Enhancement of algorithm for detection of gold strip circuit vessel sensor errors

ENHANCEMENT OF ALGORITHM FOR DETECTION OF GOLD STRIP CIRCUIT VESSEL SENSOR ERRORS

By

Eduardo Pimenta de Melo

A Thesis Submitted in Partial Fulfillment of the Requirements

for the degree of

Master of Science

in

Mining Engineering

University of Alaska Fairbanks

August 2019

APPROVED:

Rajive Ganguli, Committee Chair
Tathagata Ghosh, Committee Member
Sampurna Arya, Committee Member
Tathagata Ghosh, Acting Chair
Department of Mining and Geological Engineering
William Schnabel, Dean
College of Engineering and Mines
Michael Castellini
Dean of the Graduate School

Abstract

Sensors are used to understand the condition and flow of mineral processes. Having accurate and precise information is fundamental for proper operation. Even small errors are relevant to cost when considering the operational span of a mine. Finding small errors is hard; few algorithms can detect them and fewer still, when considering errors on the scale of 2% in magnitude.

Some tools have recently been developed using data mining techniques for detecting small errors. Rambabu Pothina (2017) created an algorithm for detecting small errors in strip vessel temperature sensors in the carbon stripping circuit in Pogo mine. The algorithm performed well and was able to detect small magnitude errors without disrupting the industrial process. This thesis improves the understanding of the performance of the algorithm, while also making some minor changes.

First, a statistical analysis of the results of the algorithm on baseline data revealed an inherent difference in how the carbon strip process was run with respect to the two strip vessels. This discovery provided insight into the algorithm, and how its performance depended on process characteristics.

Second, the error detection algorithm was tested under scenarios different from Pothina (2017). Three separate types of errors were artificially added to real data: a) a fixed 2% ("fixed" error increase) b) a fixed 2% decrease ("fixed" error decrease) and c) an error with a mean value of 2% of magnitude ("noisy" error). Additionally, error was added to temperature data from each strip vessel (rather than just one), though only one at a time.

The algorithm was tested under each scenario for each of the four years, 2015, 2016, 2017 and 2018. The time to detect errors ranged from 19 to 73 days. The time to detect was very high (53 to 73 days) in 2017 since there were large data gaps that year. In general, time to detect was about 30 days. The performance under noisy error were not that far below fixed error scenario. The algorithm took 10% more time to detect errors under noisy error scenario compared to fixed error scenario.

On average, the algorithm detected an error after 25 cycles, regardless of the time span this represents. This is consistent in years with plentiful data, such as 2015, as well as years with less data, 2017 and 2018. In years with data gaps, 25 cycles represent a longer time period.

Seeded errors that decreased vessel temperature have very similar results to its equivalent increase, i.e. the decrease in 2% of S2 has results similar to the increase of 2% in S1 and vice versa.

In conclusion, these additional testing and analysis helped develop a more comprehensive understanding of the behavior of the data and the algorithms. These results validate and strengthen the findings of Pothina (2017).

Dedicated to

My Father, my Mother, my Brother and my friends,

The advisors for their guidance

And

To God

Nec Flama
Nec Tempestas
Nec Pulvis
Nec Timor

Contents

List of figures

List of Tables

Acknowledgments

First and foremost, I must acknowledge the help, understand and guidance provided to me by my academic advisor, supervisor, and the committee chair, Professor Rajive Ganguli, his selfless help has been the main reason this thesis is completed. His guiding hand has helped me thought this thesis, through the worst of the winter and most important, through the worst of my depression. Likewise, I owe my deepest gratitude to the members of the committee, Dr. Tathagata Ghosh, Dr. Sampurna Arya, for their invaluable advice and critique throughout the course of my work.

I am also indebted to the Pogo Mine as they provided all the data required for this work, they have allowed all the necessary site visits and information that this work required.

I would also like to thank the University of Alaska Fairbanks: Department of Mining and Geological Engineering, Department of Labor (DoL) and the Mining Engineering Endowment for providing the financial support during my studies, without it I would not be able to complete my studies successfully. I would also like to thank the Pogo Mine and Fort Knox Mine for funding the Endowment.

I also would like to acknowledge the help from Fadi El Didi and Eduardo Pacheco for the help in finding solutions for some of the codding conundrums I found myself stuck during this research.

Finally, I would like to express my gratitude to my mother, my father, and my brother, they helped me keep going when I thought it was impossible to continue. There are no words that can define how vital your actions were for the completion of this work. I am also thankful for the help from the International Programs of UAF, Graduate School, and College of Engineering and Mines, who helped me along the way and made me feel a part of the UAF family.

Without the help from all these people, I would not have come so far.

I would also like to thank God, for giving me a guideline through all this.

My profound and sincere thanks to all, I am indebted for life.

Chapter 1. Introduction

1.1. Background

Process control is an important part of mineral processing. It has become one of the most effective investment capital for a mill (Mular L., Halbe, & Barratt, 2002). In order to ensure an accurate process control procedure, data is needed, and it must be accurate. This data can only be gathered, in a continuous, non-disruptive manner through sensors.

Technologies in the industrial world were born with sensors: the measurement tools have now reached greater levels of automation and optimization. The mining industry has also embraced this innovation. Due to larger variability of mineral matter content, the sensor data has become a key part of mineral processing. Thus, sensors are today a major component of the industry.

Recent studies indicate that for a moderately sized mine, the economic benefit of sensor data ranges from $10 to $100 million dollars (Buxton & Benndorf, 2013). This economic benefit has resulted in exploration of even lower grade deposits, and made new mines economically viable.

This benefit, however, comes with some vulnerability. Bad sensor data can and does lead to losses in revenue. The oil industry, which faces similar challenges has lost what is estimated to be between 3% to 8% of all the United Sates oil production. This amounts to a $ 20 billion in annual revenue (Hong, 2009).

The amount of investment needed for installing sensors system is not modest. The capital cost for the installation are estimated to be somewhere between of 2-8% of the fixed operational costs in each process in a plant (Narasimhan & Rengaswamy, 2004).

All these figures do not account for other operational costs that comes with the sensors: such as its calibration and error repairing, which demands for a part of the operation to be stopped and it leads to major operational losses.

There is also the human cost of faulty sensors. Accurate sensor data is fundamental for safer working conditions. Faulty temperature sensors in underground mines also leads to unsafe work conditions and direct OSHA violations.

The common solution for these issues includes scheduled system calibrations and regular checks for any sights of malfunction. This is done to improve the quality of generated data, but it has the drawback of having to check those sensors who also are working properly. Nevertheless, there is no way of remote validation of the data. The whole operation must be stopped for checking sensor health.

Knowledge of the process play a large part in sensor monitoring and gross errors can be easily detected by experienced operators. Failures are easily detected during periodic checks. The major errors on the sensors are easily detected and corrected by the everyday techniques that are available to the operators.

Other sorts of errors such as small bias and small offsets from calibration are much more pernicious, even for an experienced operator, as small errors (such as a 2% bias on a sensor) cannot be detected. Despite being small, the