a starting point for this study.

Goosen [12] investigated the quality of industrial data using various methods. From the investigation, a system was created to estimate a quality score for each data point. The study stated that contextual knowledge could increase the accuracy of the data quality calculations. Similar methods could be applied to estimate the reliability of CBM data in an isolated manner.

De Meyer [16] implemented a system to calculate the integrity of CBM data in a contextual manner. The system took a contextual-isolated approach, as the data streams were evaluated in isolation, but with contextual information about the stream itself. This approach can be applied to expand on the isolated reliability calculations to cater for each data stream.

To estimate the isolated reliability, this study will use the metrics identified by Goosen [12], and will include contextual information from each stream, similar to the approach of de Meyer [16]. This study will also include contextual reliability methods and combine the results of both approaches to estimate the overall reliability of the data, as suggested by both Goosen and de Meyer.

This study proposes a system for estimating the reliability of single-source CBM data by making use of the combination of the isolated and contextual reliability methods.

3 METHODOLOGY

A simplified organisational structure, depicted in Figure 1, is used throughout this paper to represent mines generically.

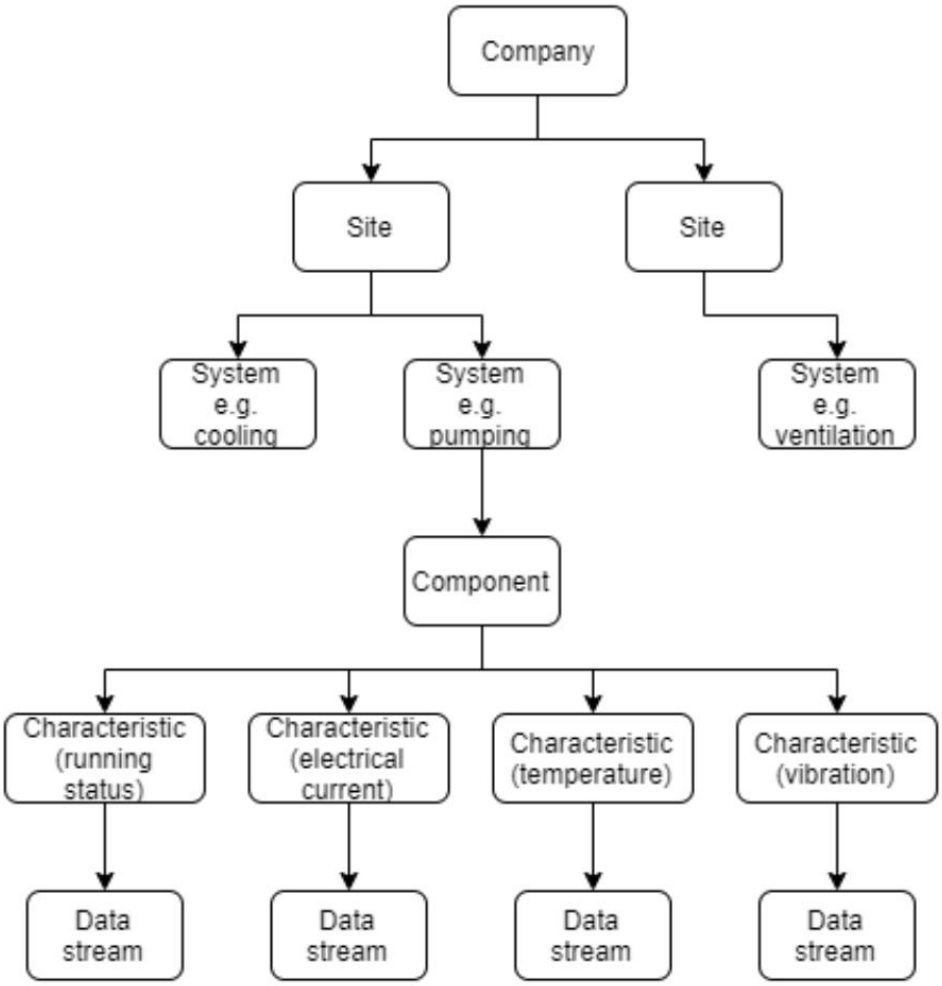

Figure 1: Diagram illustrating the organisational structure used for the study

Figure 1 illustrates the terminology used in this paper, and how the different entities relate to one another. A company is the highest level of the organisation. Each company can have multiple sites that are located throughout the world. Every site categorises its machinery into systems with a main focus — e.g., cooling, pumping, ventilation.

Each system can have multiple components, or machines. Each component has various characteristics that are monitored. Depending on the component and monitoring technique, the monitored characteristics can include vibration, sound, temperature, chemicals and particles released into the environment, and physical effects. [17]. For the purposes of this paper, these characteristics are limited to the running status, electrical current, temperature, and vibration of a machine. Each of these characteristics is monitored, and their measurements are associated with a data stream.

3.1 Data collection

The CBM data is recorded at the mine and sent through a transmission channel to a database. The transmission channel is illustrated in Figure 2.

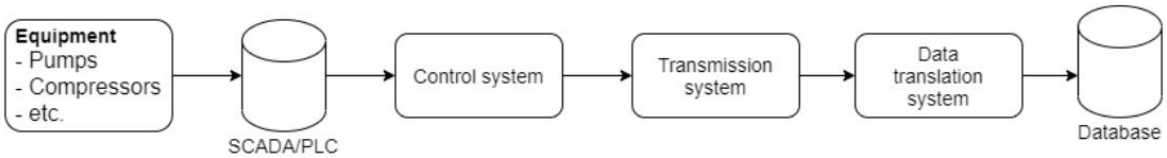

Figure 2: Diagram illustrating the transmission channel for data from machinery on site

Each component is monitored by a monitoring system such as a supervisory control and data acquisition (SCADA) system. The monitoring system sends the data to the control system that controls the components. The control system then sends the data to a transmission system that transfers the data over the internet to a translation system. This system translates the incoming data into a more readable format for the end-systems that will use the data before saving it into the database.

Each measurement has two main properties; the timestamp of when the measurement is taken, and the value. For this paper, half-hourly data is used, as this is considered a common measurement interval, and reduces the required storage in the database while maintaining enough information to reflect the operation of the machine accurately. When higher resolution data is available, the half-hourly average is used.

3.2 Machine characteristics

As mentioned earlier, four main characteristics that are used in the mining industry and are commonly monitored have been identified [17]. These four characteristics each have a data stream associated with them, and together they give an overview of the operational status of the machine at any point in time. These characteristics are running status, electrical current, temperature, and vibration, which are described in Table 1.

Table 1: Table of commonly monitored machine characteristics for CBM

Characteristic	Description	Unit
Running status	Describes whether the machine is switched on (1) or off (0), commonly connected to the on-off switch of the machine or controlled by programmable logic controllers (PLCs) or SCADA.	–
Electrical current	Represents the current drawn by the machine at any given point in time. The value range is specific to the component.	A
Temperature	Represents the temperature of the machine at a specific place at any given point in time. The values, similar to the electrical current, differ depending on the component and conditions.	°C
Vibration	The vibration experienced by the component at any given point in time. The vibration experienced by a component is the sum of the vibration generated by the component itself and the vibration in the environment caused by, for example, other components.	$mm \cdot s^{-1}$

3.3 Relationships

Each of the characteristics discussed earlier has a relationship with the other characteristics when evaluating them from a contextual point of view. When a component is running, it has to draw electrical current, its temperature should increase to a specific steady value, and it should vibrate more, as described in Equations 1—3 respectively:

$$E_{current} > 0, when\ R_{status} = 1 \tag{1}$$

where

- $E_{current}$ is the electrical current drawn, and
- R_{status} is the running status of the component.

$$T_{ambient} \leq T_{current} = t_{on} \times T_{gain} + T_{ambient} \leq T_{limit}, when\ R_{status} = 1 \tag{2}$$

where

- $T_{ambient}$ is the ambient temperature at the component,
- $T_{current}$ is the current temperature of the component,
- t_{on} is the duration for which the component has been running,
- T_{gain} is the rate at which the temperature of the component increases while in operation,
- T_{limit} is the safety limit of the component where it is switched off, and
- R_{status} is the running status of the component.

$$V_{environment} < V_{current} \leq V_{limit}, when\ R_{status} = 1 \tag{3}$$

where

- $V_{environment}$ is the vibration experienced by the component caused by external factors,
- $V_{current}$ is the current vibration measured on the component,
- V_{limit} is the safety limit of the component where it is switched off, and
- R_{status} is the running status of the component.

Similarly, when a component is not running, it cannot draw electrical current, it will start to cool down, and it will vibrate less, as described in Equations 4—6 respectively:

$$E_{current} = 0, when\ R_{status} = 0 \tag{4}$$

where

- $E_{current}$ is the electrical current drawn, and
- R_{status} is the running status of the component.

$$T_{ambient} \leq T_{current} = t_{off} \times T_{loss} + T_{ambient}, when\ R_{status} = 0 \tag{5}$$

where

- $T_{ambient}$ is the ambient temperature,
- $T_{current}$ is the current temperature of the component,
- t_{off} is the time the component has been switched off,
- T_{loss} is the rate at which the component cools down, and
- R_{status} is the running state of the component.

$$V_{environment} \geq V_{current}, when\ R_{status} = 0 \tag{6}$$

where

- $V_{environment}$ is the vibration experienced by the component from external sources,
- $V_{current}$ is the current vibration of the component, and
- R_{status} is the running state of the component.

Applying Newton's law for the conservation of energy to the component, it can be assumed that, when the component draws electrical current, it will convert some of it into heat and kinetic energy, resulting in a rise in temperature and increase in vibration, as described by Equations 7—8:

$$T_{ambient} \leq T_{current} = t_{energy\ consumed} \times T_{gain} + T_{ambient}, when\ E_{current} > 0 \tag{7}$$

where

- $T_{ambient}$ is the ambient temperature,
- $T_{current}$ is the current temperature of the component,
- $t_{energy\ consumed}$ is the time for which the component has been drawing electrical current,
- T_{gain} is the rate at which the temperature of the component increases when in operation, and
- $E_{current}$ is the electrical current drawn by the component.

$$V_{environment} < V_{current} \leq V_{limit}, when\ E_{current} > 0 \tag{8}$$

where

- $V_{environment}$ is the vibration experienced by the component, caused by external factors,
- $V_{current}$ is the current vibration of the component,
- V_{limit} is the safety limit where the component will be shut off, and
- $E_{current}$ is the electrical current drawn by the component.

Similarly, when a component stops drawing electrical current, it will eventually cool down and vibrate less, as described in Equations 9—10:

$$T_{ambient} \leq T_{current} = t_{last\ consumed} \times T_{loss} + T_{ambient}, when\ E_{current} = 0 \tag{9}$$

where

- $T_{ambient}$ is the ambient temperature,
- $T_{current}$ is the component temperature,
- $t_{last\ consumed}$ is the time since the component last drew electrical current,
- T_{loss} is the rate at which the component cools down when not in operation, and
- $E_{current}$ is the electrical current drawn by the component.

$$V_{environment} \geq V_{current}, when\ E_{current} = 0 \tag{10}$$

where

- $V_{environment}$ is the vibration experienced by the component from external factors,
- $V_{current}$ is the current vibration of the component, and
- $E_{current}$ is the electrical current drawn by the component.

From Equations 1—10, it can be seen that characteristics influence one another, which is important to consider when evaluating the reliability of the data from a contextual viewpoint.

3.4 System design

A system was designed to calculate the combined reliability of CBM data. The system flow is illustrated in Figure 3.

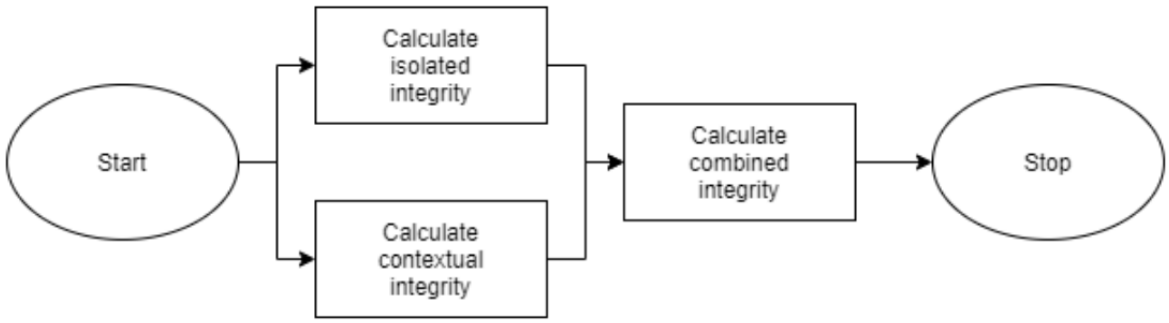

Figure 3: Diagram of the system flow to calculate the combined data integrity

From Figure 3, it is shown that the system calculates the isolated and contextual reliability of each data point. It then uses these results to determine the overall reliability.

3.4.1 Isolated reliability

The system calculates the isolated reliability of each data point using the flow described in Figure 4.

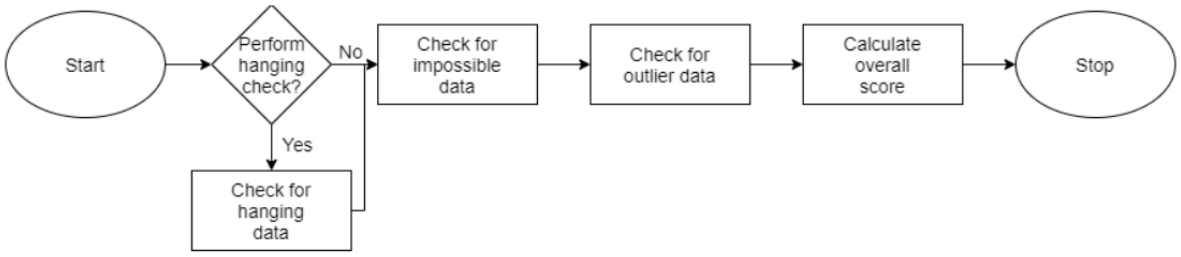

Figure 4: Diagram of the system flow for calculating isolated data integrity

The figure above shows how the system will calculate the data integrity for data points from an isolated perspective. This process consists of four stages, with the first being optional, depending on whether or not it is applicable to the data stream.

Hanging data implies a repeating pattern of the same value over an extended period of time. It usually indicates that a sensor is disconnected or broken. This metric check is optional, as it is not applicable to all characteristics. For this study, the running status characteristic will not be evaluated for hanging data, as the data only assumes two values: 0 or 1.

Impossible data are data points that are not possible within the context of the data stream. These include negative and extreme values that are outside the bounds of the component. These impossible values can be used to identify uncalibrated sensors.

Outlier data are data points that fall inside the bounds of possibility for the component, yet do not conform to the expected profile of the data. They can be used to identify malfunctioning sensors or interrupted measurement and/or transmission processes.

If a data point is flagged by any of the data checks, the data point is deemed unreliable from an isolated reliability perspective.

3.4.2 Contextual integrity

Following the calculation of the isolated reliability calculations, the data streams for each of the four characteristics of a machine are then evaluated from a contextual perspective, as shown in Figure 5.

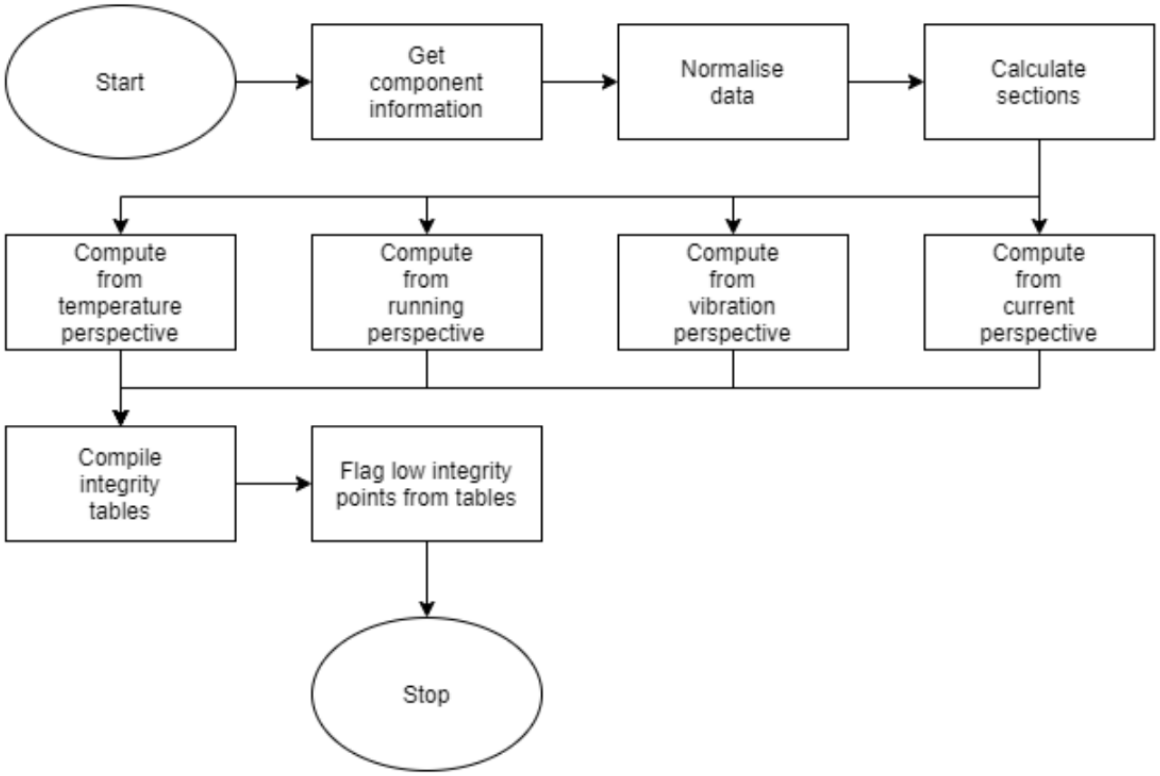

Figure 5: Diagram of the system flow for calculating contextual data integrity

Figure 5 illustrates the steps the system will take to calculate the contextual reliability of the data points.

First, the system will get the machine model information, which gives context to the data streams linked to the four main characteristics.

Second, the system will normalise all values for a specified range to a range of 0—1. This is done more easily to compare the different characteristics with one another according to the data trends. The unnormalised values are evaluated in the isolated integrity calculations, allowing the contextual calculations to focus on the trends.

Third, the system calculates each section per data stream. This involves breaking the profile into sections, based on whether the data has an increasing, decreasing, or flat trendline.

Using the relationships described in Equations 1—10, the system compares each characteristic's data with the data of the other three characteristics, and compiles a truth table, as illustrated in Table 2.

Table 2: Truth table used for contextual data reliability calculations

		Characteristic evaluated			
		Running	Current	Temperature	Vibration
Characteristic perspective	Running	-	T/F	T/F	T/F
	Current	T/F	-	T/F	T/F
	Temperature	T/F	T/F	-	T/F
	Vibration	T/F	T/F	T/F	-

The truth table compares the reliability of each characteristic from the viewpoint of a different characteristic. From this table, the reliability of the data point is calculated using Equation 11:

$$R_c = \frac{S_{received}}{S_{given}} \geq 0.6, where\ S_{received} \geq 2\ and\ S_{given} \geq 2\ and\ S_{given} \neq 0 \quad (11)$$

where

- R_c is the contextual reliability of the data point,
- $S_{received}$ is the number of high reliability scores (T) for the characteristic awarded by the other three characteristics listed in the characteristic column, and
- S_{given} is the number of high reliability scores (T) given to the other three characteristics by this characteristic, as listed in the characteristic row.

For a data point to be seen as reliable, it should consider other data points to be reliable, and should be considered as reliable by other data points. By considering the reliability of a data point from both perspectives, only data points that fit into the context of the component should be classified as reliable. To this end, the ratio between the scores received and given should be greater than 60 percent. Both scores should also be at least 2 to try to reduce the number of falsely classified high-reliability data points.

3.4.3 Combined integrity

The combined reliability of each data point is the sum of its isolated and contextual reliability scores, as illustrated by Equation 12:

$$0 \leq R_T = \frac{(R_I + R_C)}{2} \leq 1 \quad (12)$$

where

- R_T is the combined reliability of a data point,
- R_I is the isolated reliability, and
- R_C is the contextual reliability.

For a data point to be considered reliable, R_T should have a value of 1. If the value is less than 1, it will be rounded down to 0 and will be considered unreliable.

3.5 Verification

The system was verified using a testing data set with erroneous values to ensure that the system correctly classified the reliability of at least 95 percent of the data points. The results of the testing data set were manually reviewed to ensure that the system was working as expected.

An erroneous dataset was manually selected for two days' worth of data for a compressor. This dataset was deemed a sufficient representation of the compressor's operation, as it contained the following common operation cycles:

- extended running periods,
- extended periods of being switched off,
- periods in which the compressor was switched on and off, and
- periods with data loss.

The dataset was examined manually, and the low reliability data points from both an isolated and a contextual perspective were classified. This pre-classified dataset would enable the calculation of the system's accuracy. The results of the verification dataset are shown in Figure 6.

As seen in Figure 6, the system was able to classify 376 of the 384 data points correctly, resulting in an accuracy of 97.92 percent. The system incorrectly classified data points from the contextual perspective for eight data points over three of the characteristics. All eight of these data points were shared timestamps, suggesting that the system was not fully calibrated to the component.

On further investigation it was found that the system was incorrectly classifying data points at the start of steep gradients. To correct this issue, the system was re-calibrated by adjusting the contextual parameters for the component.

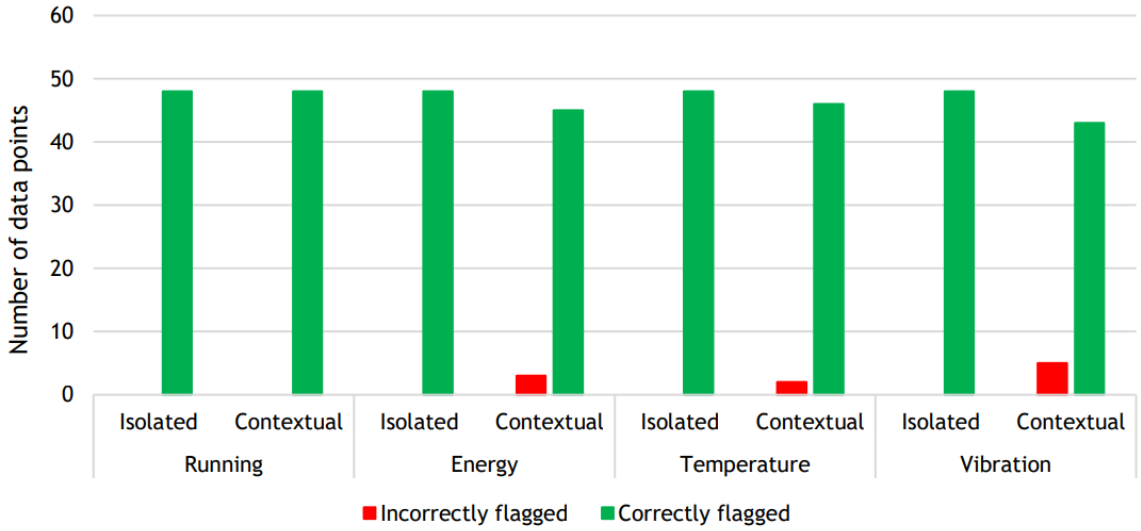

Figure 6: Bar chart displaying the results of the verification test

4 IMPLEMENTATION AND RESULTS

The system was applied to a case study compressor for a month's data. The dataset included data for each of the four characteristics — running status, electrical current, temperature, and vibration. In the dataset, each of the data streams had instances of low-reliability data points from both isolated and contextual perspectives.

Similar to the verification, each of the data points in the dataset was manually classified to calculate the system accuracy. After implementing the system, the electrical current data stream was classified with low reliability over the majority of the dataset. On further investigation, it was found that the incorrect data stream was linked to the component. After correcting the configuration error, the system was applied again, and obtained the results shown in Figure 7. From Figure 7, it can be seen that the system performed well overall.

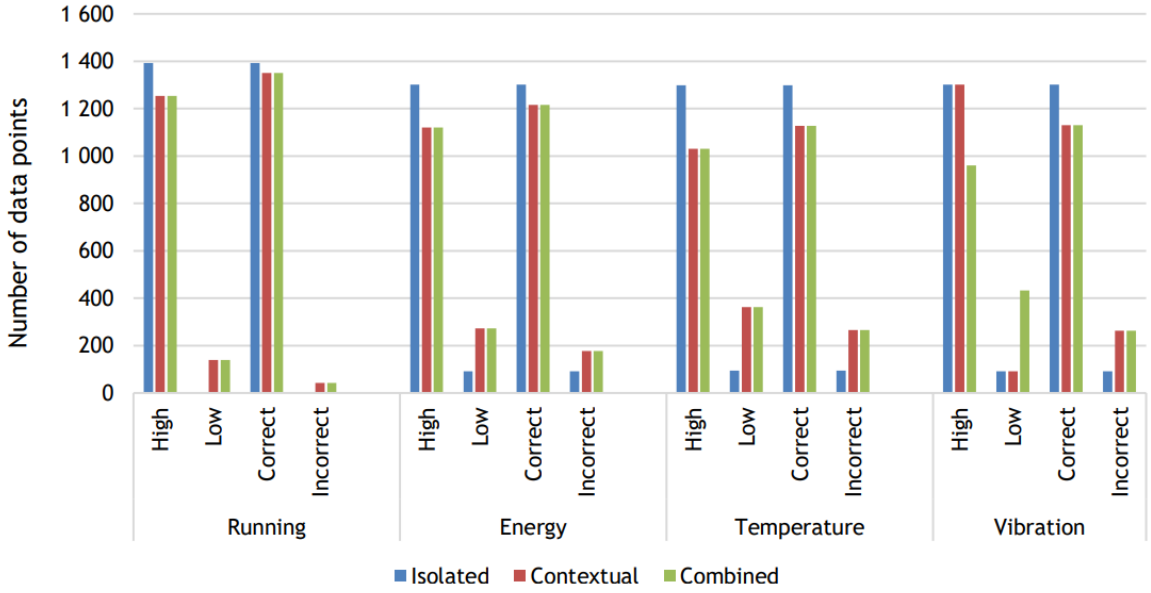

Figure 7: Bar chart displaying results for the case study compressor

181

The system experienced the most difficulty with classifying the temperature and vibration characteristics. This was surprising, as the relationship between these two characteristics initially seemed to be the weakest. From the results, however, it would seem that this relationship played a large role in the contextual calculations, as the incorrectly classified temperature data points correlated with the incorrectly classified vibration data points. In other words, there is a high likelihood that when a temperature data point is incorrectly classified, the corresponding vibration data point will also be incorrectly classified.

From the above chart it can also be seen that neither the isolated nor the contextual reliability methods were flawless in their classification, with the isolated reliability methods being more accurate. This is to be expected, as the isolated reliability methods only consider a single data stream, leaving less room for incorrect classification owing to the reduced variables involved.

A summary of the case study's implementation is shown in Figure 8.

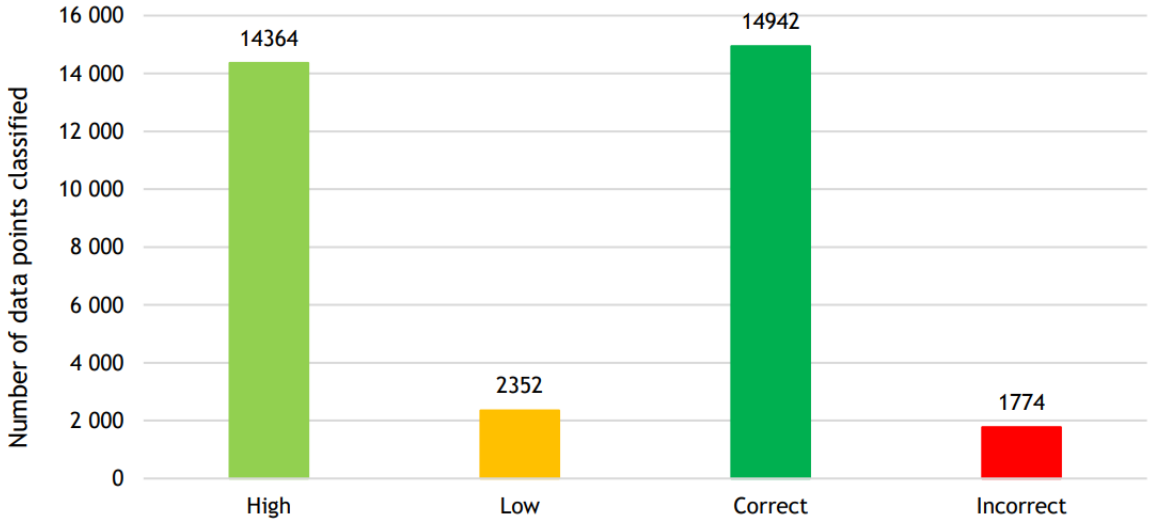

Figure 8: Bar chart indicating high-level results of case study implementation

As is evident from Figure 8, the majority of the data points were of a high reliability. Interestingly, all of the data points that were incorrectly classified were wrongly classified as high reliability points — i.e., false negatives. The system was able to classify 89.39 percent of the data points correctly — considerably less than the percentage achieved during the verification. This could be attributed to the small size of the testing data set compared with the case study data set, as the system was calibrated to the smaller set. This smaller set did not have a comprehensive amount of variety in the data, causing the system to be calibrated too finely. This can be seen as similar to over- or under-training a model.

As stated earlier, the isolated reliability methods classified the data points more accurately. However, when trying to gain insight into the component as a whole, the contextual reliability played a far larger role. In the first attempt to implement the case study, the electrical current data stream was incorrectly linked to the case study component; but the isolated reliability methods did not raise any red flags for this data stream.

However, the contextual reliability methods classified the majority of the data as unreliable, and the overall reliability was classified as low. On further investigation, the incorrect component configuration was discovered. Although the contextual reliability methods did not perform as well as the isolated reliability methods, they were crucial in avoiding false negatives over the majority of the data set. Thus the combination of isolated and contextual reliability methods produced a more reliable result than what would have been produced if the methods had been implemented separately.

5 CONCLUSION AND RECOMMENDATIONS

Machinery in the mining industry endures harsh operating conditions. To ensure an extended lifetime, efficient maintenance strategies such as condition-based maintenance should be implemented. To get the maximum benefit from the strategies, the data used for decision-making should be reliable.

This paper presented a method for estimating the reliability of single-source condition-based maintenance data by making use of isolated and contextual reliability calculations. A system was created, verified, and implemented on a case study. There was an approximate 10 percent difference in accuracy between a testing data set and the case study results. This could be attributed to the small testing data set, which in effect under-trained the system.

Although the isolated reliability methods produced more accurate results in the case study, they have their limitations. Incorrectly configured data streams for components are not identified by the isolated reliability methods, which produces false negatives. Using the combination of isolated and contextual reliability methods will ultimately produce a more accurate classification of the data streams over a large dataset than the two methods individually.
Although the system was accurate during the case study, there is room for improvement. When calibrating the system, a larger data set should be used to ensure that the system is capable of handling most data situations. The relationship between temperature and vibration should be revisited to ensure that this relationship is thoroughly catered for in the system.

An edge case arose during the case study, in which the temperature rose for a period whenever the component was shut off. The initial design did not cater for this with the simple relationship equations between the temperature and the running status (Equations 2 & 5) and between the temperature and the electrical current drawn (Equations 7 & 9). This rise in temperature could most likely be attributed to Newton's law of energy conservation, in which the rotational energy, along with friction, was converted to heat. Once the rotational energy of the component had been depleted, the component started to cool down.

Currently the system only classifies the reliability of the data points. Future work could include using these reliability results to try to identify and classify the events that cause the low reliability.

By implementing the proposed system on mines, unnecessary maintenance time could be reduced. This would result in a reduction of wasted capital and maintenance hours and an increase in the machinery's uptime, and could lead to increased production. Ultimately, implementation of the system will result in a reduction of wasted time, as decisions made using reliable data will be more reliable.

REFERENCES

[1] S. van Jaarsveld, "Developing an integrated information system to assess the operational condition of deep level mine equipment", PhD dissertation, Dept. Compt. & Elect. Eng., NWU, Potchefstroom, 2018.

[2] X. Xu, Y. Lei and X. Zhou, "A LOF-based method for abnormal segment detection in machinery condition monitoring", *2018 Prognostics and System Health Management Conference (PHM-Chongqing)*, 2018, pp. 125—128, doi: 10.1109/PHM-Chongqing.2018.00027.

[3] S. Jong-Ho and J. Hong-Bae, "On condition based maintenance policy", *Journal of Computational Design and Engineering*, vol. 2, no. 2, pp. 119—127, Jan. 2015, doi: 10.1016/j.jcde.2014.12.006.

[4] S. Telford, M. Mazhar and I. Howard, "Condition based maintenance (CBM) in the oil and gas industry: An overview of methods and techniques", *International Conference on Industrial Engineering and Operations Management*, 2011, pp. 1152–1159.

[5] A.K.S. Jardine, D. Lin and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance", *Mechanical Systems and Signal Processing*, vol. 20, no. 7, Oct. 2006, pp. 1483—1510, doi: 10.1016/j.ymssp.2005.09.012.

[6] S. Turrin, S. Subbiah, G. Leone and L. Cristaldi, "An algorithm for data-driven prognostics based on statistical analysis of condition monitoring data on a fleet level", *2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, 2015, pp. 629—634, doi: 10.1109/I2MTC.2015.7151341.

[7] W. Hamer, "A practical approach to quantify RSA Section 12L EE tax incentives for large industry", *PhD dissertation*, Dept. Mech. Eng., NWU, 2016.

[8] C. Cichy and S. Rass, "An overview of data quality frameworks", *IEEE Access*, vol. 7, pp. 24634—24648, 2019, doi: 10.1109/ACCESS.2019.2899751.

[9] Y. Ishizuka, W. Chen and I. Paik, "Workflow transformation for real-time big data processing", *2016 IEEE International Congress on Big Data (BigData Congress)*, 2016, pp. 315—318, doi: 10.1109/BigDataCongress.2016.47.

[10] J. Rabcan, P. Rusnak, E. Zaitseva, D. Macekova, M. Kvassay and I. Sotakova, "Analysis of data reliability based on importance analysis", *2019 International Conference on Information and Digital Technologies (IDT)*, 2019, pp. 402–408, doi: 10.1109/DT.2019.8813668.
[11] S. Zhang, W. Yao, P. Sun and Y. Zhang, "A condition monitoring data cleaning method for power equipment based on correlation analysis and ensemble learning", *2020 IEEE International Conference on High Voltage Engineering and Application (ICHVE)*, 2020, pp. 1–4, doi: 10.1109/ICHVE49031.2020.9279409.
[12] A. Goosen, "A system to quantify industrial data quality", *MEng thesis*, Dept. Compt. & Elect. Eng., NWU, 2018.
[13] H. Liu, F. Huang, H. Li, W. Liu and T. Wang, "A big data framework for electric power data quality assessment", *2017 14th Web Information Systems and Applications Conference (WISA)*, 2017, pp. 289–292, doi: 10.1109/WISA.2017.29.
[14] A. Juneja and N.N. Das, "Big data quality framework: Pre-processing data in weather monitoring application", *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, 2019, pp. 559–563, doi: 10.1109/COMITCon.2019.8862267.
[15] A. Immonen, P. Pääkkönen and E. Ovaska, "Evaluating the quality of social media data in big data architecture", *IEEE Access*, vol. 3, pp. 2028–2043, 2015, doi: 10.1109/ACCESS.2015.2490723.
[16] J.N. de Meyer, "Validating the integrity of single source condition monitoring data", *MEng thesis*, Dept. Compt. & Elect. Eng., NWU, 2020.
[17] B. Chindondondo, L. Nyanga, A. van der Merwe, T. Mupinga and S. Mhlanga, "Development of a condition based maintenance system for a sugar producing company", *SAIIE26 Proceedings*, 2014, pp. 1–14.

Appendix B: Verification results

Clean Dataset 1

Figure 53 illustrates the unnormalised running-status data stream analysed by the system for Clean Dataset 1. As expected, the system correctly identified all data points as reliable from an intrinsic, contextual, and overall integrity perspective.

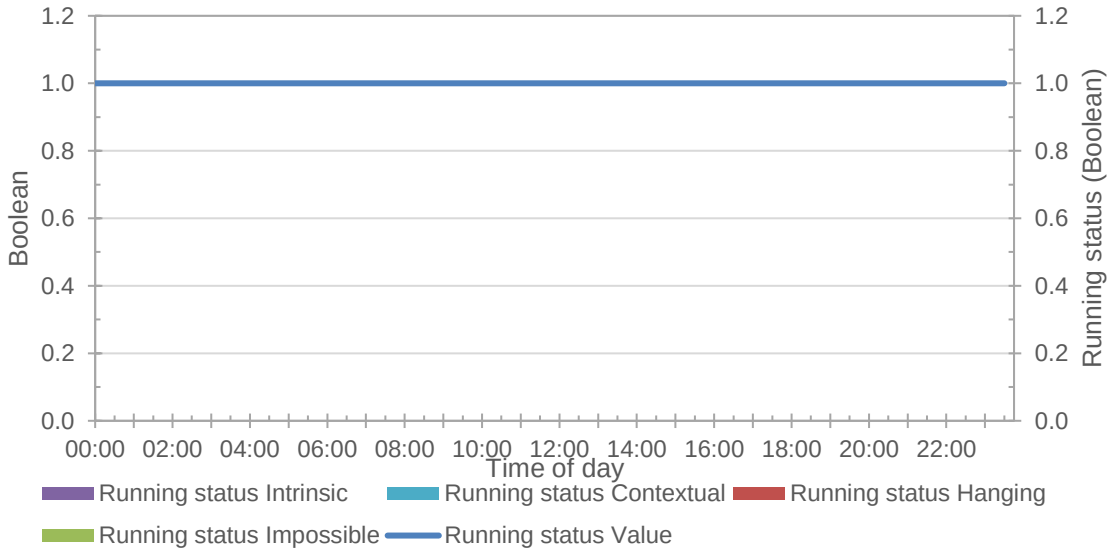

Figure 53: Clean running verification results

Figure 54 illustrates the unnormalised electrical current data stream analysed by the system and illustrates that the system incorrectly flagged three data points as low-integrity points from a contextual integrity perspective. This caused the overall integrity to be incorrectly indicated as low-integrity points. These three points were incorrectly flagged due to the temperature and vibration data streams having sudden changes at these times, which resulted in the data points being incorrectly flagged as low-integrity points from two characteristic perspectives.

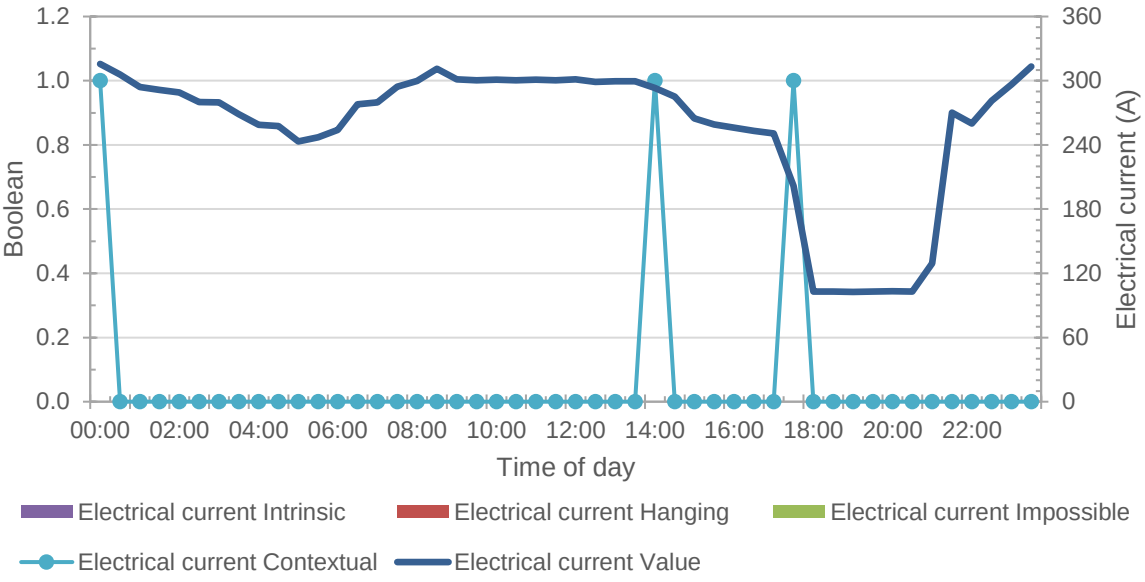

Figure 54: Clean electrical current verification results

Figure 55 illustrates the unnormalised temperature data stream, showing that the system incorrectly flagged two temperature points. This incident was caused by the electrical current and vibration data streams rising sharply. The system expected a larger increase in temperature values, ultimately indicating that these temperature values were untrustworthy.

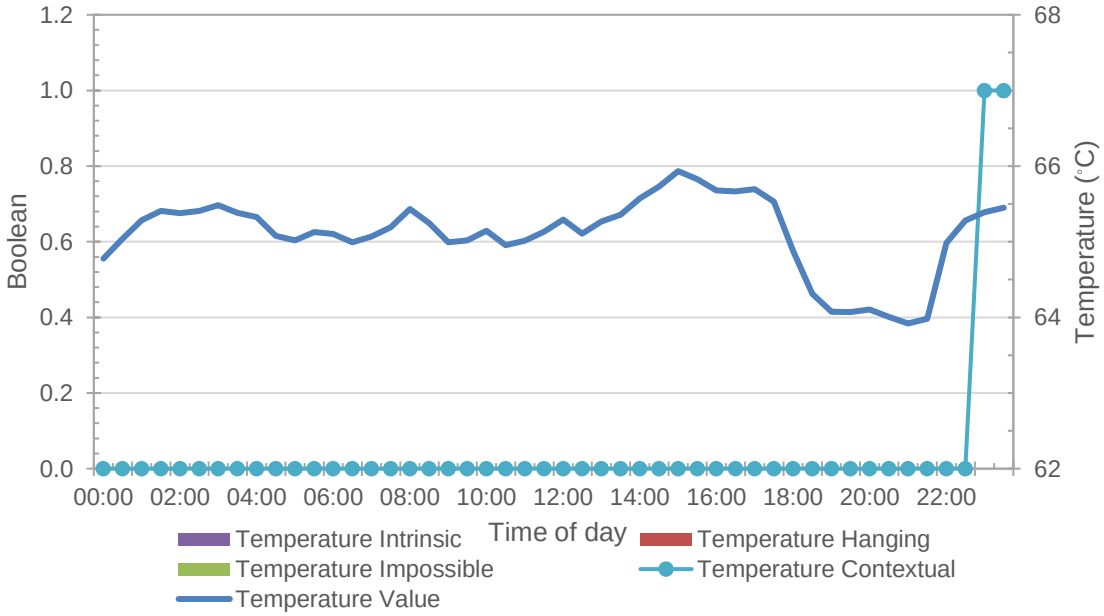

Figure 55: Clean temperature verification results

Figure 56 illustrates the unnormalised vibration data stream, showing that the system incorrectly identified five data points as low integrity. These instances are the combination of the incorrectly classified data points in Figure 54 and Figure 55.

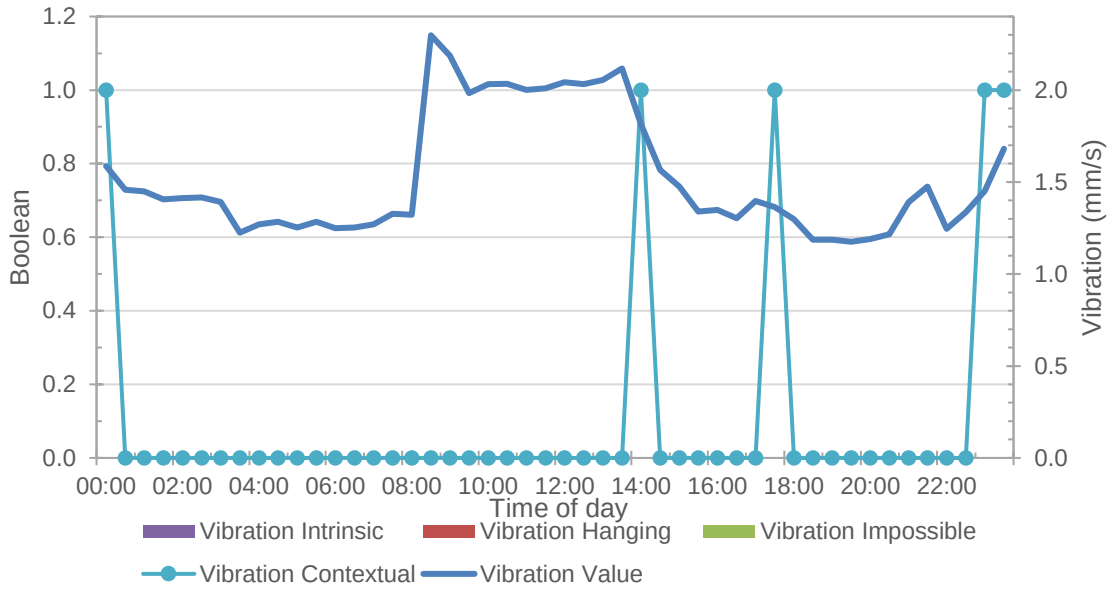

Figure 56: Clean vibration verification results

Clean Dataset 2

The results for the running-status data stream for Clean Dataset 2 were evaluated and are displayed in Figure 57 which illustrates that two data points were incorrectly flagged as unreliable at the point when the component switched off.

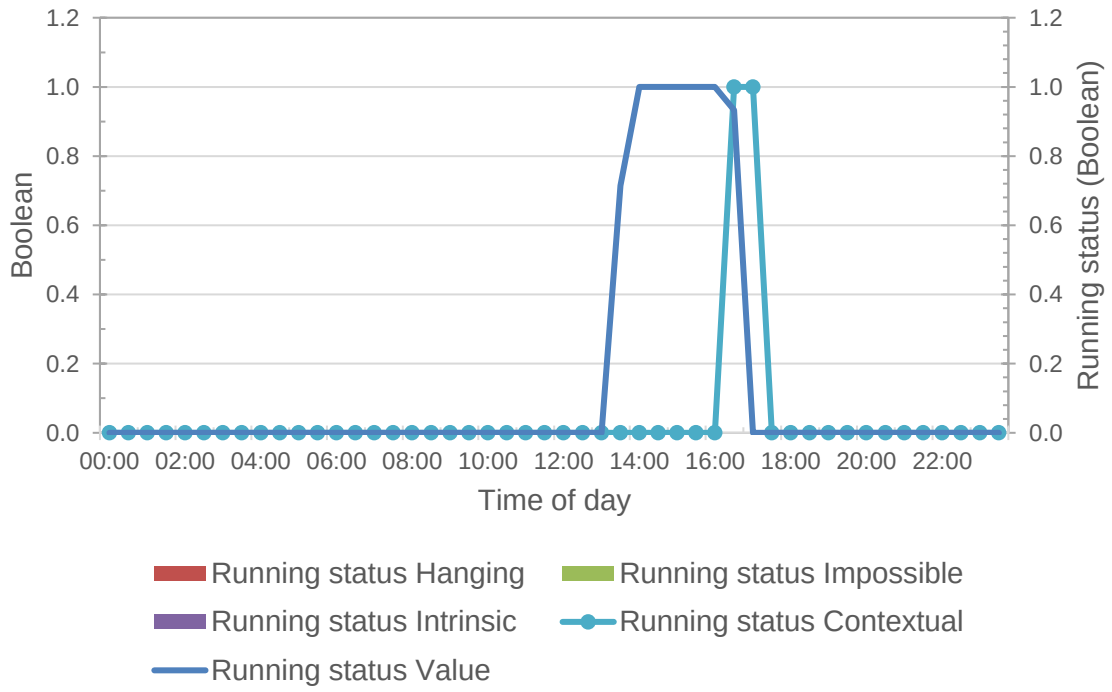

Figure 57: State-switching clean running verification results

The electrical current results contained three incorrectly flagged unreliable data points, as illustrated in Figure 58.

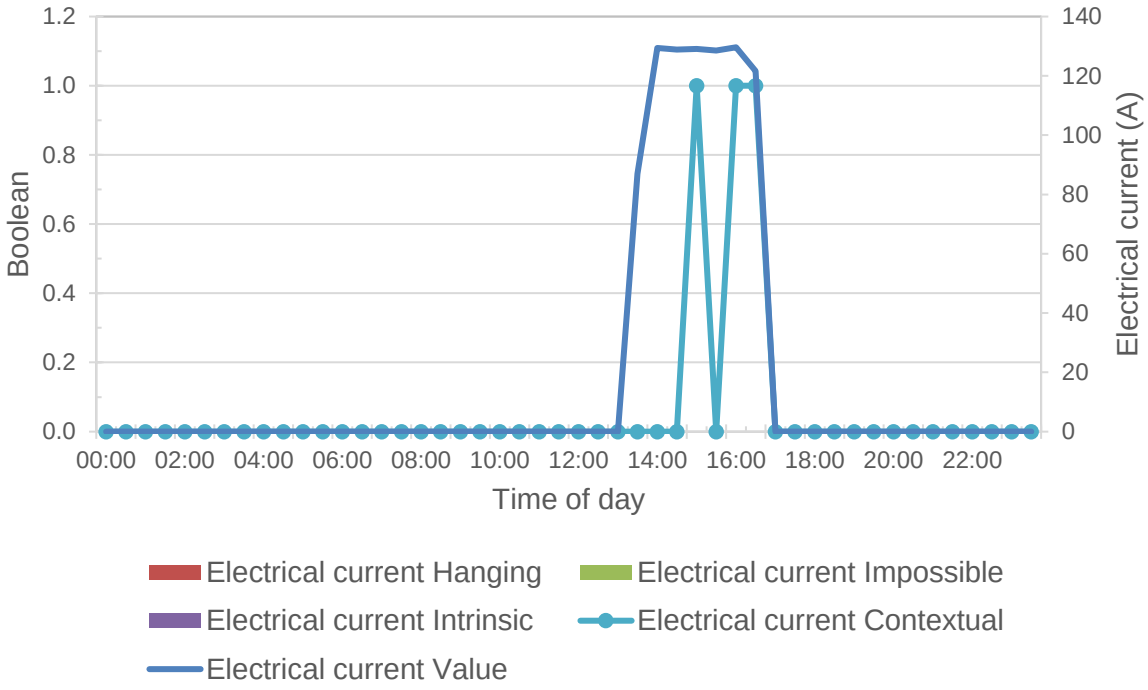

Figure 58: State-switching clean electrical current verification results

In Figure 59, the two data points incorrectly flagged as unreliable correspond to those incorrectly flagged for the running status and electrical current.

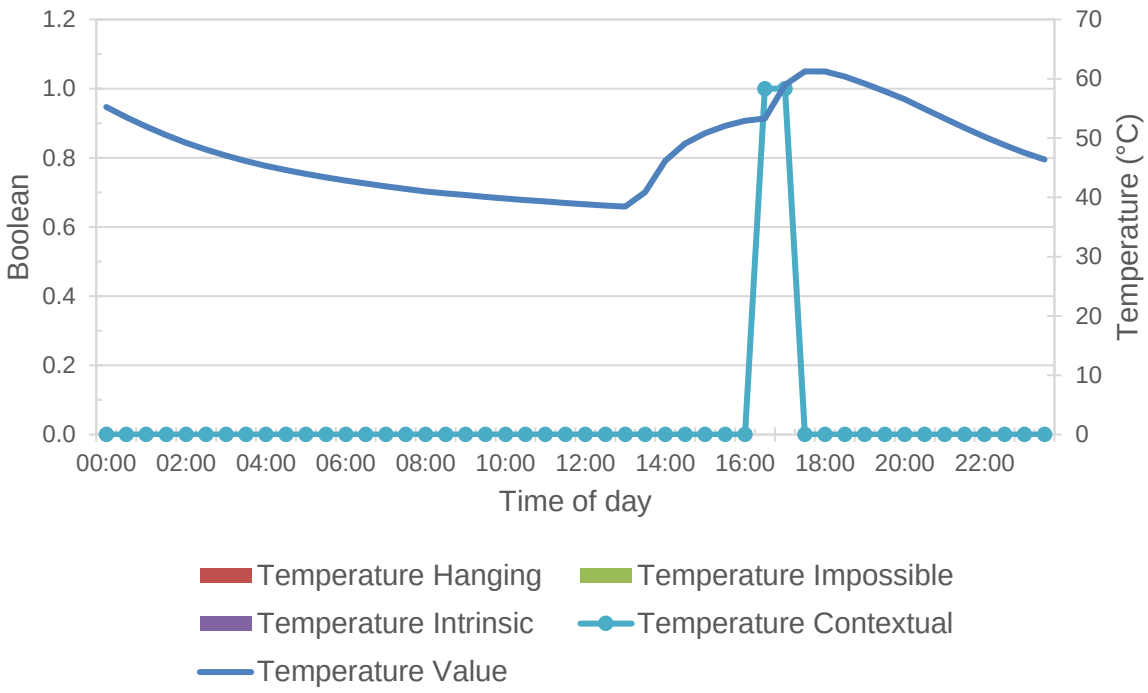

Figure 59: State-switching clean temperature verification results

Figure 60 shows the results for the vibration data stream and indicates incorrectly flagged unreliable data points that correlate to the incorrectly flagged data points of the electrical current data stream.

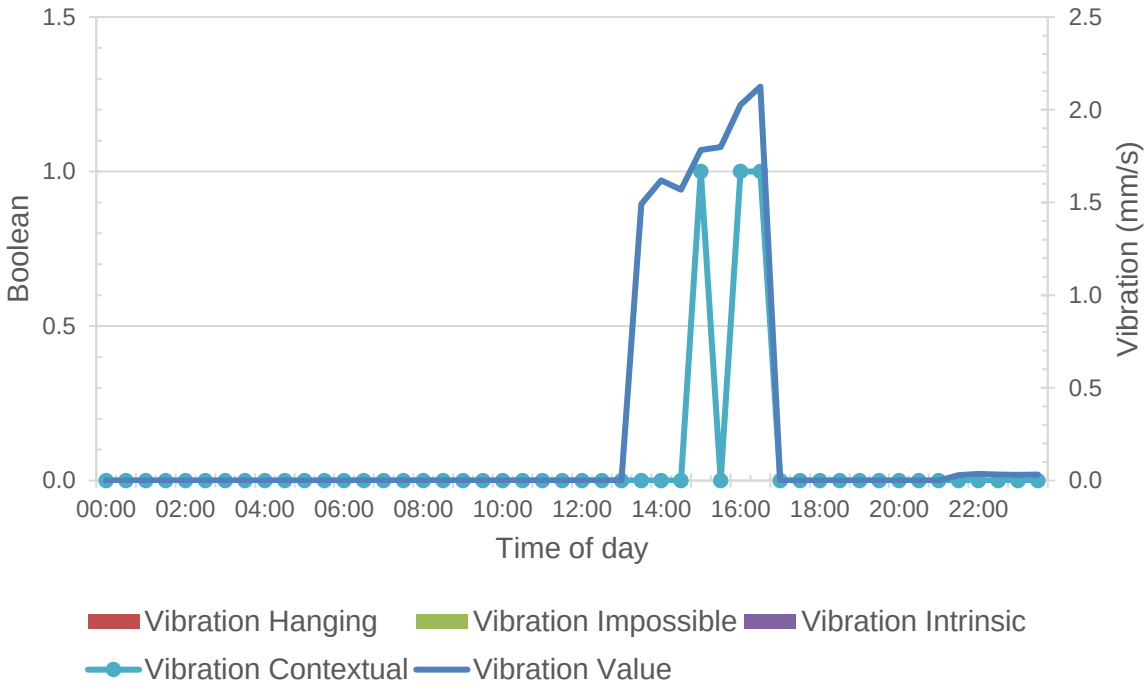

Figure 60: State-switching clean vibration verification results

Erroneous Dataset

Figure 61 illustrates the results for the erroneous dataset for the running-status data stream. The system identified two data points with low reliability. One of the data points identified was indeed an erroneous data point and was correctly identified.

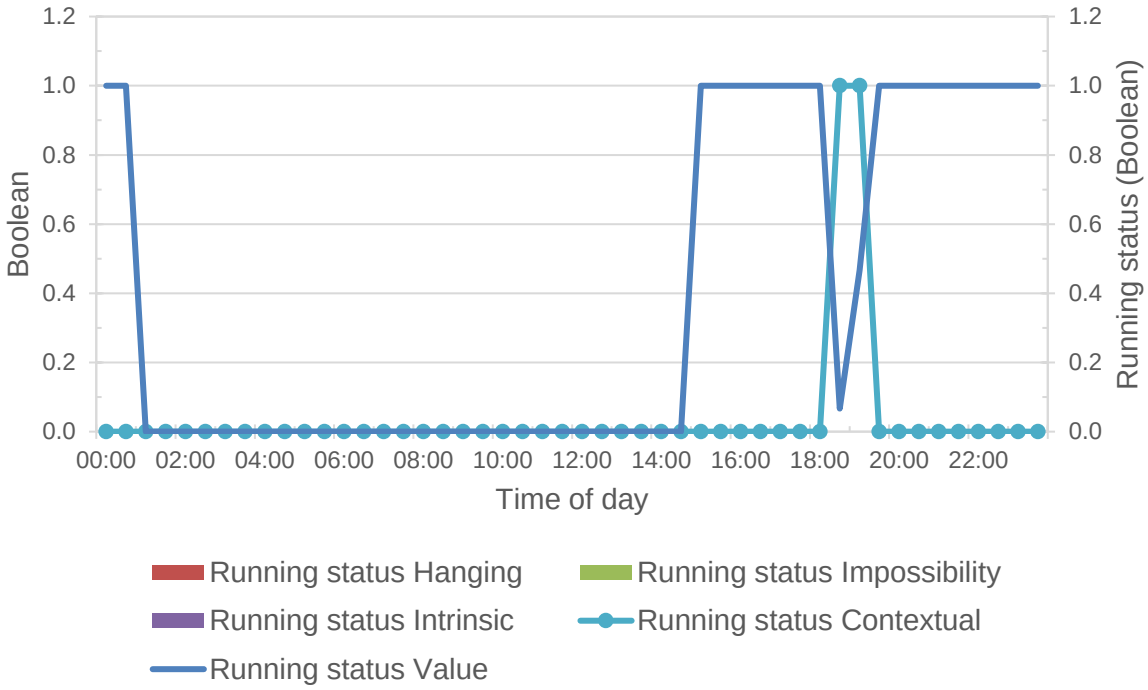

Figure 61: Erroneous dataset running verification results

The results for the electrical current data stream are displayed in Figure 62. Three data points were flagged as low reliability. Unfortunately, the system incorrectly identified all three points. However, the last identified data point was close to an erroneous one.

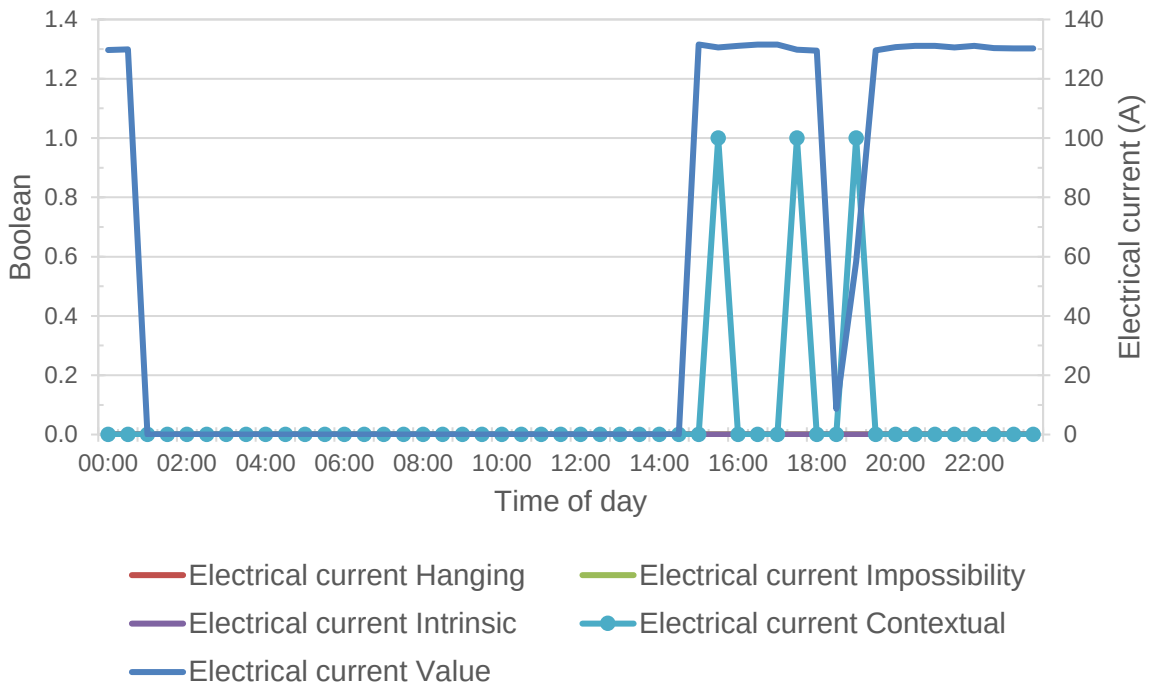

Figure 62: Erroneous dataset electrical current verification results

The results for the temperature data stream are displayed in Figure 63. The system identified five data points as being unreliable. Of the five identified data points, three were incorrectly identified.

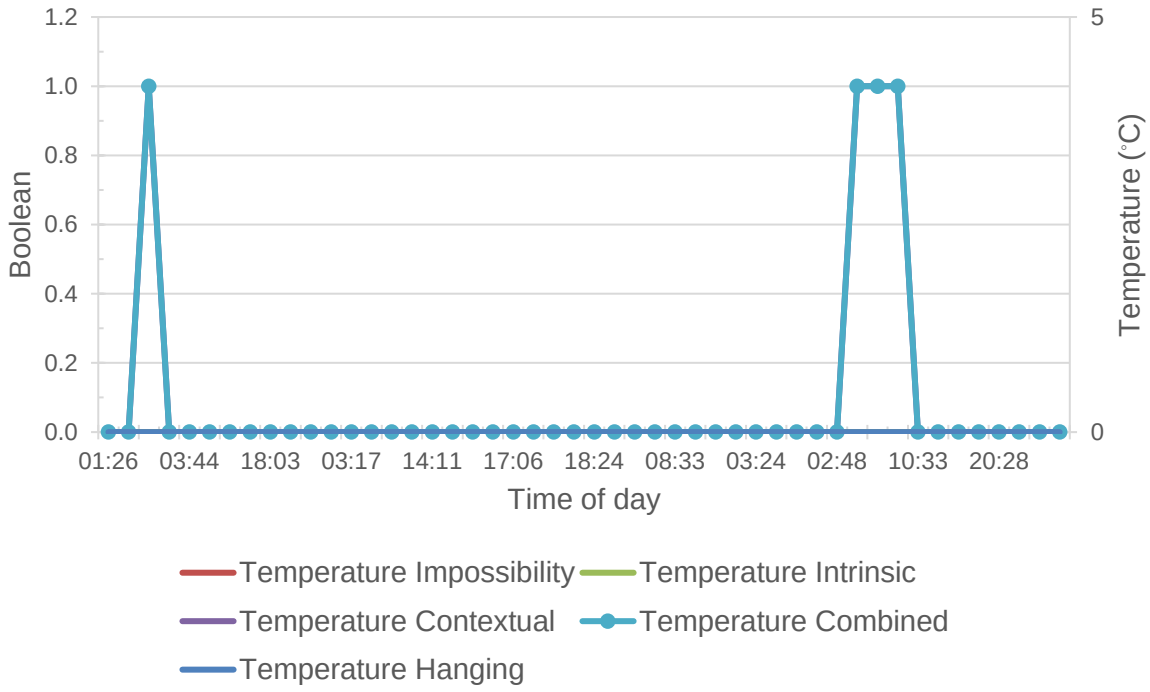

Figure 63: Erroneous dataset temperature verification results

Figure 64 displays the results for the vibration data stream. In the above figure, it can be seen that the system identified one data point as being unreliable. Unfortunately, the system incorrectly flagged this data point, leaving the unreliable data point undetected.

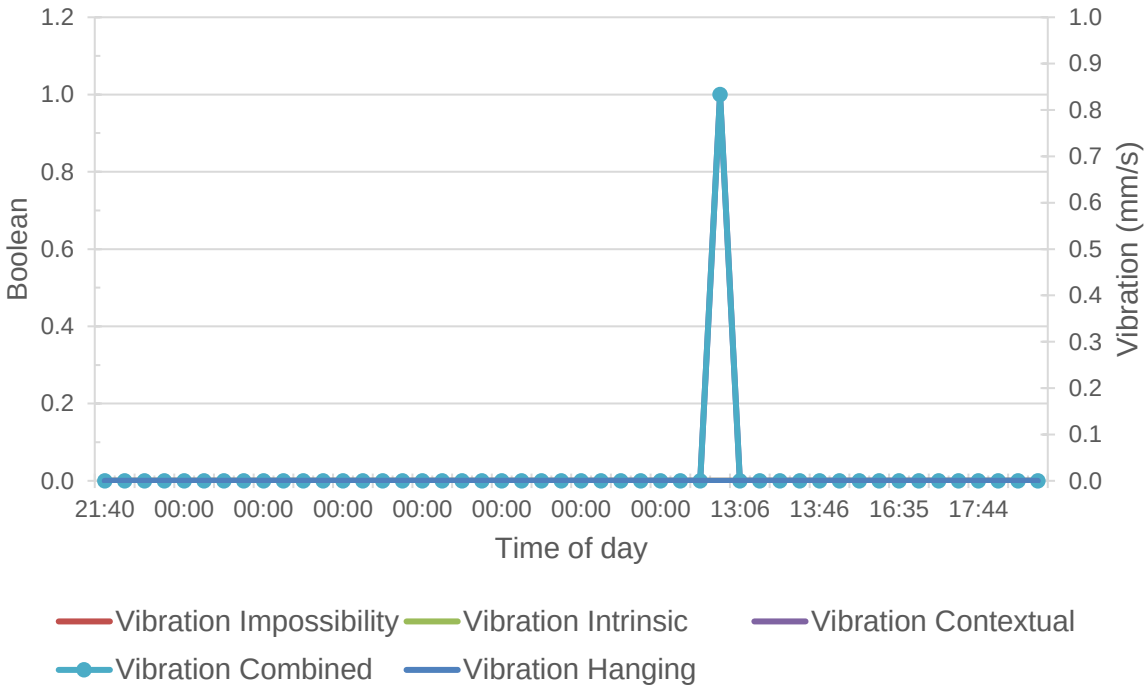

Figure 64: Erroneous dataset vibration verification results

Appendix C: Case study results

C-1: Component A

Component A was deemed to have reliable data, as only 11% of the data points were flagged as unreliable (Figure 65). An overwhelming majority of the unreliable data points identified by the system were identified using contextual methods.

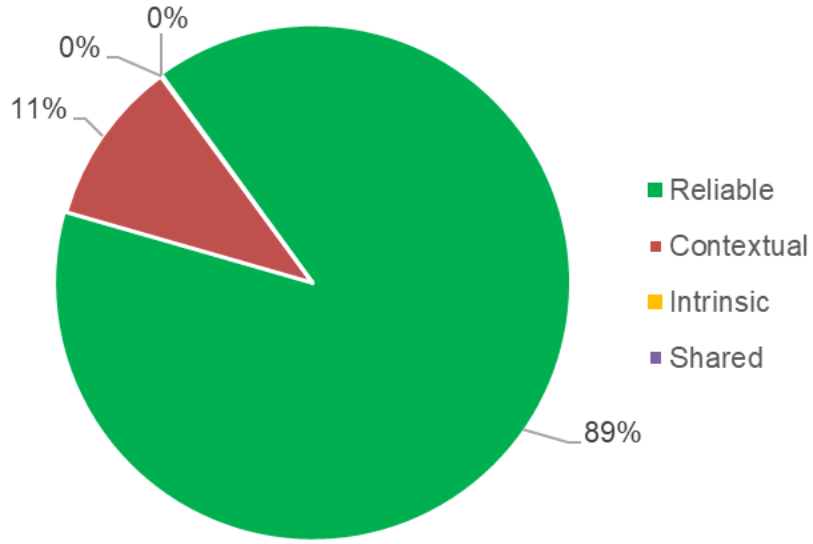

Figure 65: Component A reliability results percentage breakdown

Figure 66 shows the categorisation of valid and flagged data points by the characteristics for Component A. The vibration and temperature characteristics contained most of the unreliable data points.

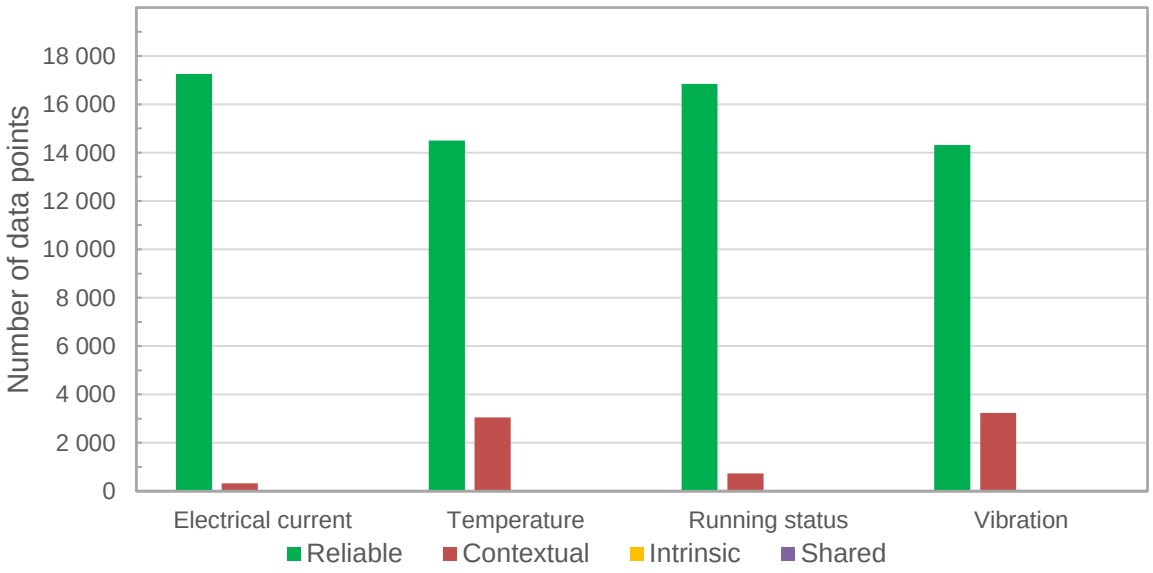

Figure 66: Component A reliability results breakdown

C-2: Component B

Component B was identified as having the highest fraction of unreliable data, highlighting a problem in the measurement equipment used to monitor this component. Figure 67 displays the troubling reliability breakdown. Roughly 40% of the data points were flagged as unreliable by contextual methods.

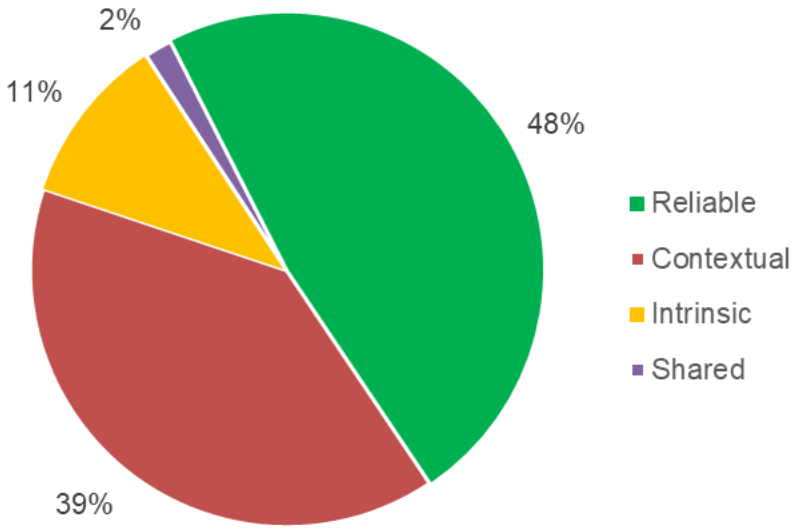

Figure 67: Component B reliability results percentage breakdown

Figure 68 displays the spread of unreliable data across the different characteristics. The temperature characteristic was the only data stream with more than 60% reliable data. The vibration characteristic was heavily flagged by the intrinsic methods, accounting for more than 40% of the data points in the data stream.

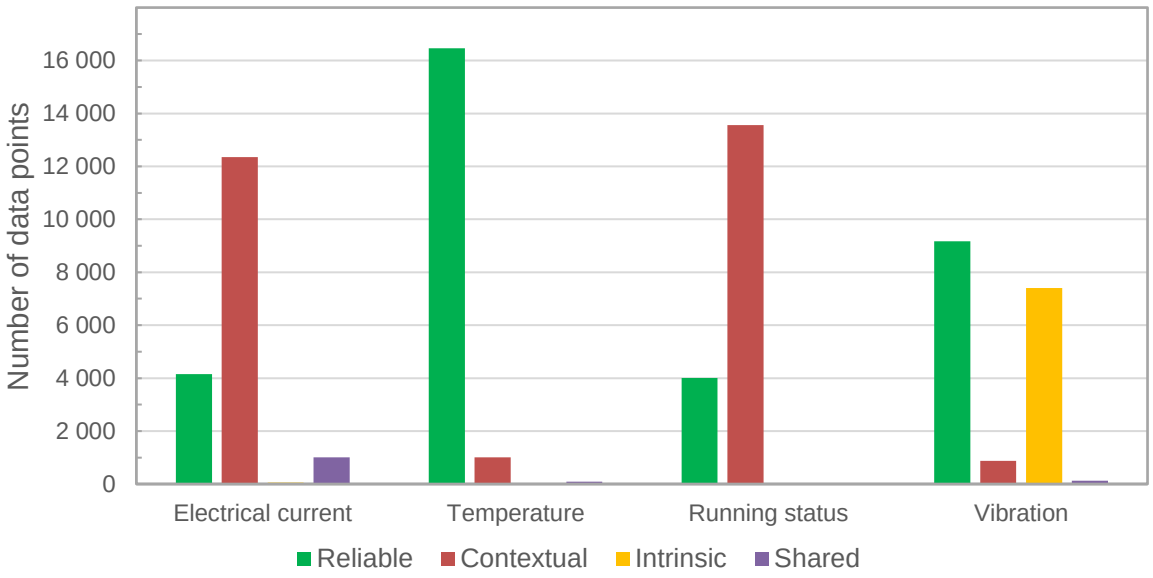

Figure 68: Component B reliability results breakdown

Figure 69 displays a diurnal plot for 4th January 2020 for each of the four characteristics. It can be seen that the component was switched off as both the running status and electrical current drawn are zero. Considering the minimal variation in the temperature values, further claims are made that the component was indeed in the *off* state. A corresponding vibration value of between zero and the environmental vibration is expected. For this component, the environmental vibration was calibrated to zero. As the vibration measurements never go below 0.19, this indicates that the vibration sensor was not correctly calibrated, and as a result, has a floating value.

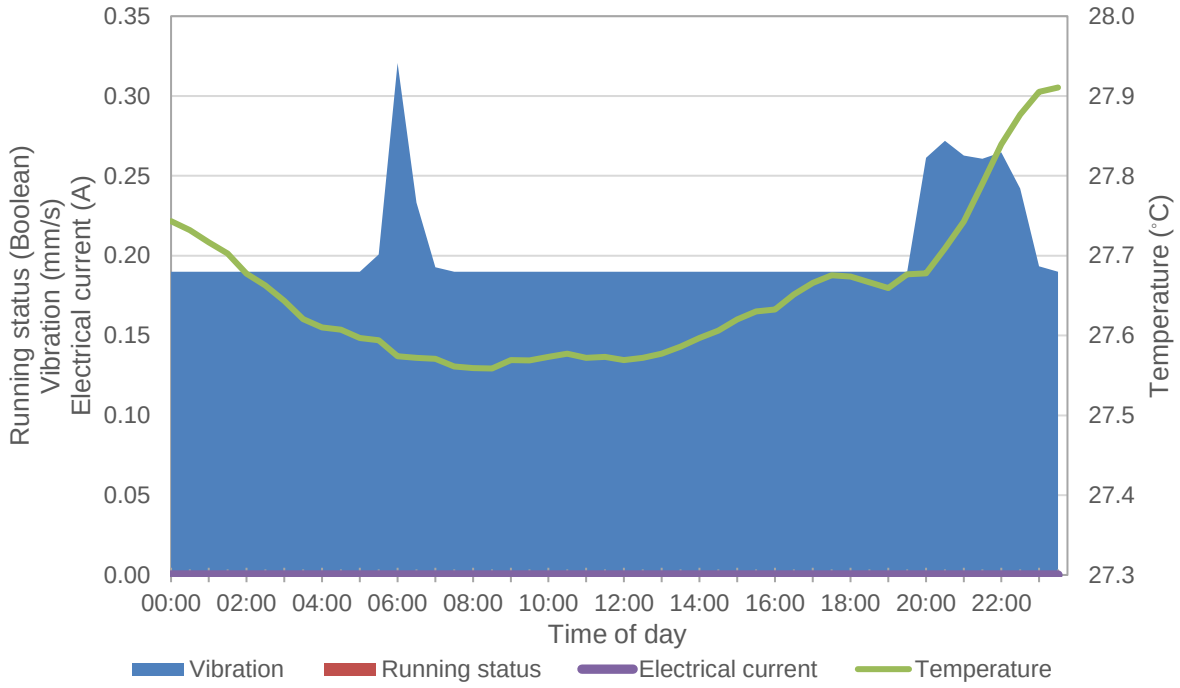

Figure 69: Uncalibrated vibration sensor for Component B (4 January 2020)

Figure 68 indicates a large number of unreliable data flagged by the contextual methods. The condition of the sensors may have deteriorated during the year, as reflected by examining the diurnal operation of Component B at year-end (31st December 2020) as an example.

Figure 70 illustrates conflicting operational states between the different characteristics. From a temperature perspective, the component was off between 00:00 and 03:00, switched on at 03:00, switched off again between 05:00 and 06:00, and remained off for the rest of the day. From an electrical current perspective, the component was in operation for the entire day, with reduced loads/strain for 03:00 - 06:30 and 19:30 –

21:00. From a running perspective, the component was only switched on between 03:30 and 06:30.

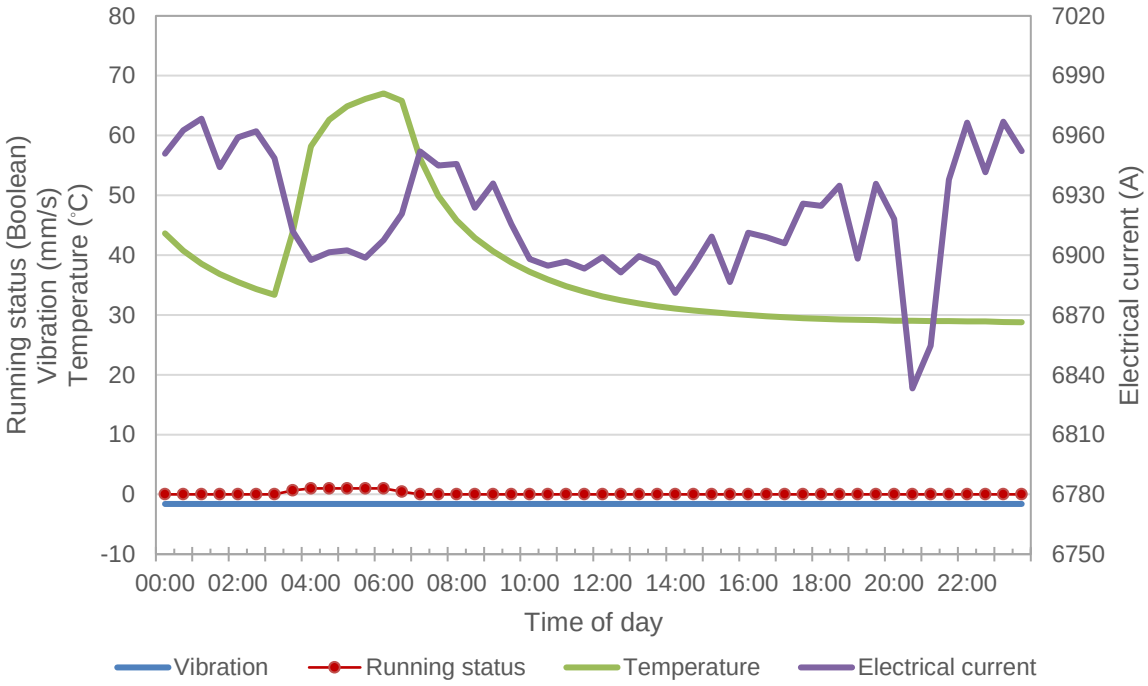

Figure 70: Severe sensor issues identified for Component B (31 December 2020)

The vibration reading implies that the component was never in operation, as it read a constant negative value. The negative values highlight that the vibration sensor was uncalibrated, as negative values are a physical impossibility. The temperature and running characteristics align and are inverted from the electrical current. It is unclear what the actual operational state of the component was for the day. The safest solution would be to dismiss all data streams as being unreliable.

This analysis of Component B supports the need for this study set out in Chapter 2, as both the intrinsic and contextual methods identified different data as unreliable.

C-3: Component C

Component C had a total of 19% unreliable data points. Figure 71 illustrates that most of these (14%) were identified by the intrinsic methods.

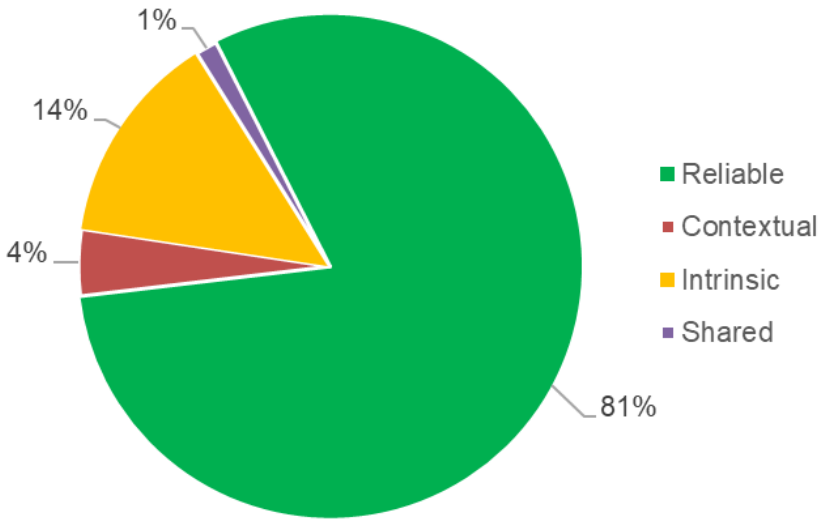

Figure 71: Component C reliability results percentage breakdown

Figure 72 illustrates that the temperature characteristic consists of mostly unreliable data points. The system flagged the data points in question as unreliable for one of three reasons:

- The measured values are impossible values for the characteristics
- The values are absent, or
- There are repeating values for extended periods outside of an expected range.

Considering the above, it was concluded that the measurement equipment disconnected or was not calibrated correctly.

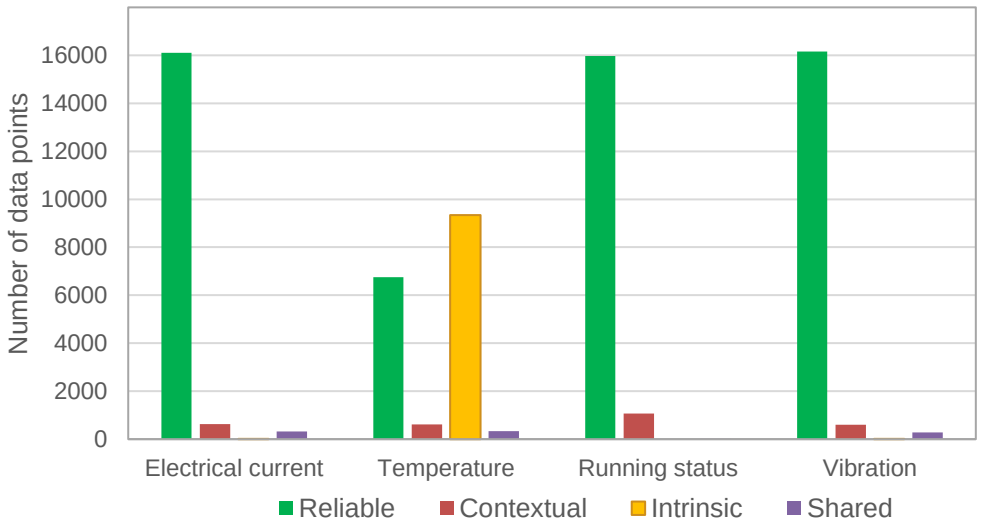

Figure 72: Component C reliability results breakdown

C-4: Component D

Component D contained an even split between data points flagged as unreliable by the intrinsic and contextual methods (Figure 73). Figure 73 suggests that the measurement equipment used for Component D was not correctly calibrated, as there is an even split between unreliable data identified by the intrinsic and contextual methods.

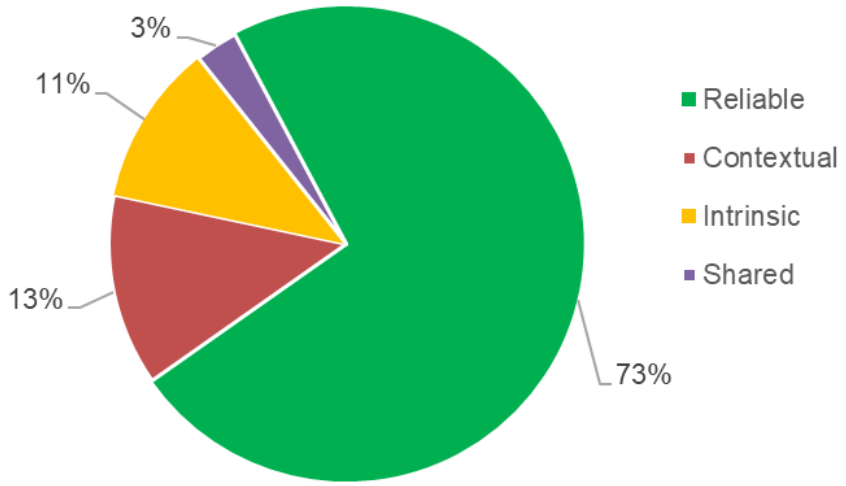

Figure 73: Component D reliability results percentage breakdown

Figure 74 illustrates how the unreliable data identified by the contextual methods are distributed between all four characteristics. It can be seen that the unreliable data identified by the intrinsic methods are concentrated on one characteristic - vibration. Conversely, the unreliable data identified by the intrinsic methods are primarily concentrated in the vibration characteristic.

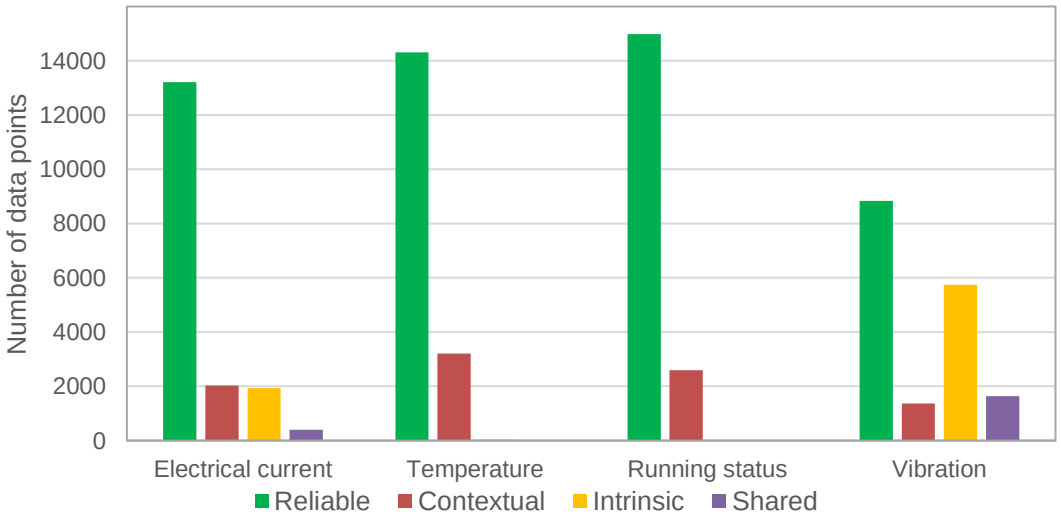

Figure 74: Component D reliability results breakdown

Upon further investigation, it was found that the vibration measurement equipment was not correctly calibrated. For example, the diurnal plot (Figure 75) illustrates that the component was switched off for most of the day. The measurements during the off stages, although hard to see, are negative values. Negative values for the vibration characteristic for this component are impossibilities and were thus flagged as unreliable, regardless of how small the negative values are.

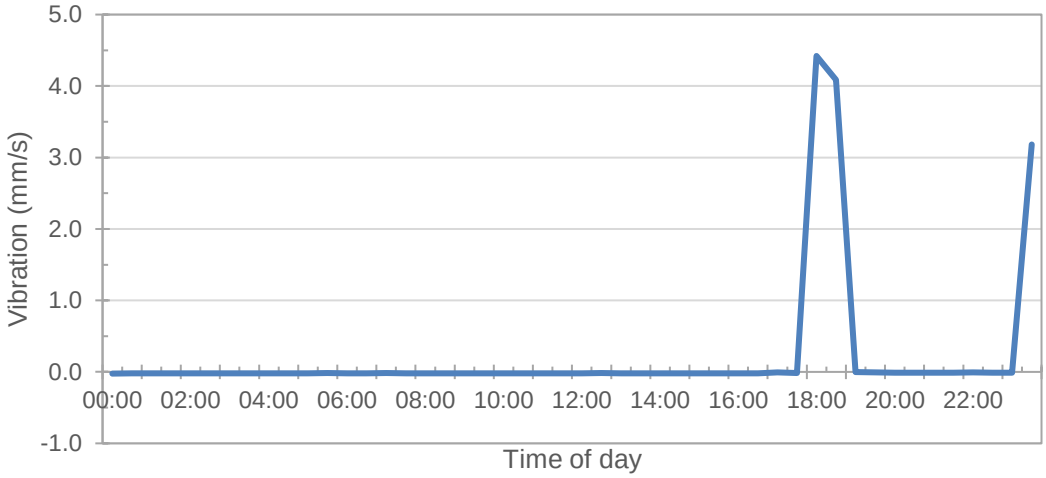

Figure 75: Uncalibrated vibration sensor for component D (1 January 2020)

C-5: Component E

Component E had a large amount (21%) of unreliable data, as displayed in Figure 76. Most of the unreliable data points (17%) were flagged as unreliable by the contextual methods.

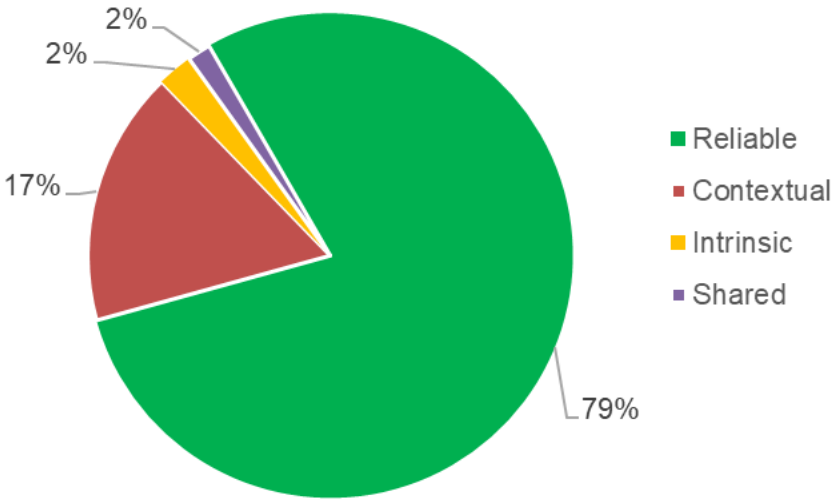

Figure 76: Component E reliability results percentage breakdown

Looking into more detail, Figure 77 shows that unreliable data are spread across all four characteristics, with the running status having the largest concentration.

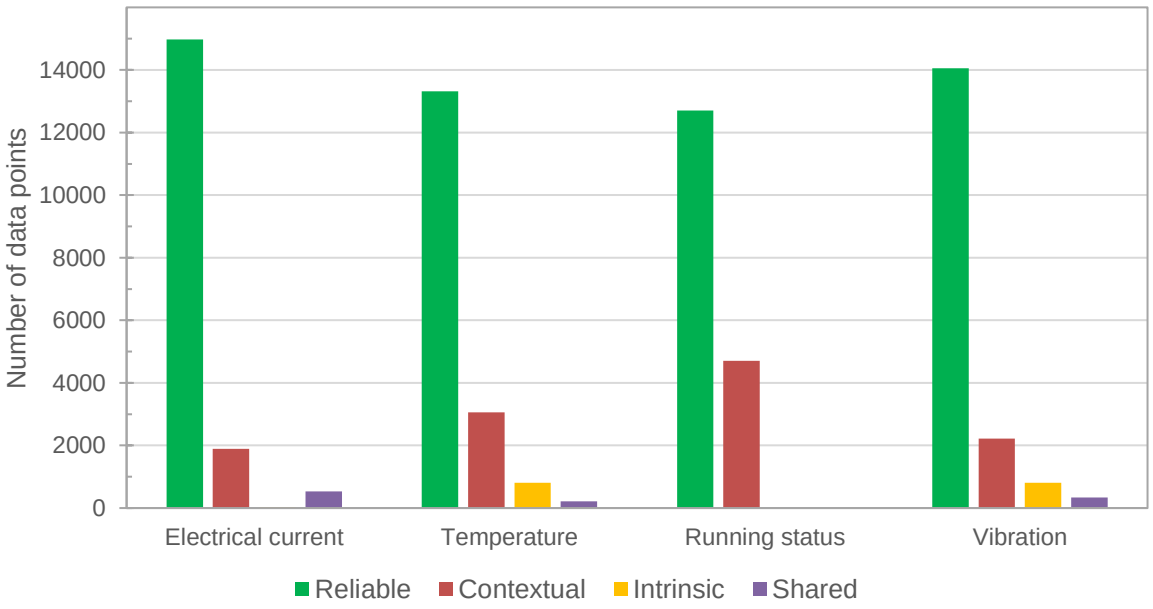

Figure 77: Component E reliability results breakdown

C-6: Component F

Component F can be seen as a well-monitored component, as it contains a low percentage of unreliable data. Figure 78 illustrates that component F has an even spread of unreliable data identified by the different methods.

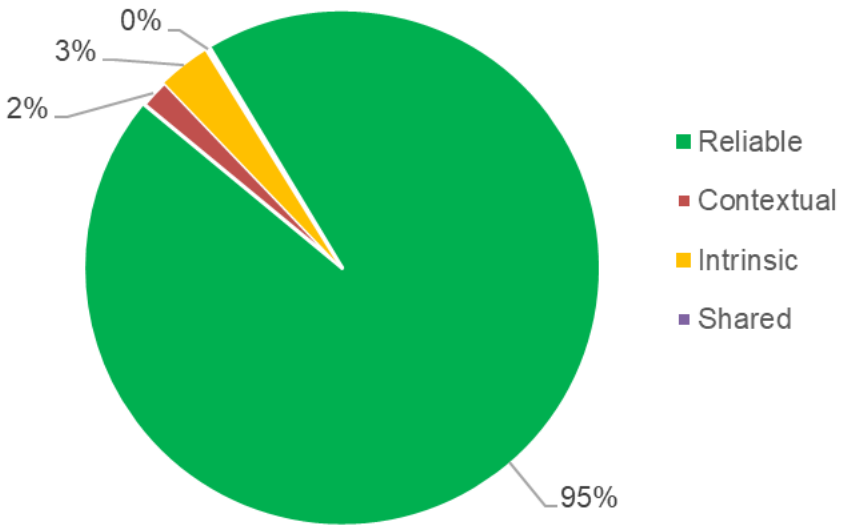

Figure 78: Component F reliability results percentage breakdown

Expanding on this statement, Figure 79 suggests an even spread of unreliable data across the four characteristics.

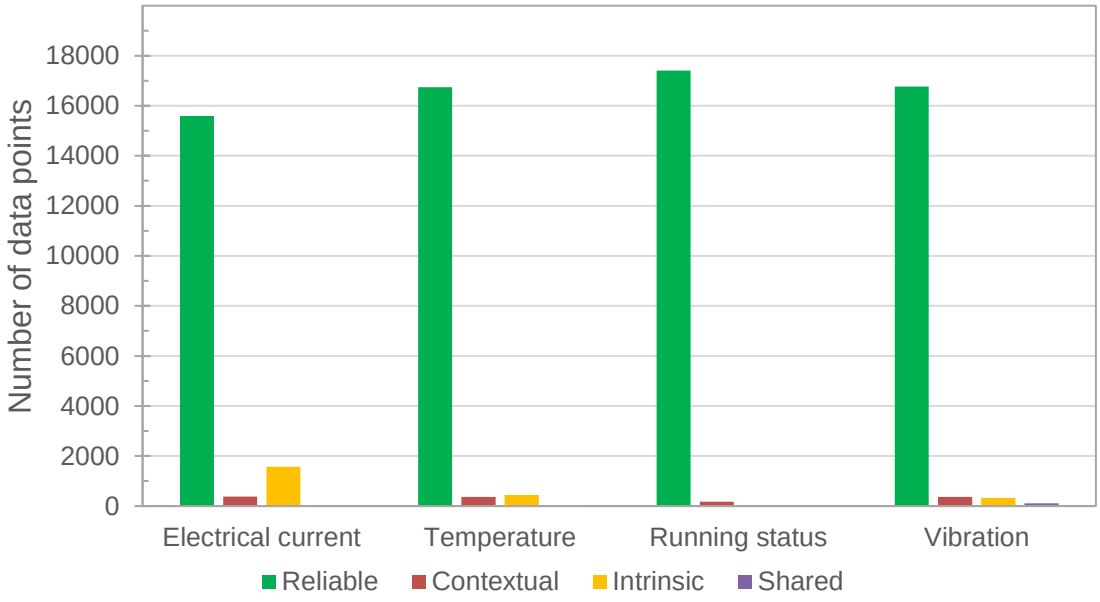

Figure 79: Component F reliability results breakdown

C-7: Component G

Component G, similar to component F, can be seen as a well-monitored component, as illustrated by Figure 80. There are minimal unreliable data (2%) spread evenly over all four characteristics, as displayed in Figure 81.

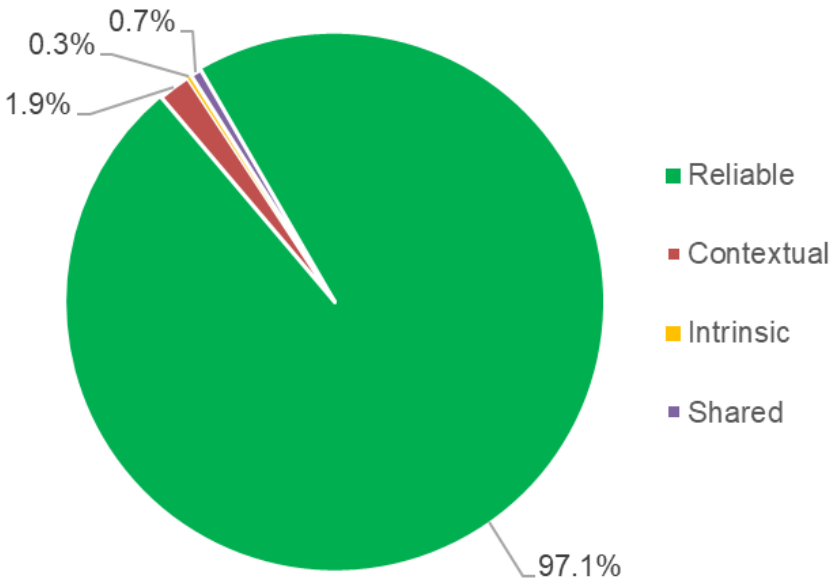

Figure 80: Component G reliability results percentage breakdown

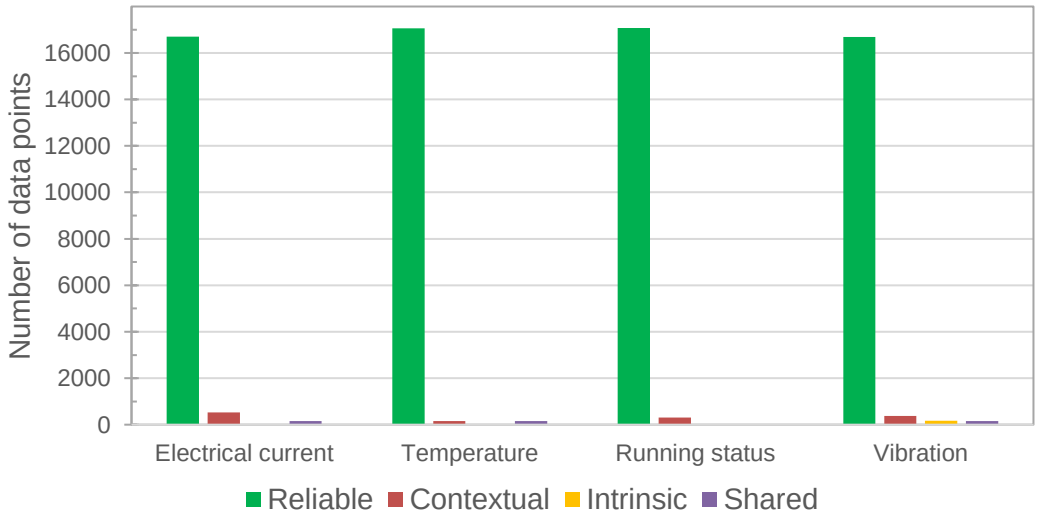

Figure 81: Component G reliability results breakdown

C-8: Component H

Component H had a large amount (34%) of unreliable data. Figure 82 illustrates how contextual methods flagged 20% of the available data.

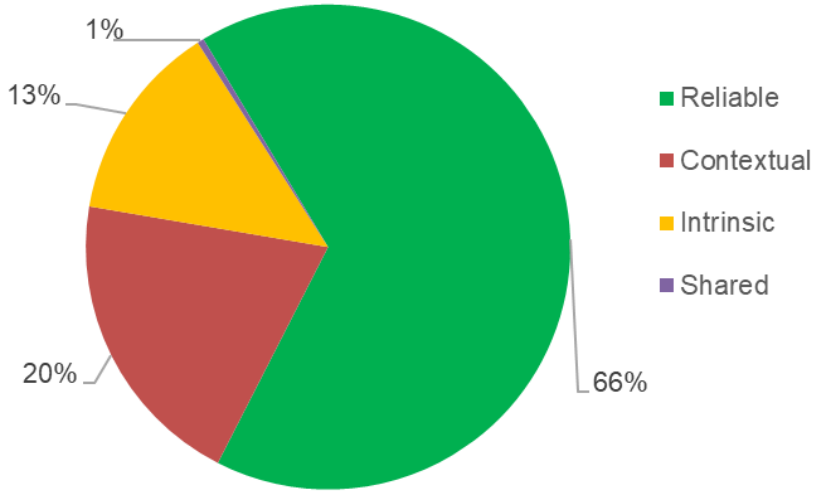

Figure 82: Component H reliability results percentage breakdown

Figure 83 shows the distribution of unreliable data across the characteristics. Unreliable data are found in the measurements of all four characteristics; however, the temperature characteristic contains more than 30% of unreliable data identified by the intrinsic methods. Similarly, the running characteristic contains more than 40% of unreliable data identified by the contextual methods.

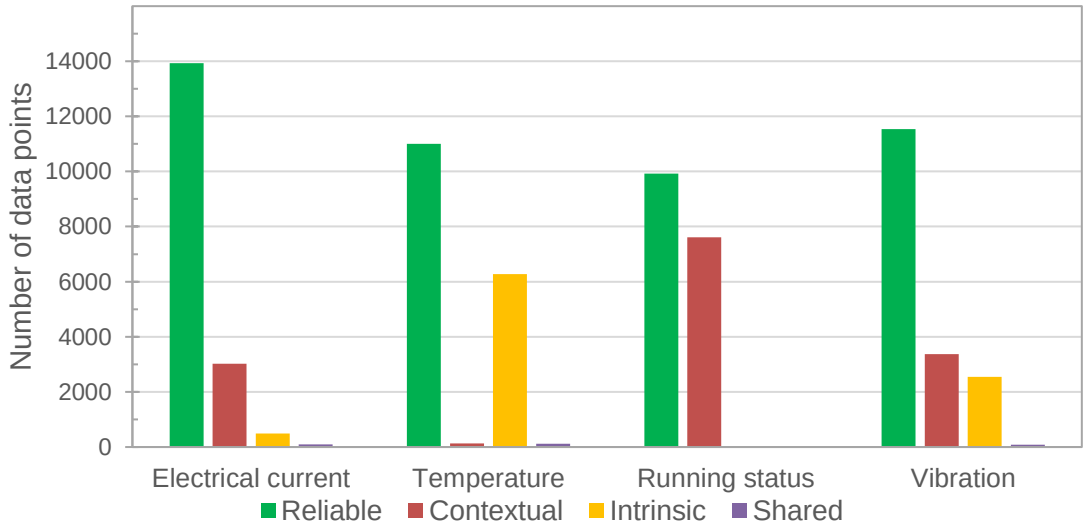

Figure 83: Component H reliability results breakdown

Figure 84 displays the characteristic diurnal profiles for Component H for 13th April 2020. The temperature reading stays constant with a single exception at 19:00.

Despite the zero electrical current and the vibration reading below the environmental vibration, the run status indicates that the component is in the *on* state. The temperature reading should not be constant, as the environmental and machine conditions constantly change. This constant value repeats for numerous datasets, resulting in the intrinsic methods flagging the data points as unreliable. Considering the electrical current and vibration values, it appears that the running status is erroneously inverted, indicating *on* rather than *off* status.

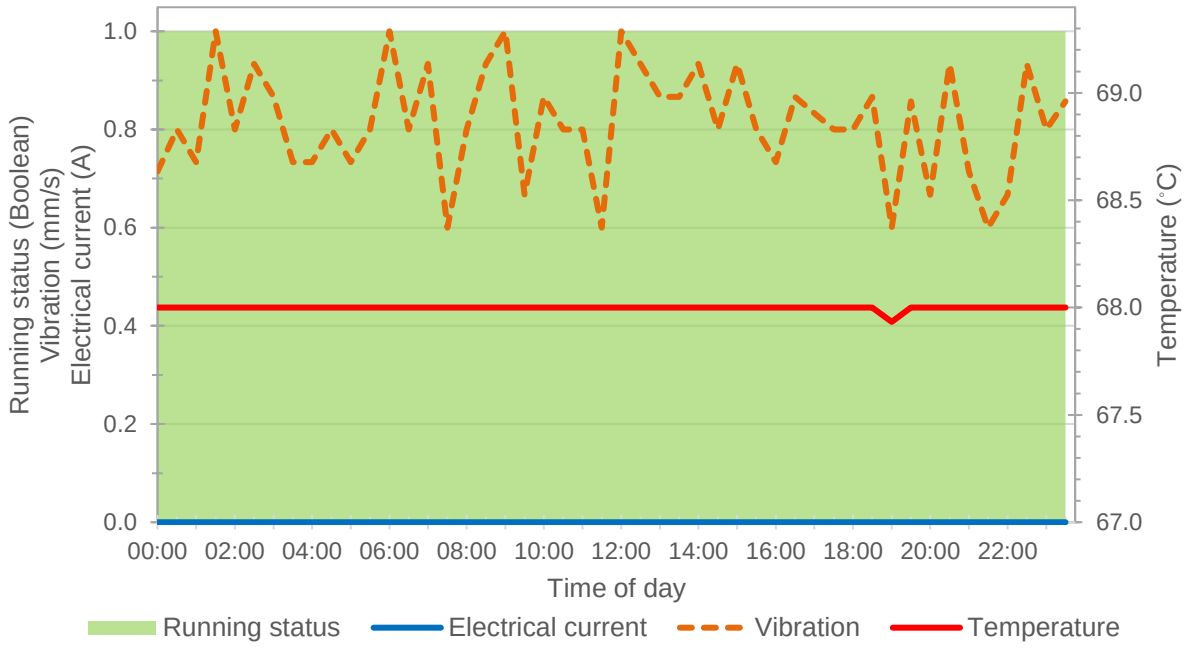

Figure 84: Diurnal characteristic profiles for Component H (13 April 2020)

C-9: Component I

Component I is a well-monitored component. Figure 85 illustrates the limited number of unreliable data points.

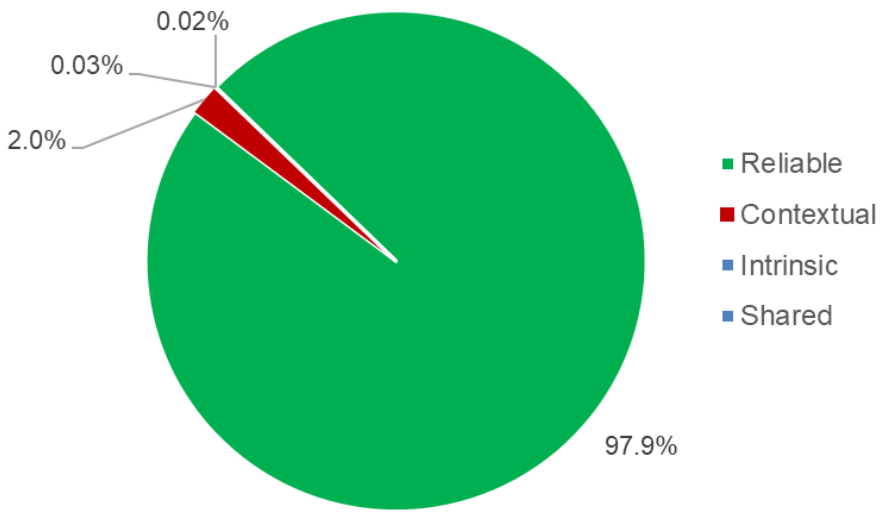

Figure 85: Component I reliability results percentage breakdown

These few unreliable data points are spread evenly across the four characteristics, as seen in Figure 86.

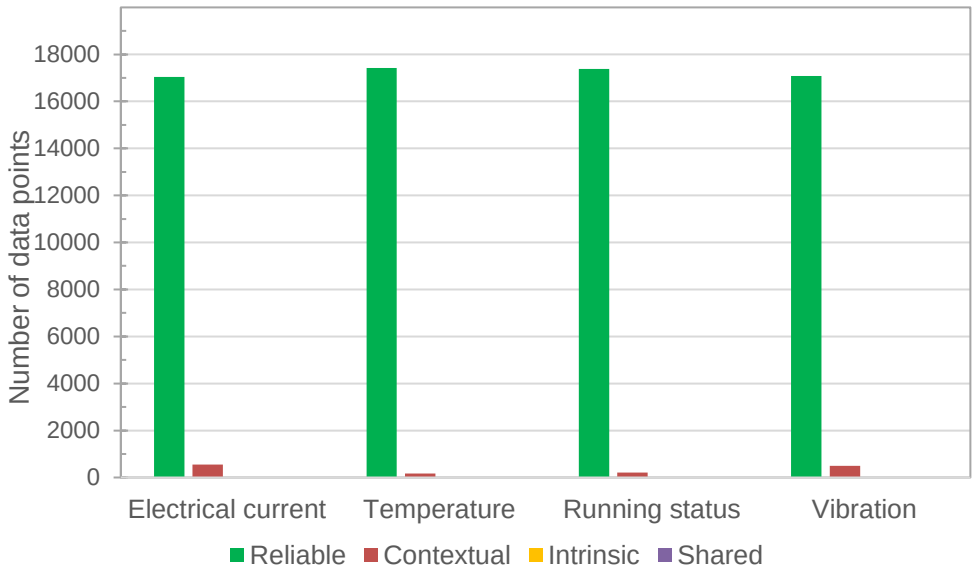

Figure 86: Component I reliability results breakdown

C-10: Component J

Component J has close to 25% unreliable data across the four data streams (Figure 87).

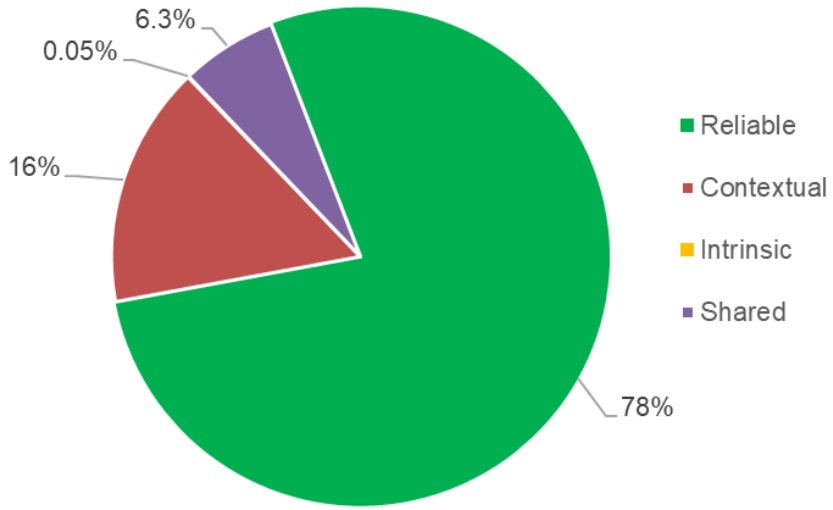

Figure 87: Component J reliability results percentage breakdown

Despite Figure 87 illustrating a high level of unreliable data, these untrustworthy data points are spread across the four characteristics, as seen in Figure 88.

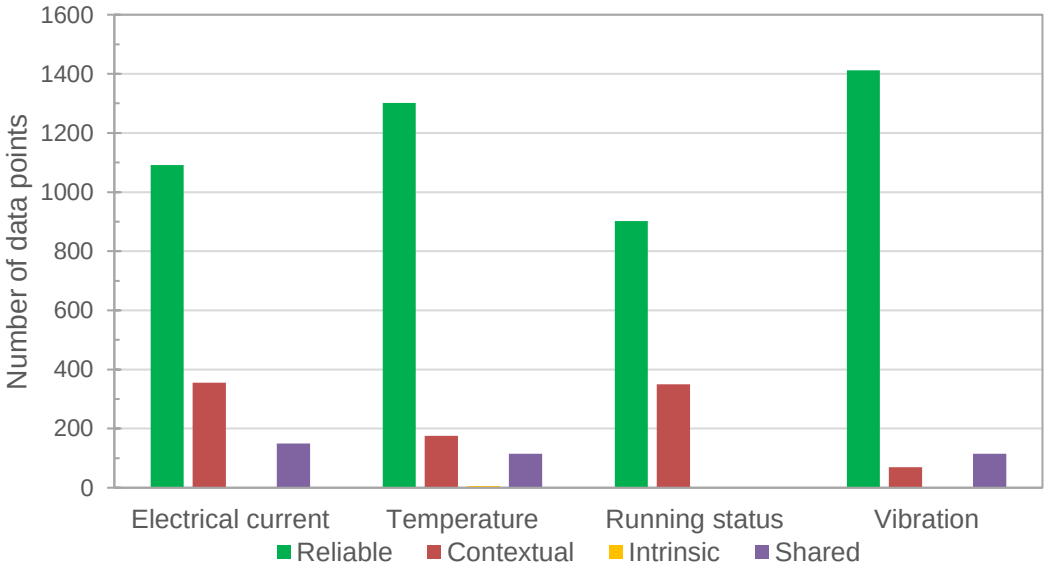

Figure 88: Component J reliability results breakdown

C-11: Component K

Figure 89 indicates that 16% of the data associated with Component K are unreliable. Most of the unreliable data are flagged by the contextual methods.

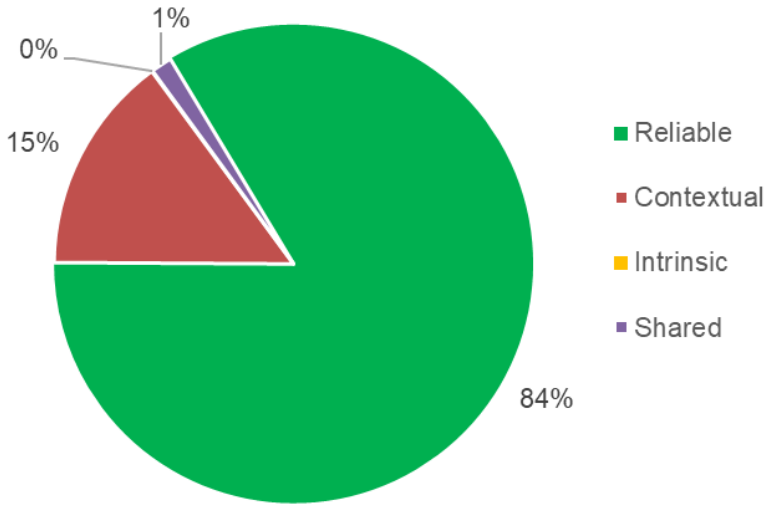

Figure 89: Component K reliability results percentage breakdown

Figure 90 illustrates how the unreliable data for Component K are divided between the four characteristics, with the temperature characteristic containing the least amount of unreliable data.

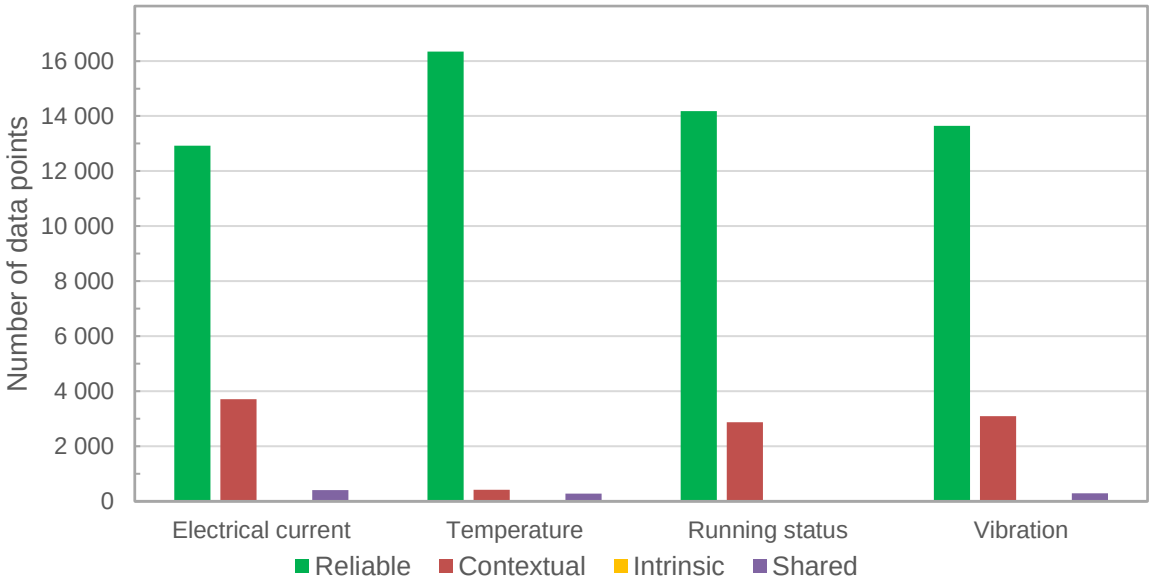

Figure 90: Component K reliability results analysis

C-12: Component L

Figure 91 indicates a large fraction (17%) of unreliable data linked to Component L, most identified by the contextual methods.

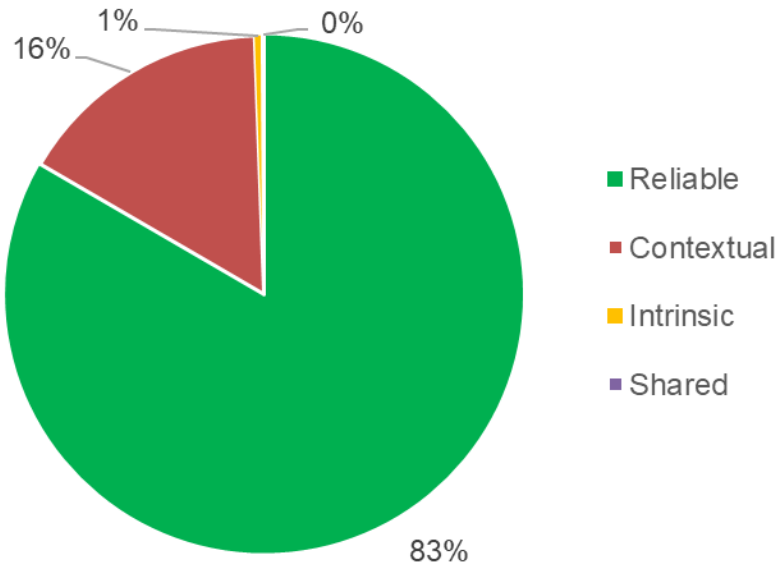

Figure 91: Component L reliability results percentage breakdown

Figure 92 shows that the unreliable data is concentrated in the electrical current and temperature characteristics.

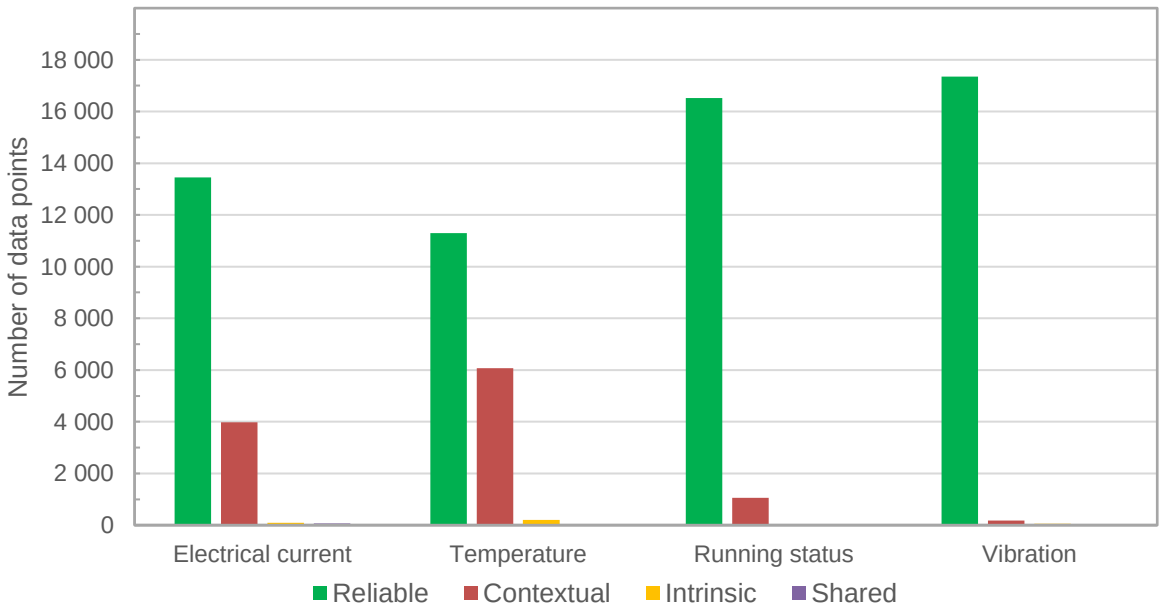

Figure 92: Component L reliability results breakdown

C-13: Component M

Figure 93 highlights a concern for Component M with 27% unreliable data, flagged by the contextual methods.

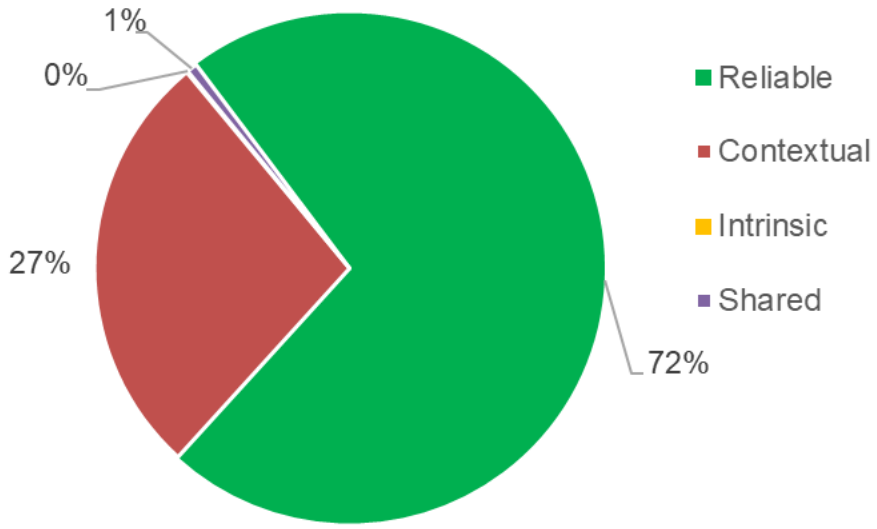

Figure 93: Component M reliability results percentage breakdown

Figure 94 was created to gain more insight into where the unreliable data was concentrated. In this case, the unreliable data were evenly spread across all four data streams.

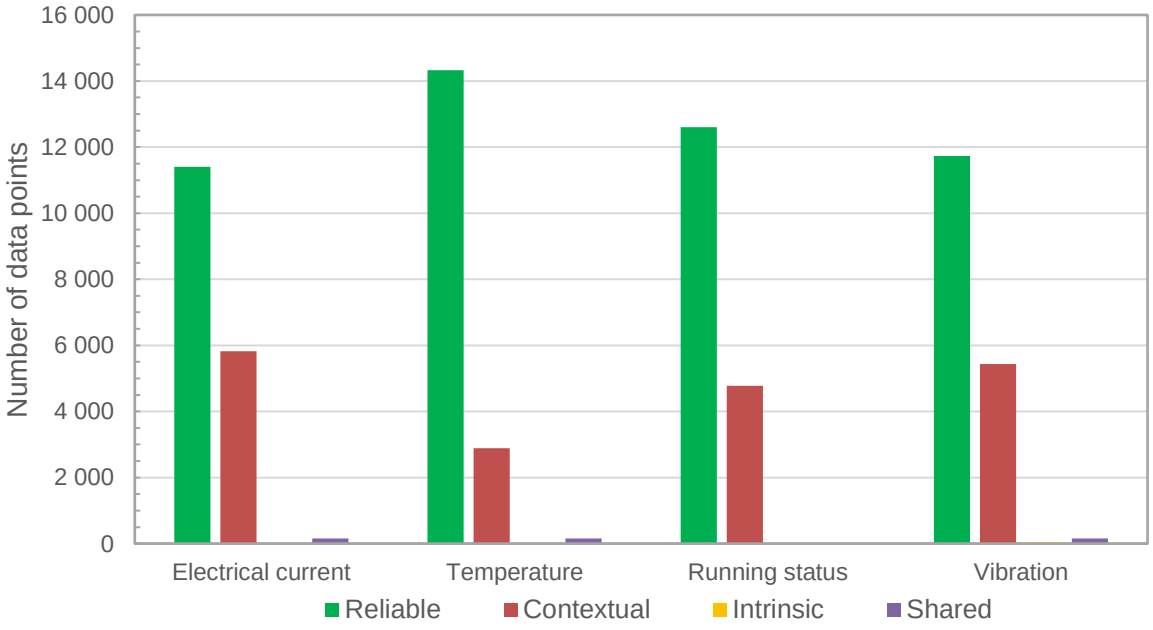

Figure 94: Component M reliability results breakdown

C-14: Component N

Component N contains limited (12%) unreliable data. Figure 95 indicates how the unreliable data identification is split between the intrinsic and contextual methods.

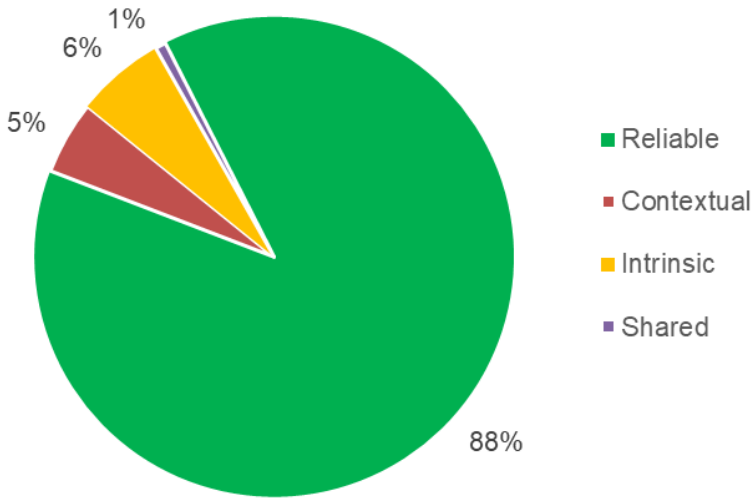

Figure 95: Component N reliability results percentage breakdown

Figure 96 illustrates the distribution of unreliable data between characteristics.

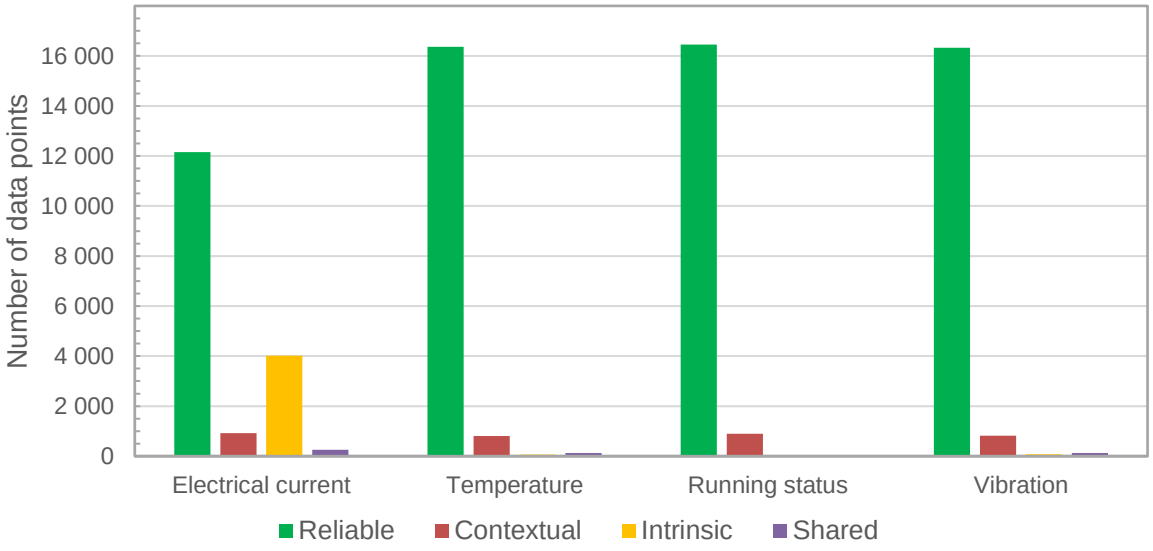

Figure 96: Component N reliability results breakdown

C-15: Component O

Component O contains a high percentage (91%) of reliable data (Figure 97). Figure 98 shows that of the unreliable data, most is concentrated in the electrical current characteristic.

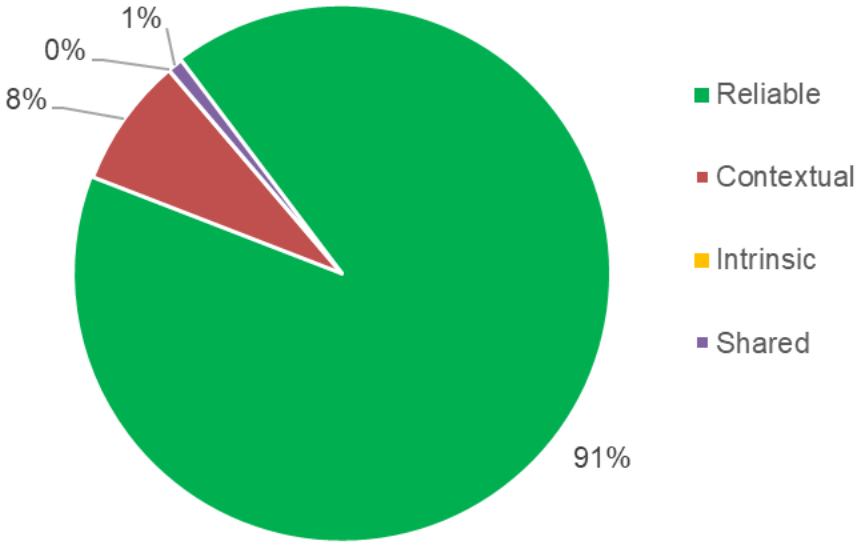

Figure 97: Component O reliability results percentage breakdown

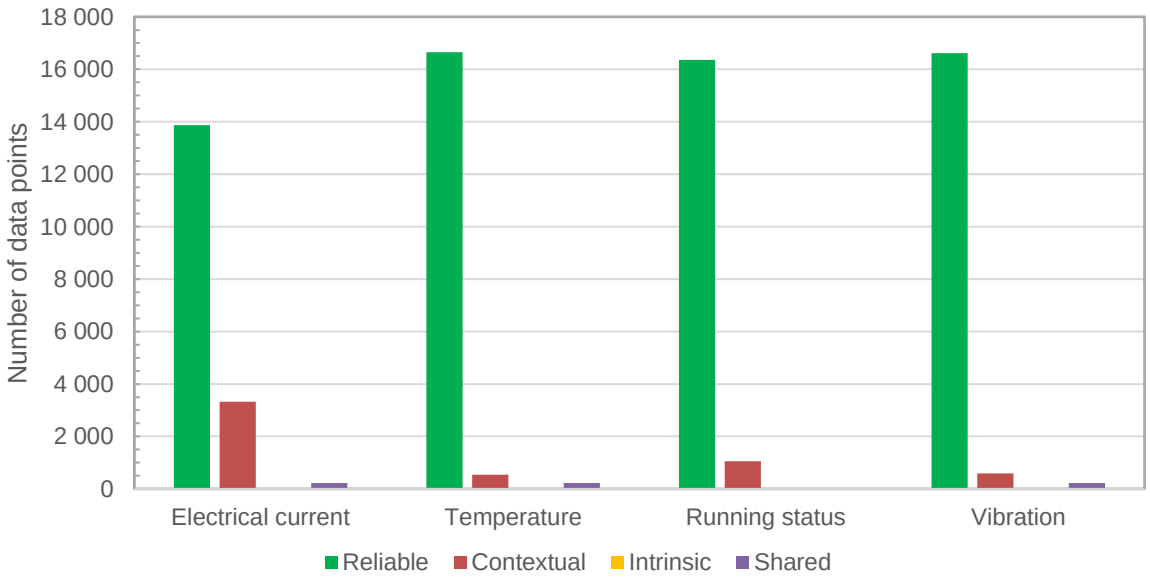

Figure 98: Component O reliability results breakdown

C-16: Component P

More than 50% of the data for Component P is unreliable (Figure 99). Figure 100 illustrates that both the intrinsic and contextual methods flagged a significant number of data points as unreliable. Figure 100 illustrates that unreliable data are distributed across all four characteristics.

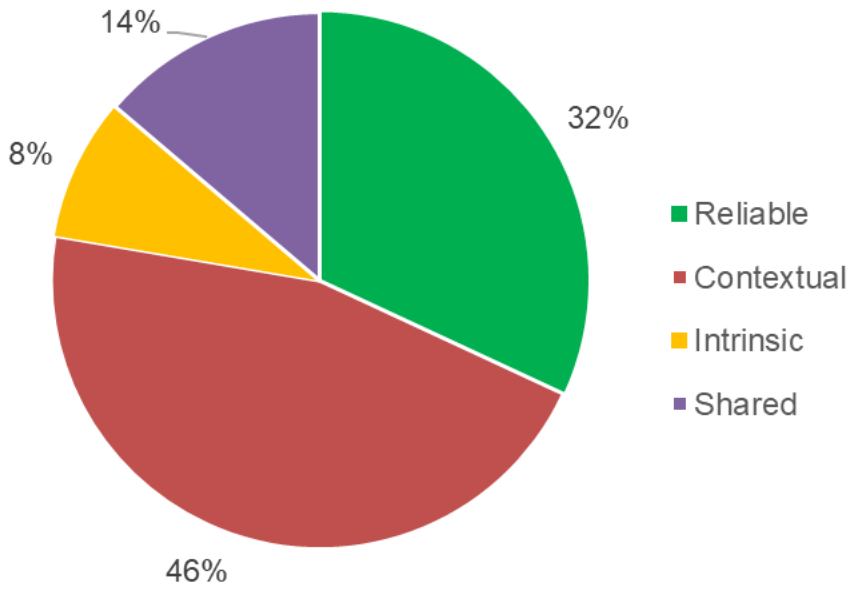

Figure 99: Component P reliability results percentage breakdown

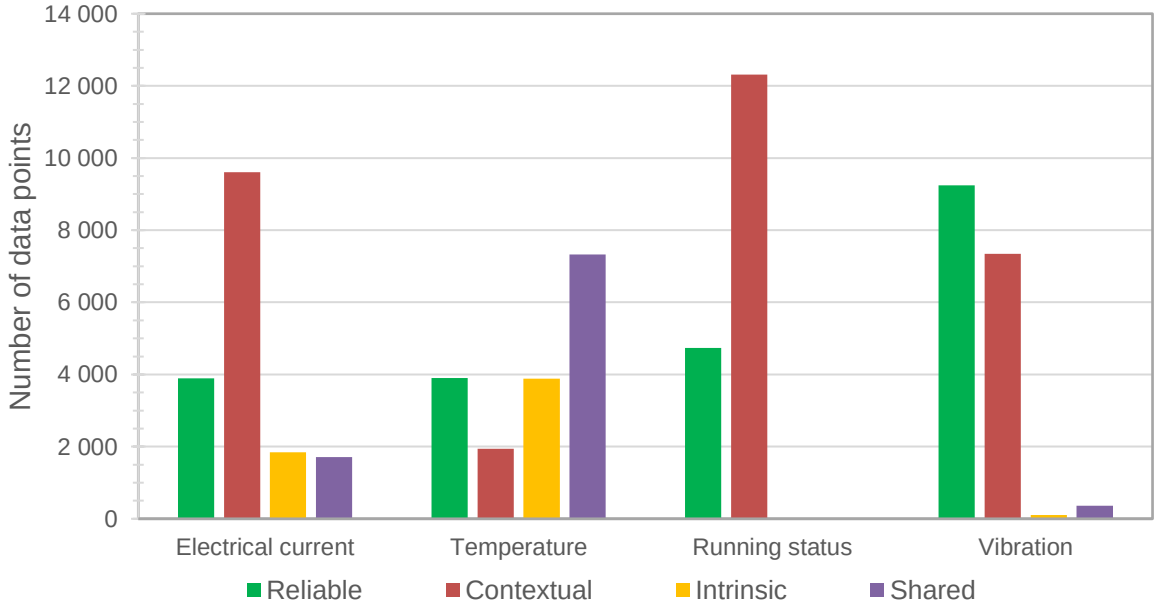

Figure 100: Component P reliability results breakdown

Figure 101 displays the characteristic diurnal profiles for 25th September 2020 to illustrate the data failures displayed in Figure 100. The running status indicates that the component was in operation for the entire day, excluding 17:30 to 21:00. Looking at the other three characteristics, it is implied that the component was actually in the *off* state. Despite the other characteristics contradicting the running characteristic, the changes in the running status do not reflect in any of the other characteristics. This brings the reliability of the running status measurements into question.

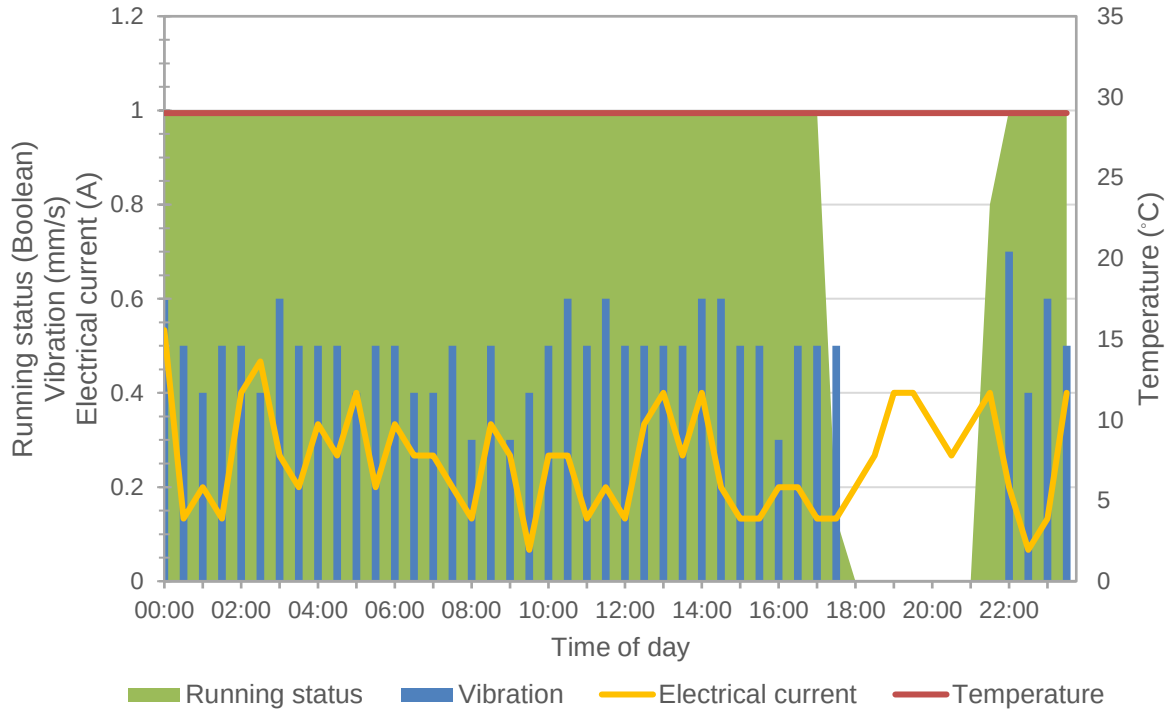

Figure 101: Diurnal profiles for Component P (25 September 2020)

Evaluating the electrical current drawn, it is noted that the value is not in the expected region of Component P when it is in the on state, implying that the component is in the off state. However, the values are also not zero, implying that the measuring equipment was not calibrated correctly.

The temperature profile flat-lines on a value, which is unexpected for a temperature profile. This brings the reliability of the measurements into question. It appears that the temperature sensor was disconnected from the component and was registering to a floating value.

C-17: Component Q

Component Q contains 35% unreliable data, as displayed in Figure 102.

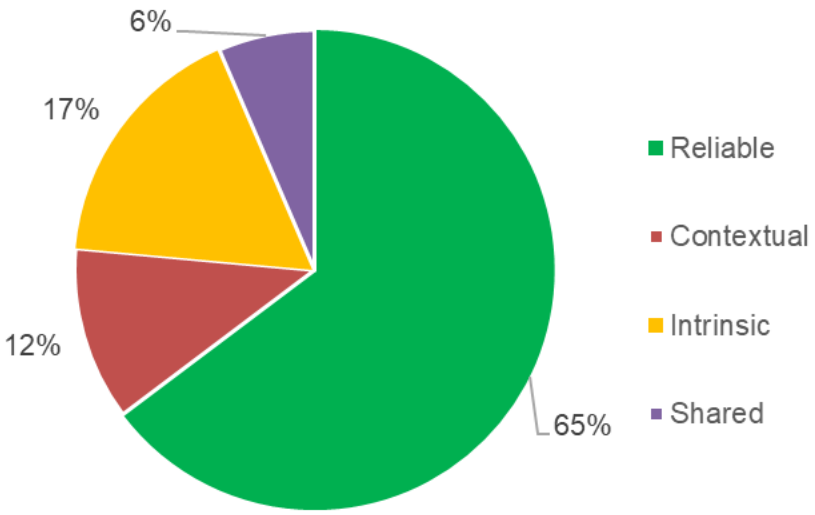

Figure 102: Component Q reliability results percentage breakdown

Figure 103 highlights the breakdown of data point reliability classification. The intrinsic methods classify many unreliable data points for the electrical current characteristic. Upon investigation, it was found that the electrical current data stream contains a lot of static, negative values. This would imply that the measurement equipment might have disconnected from the component on multiple occasions and was incorrectly calibrated.

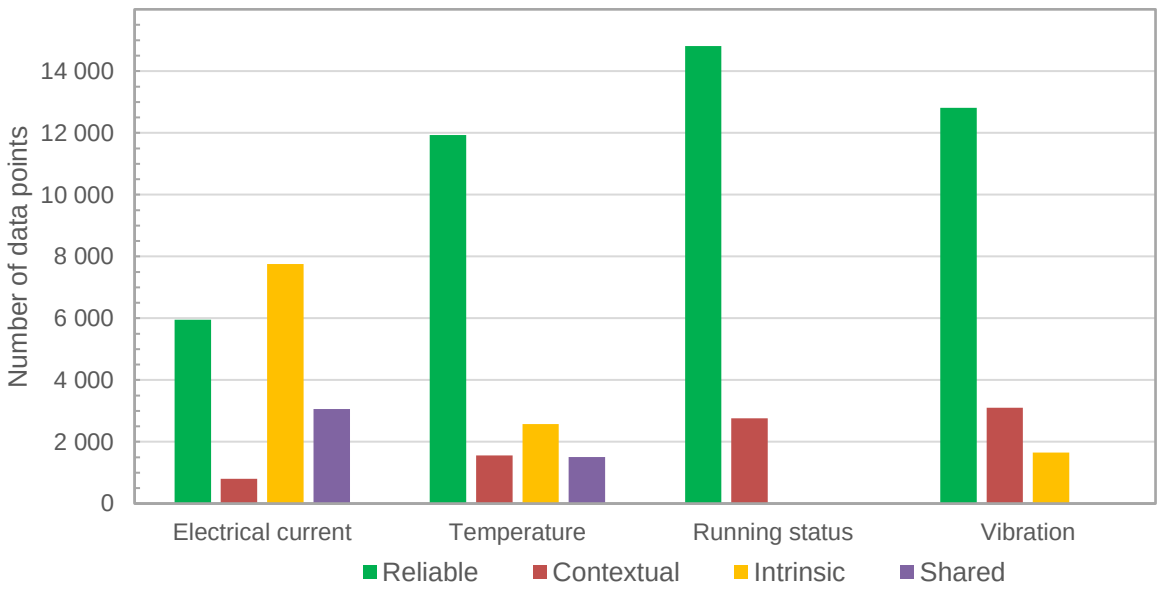

Figure 103: Component Q reliability results breakdown

C-18: Component R

The data reliability of the data streams linked to Component R is displayed in Figure 104.

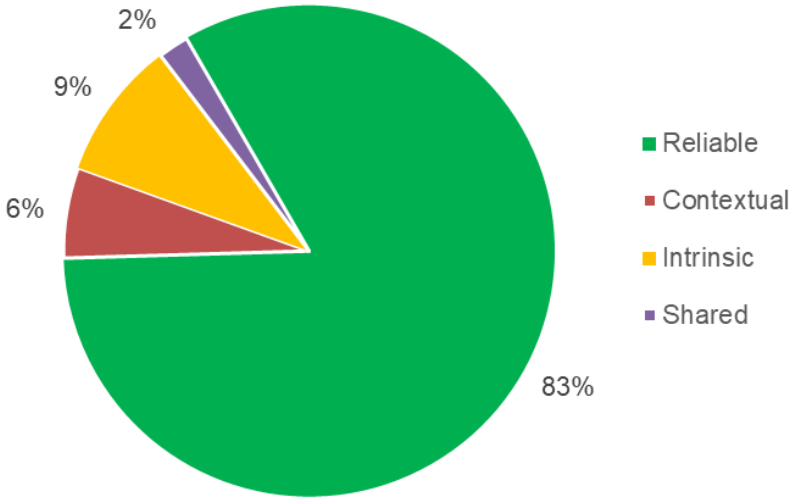

Figure 104: Component R reliability results percentage breakdown

Figure 104 was analysed from a characteristic perspective. Figure 105 illustrates that the electrical current characteristic contains a large amount of unreliable data identified by the intrinsic methods. During a further investigation, it was found that the measurements exceeded the upper limit of possibility defined for the component. Due to the number of such instances, it was concluded that the upper limit was incorrectly configured. Data points were thus flagged as unreliable that might otherwise be reliable when the component limits were correctly configured.

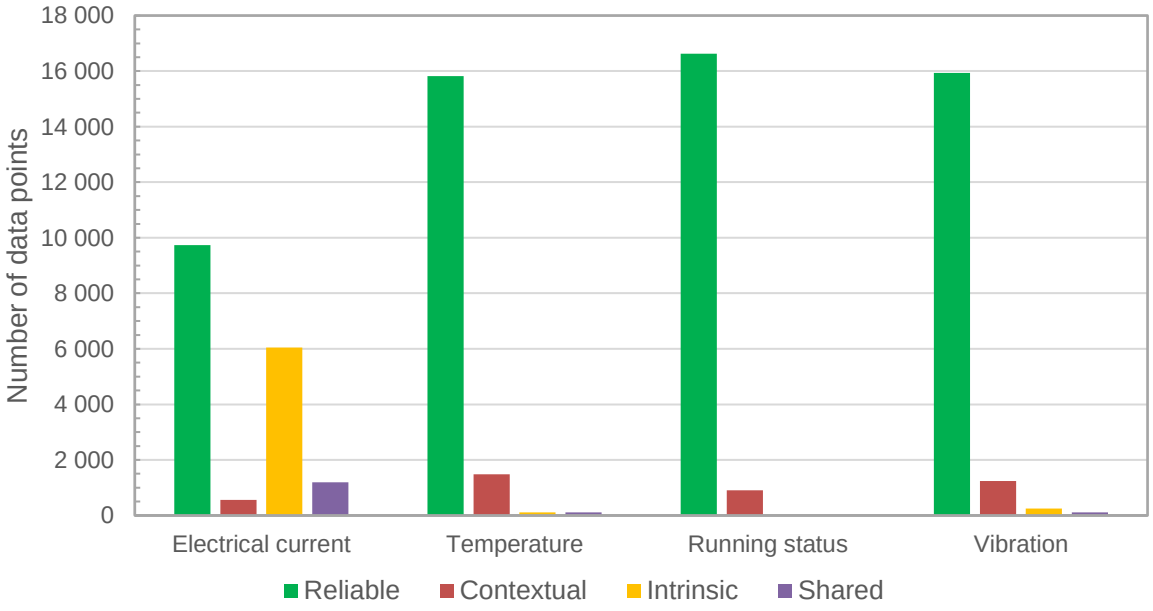

Figure 105: Component R reliability results breakdown

C-19: Component S

Figure 106 illustrates that unreliable data makes up 28% of the available data for Component S.

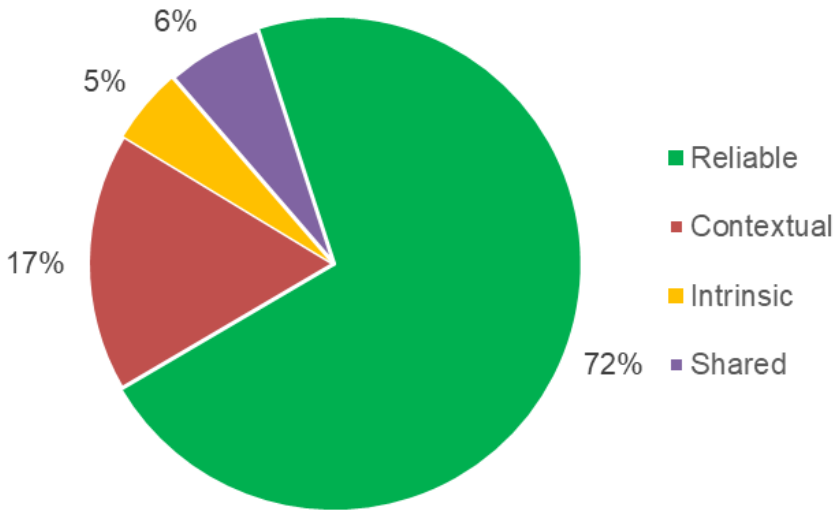

Figure 106: Component S reliability results percentage breakdown

Figure 107 displays the distribution of data reliability across the different characteristics. From this figure, it can be seen that unreliable data are present in all four characteristics.

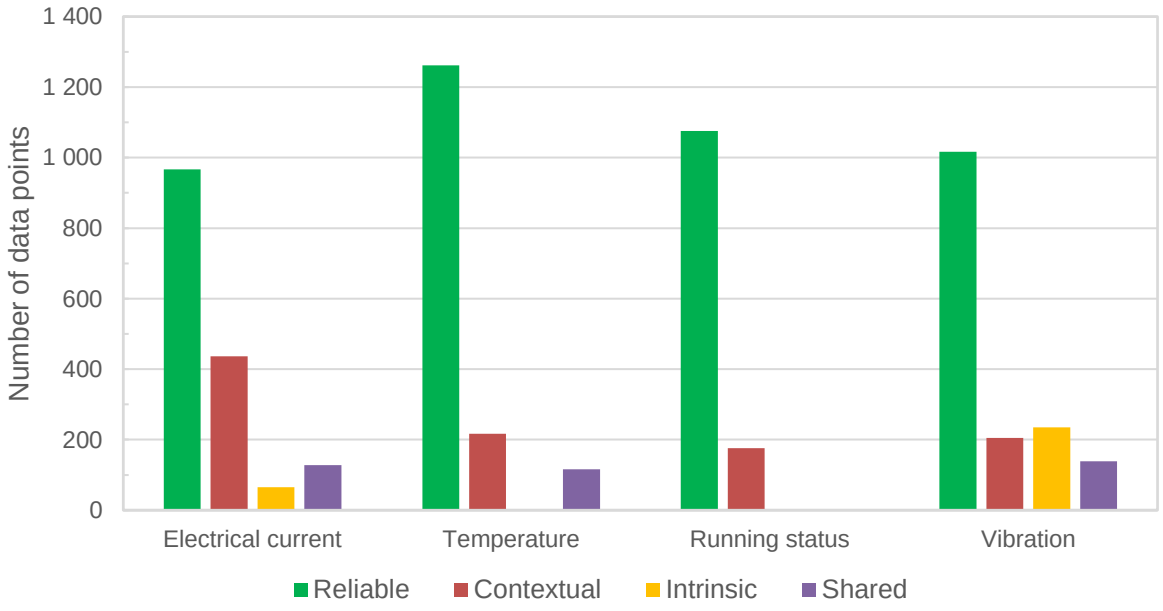

Figure 107: Component S reliability results breakdown

C-20: Component T

The data reliability results for Component T are displayed in Figure 108. Contextually unreliable data make up the largest portion of unreliable data.

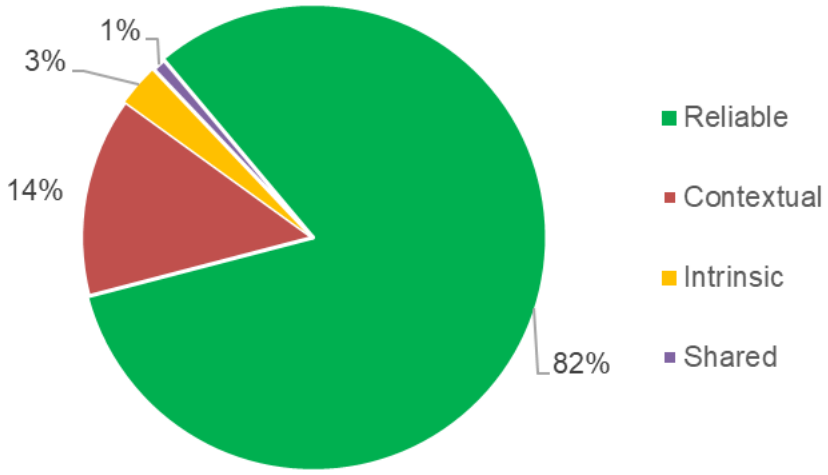

Figure 108: Component T reliability results percentage breakdown

Figure 109 illustrates how data reliability was classified on a characteristic level.

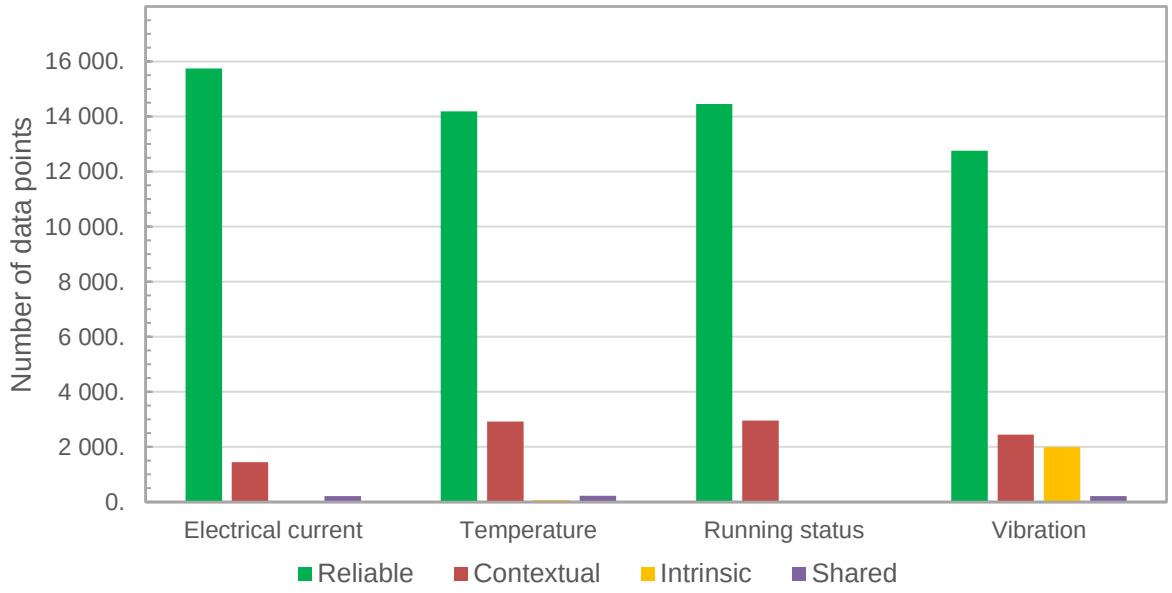

Figure 109: Component T reliability results breakdown

C-21: Component A data resolution

To investigate how data resolution impacts the accuracy of the proposed system, the system evaluated data for the same time range on 2-minute and 30-minute resolution data. The measurements were recorded at 2-minute intervals. An equivalent 30-minute resolution data stream was created by averaging the data over 30-minute intervals.

The percentage breakdown of the integrity for the 2-minute dataset is displayed in Figure 110, which illustrates that only a small percentage of data was identified as unreliable.

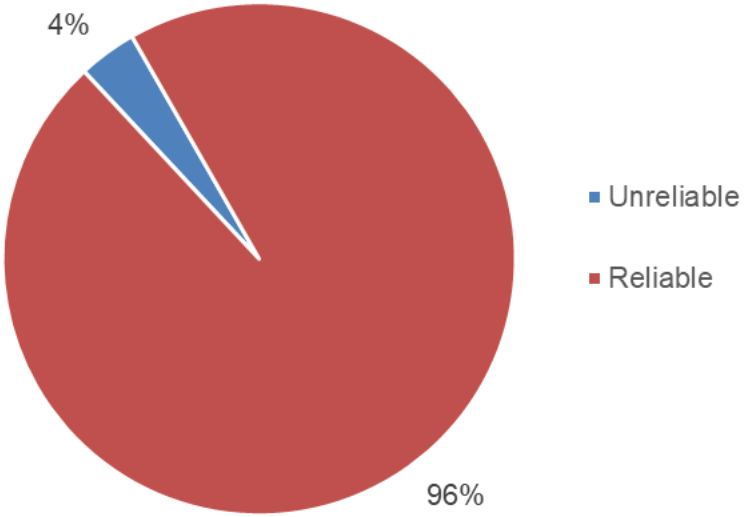

Figure 110: Two-minute resolution dataset reliability percentage results

A detailed breakdown of the characteristics is illustrated in Figure 111.

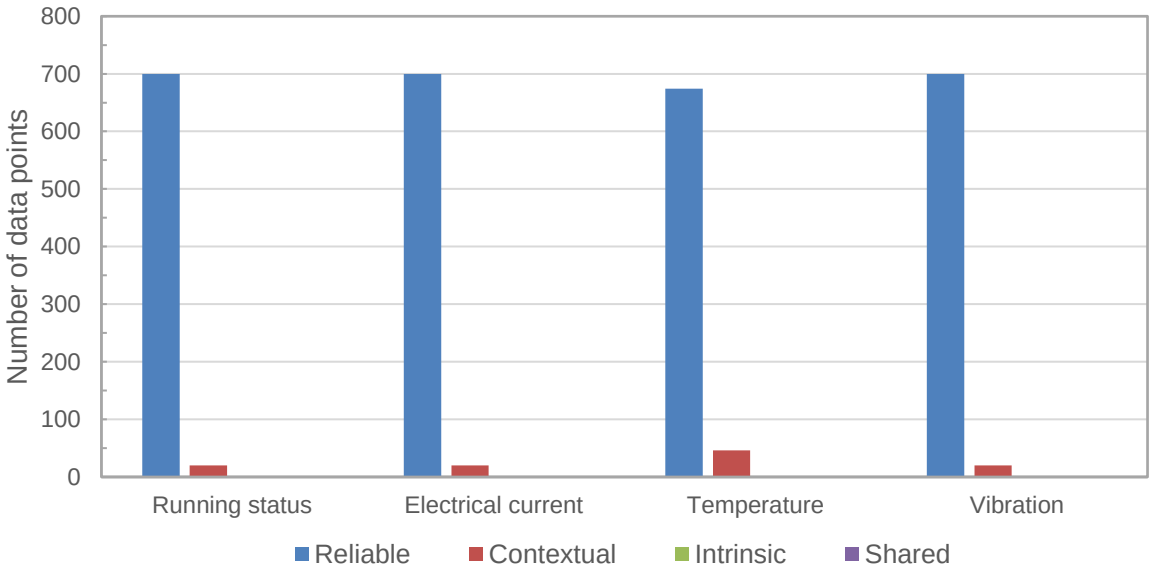

Figure 111: Two-minute resolution dataset reliability results

In contrast, Figure 112 displays the percentage breakdown of data integrity for the 30-minute resolution dataset. The 30-minute resolution dataset has a five times higher percentage of unreliable data points (10%) than the 2-minute resolution dataset.

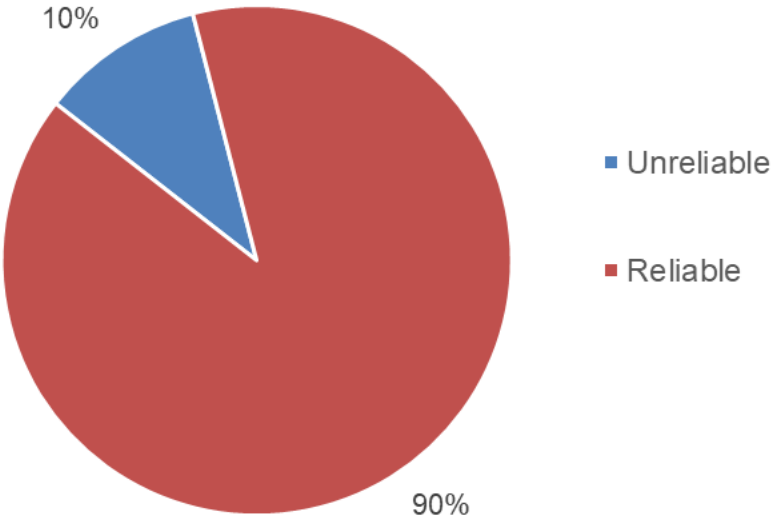

Figure 112: Thirty-minute resolution dataset reliability percentage results

A detailed breakdown of the results is displayed in Figure 113.

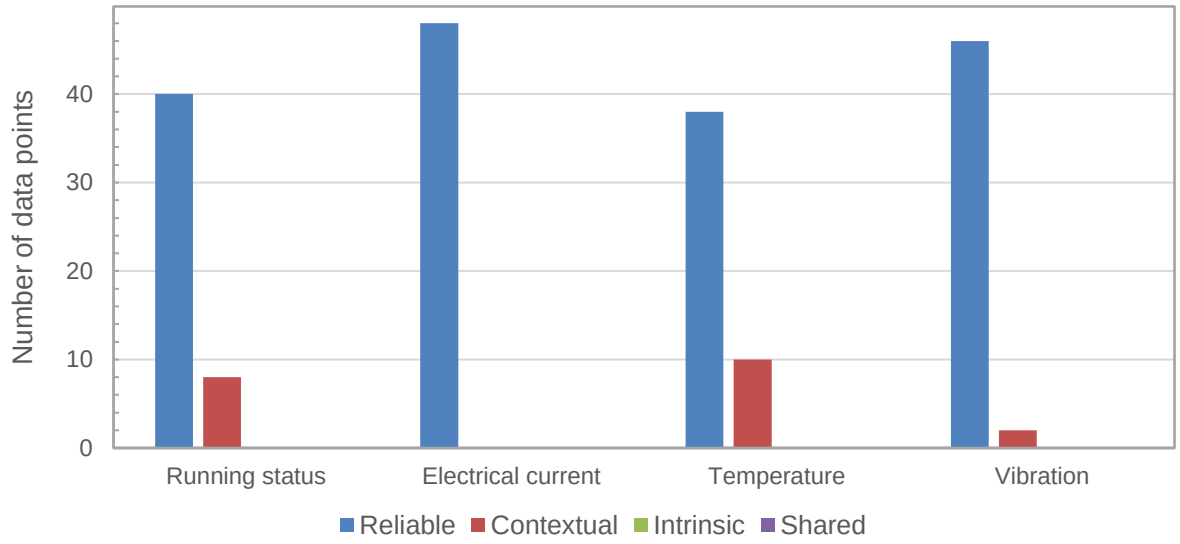

Figure 113: Thirty-minute resolution dataset reliability results

By comparing the results from the 2-minute and 30-minute resolution datasets, it can be concluded that the higher resolution data yields an increased percentage of reliable results, ultimately resulting in more accurate system analyses.

– — O — –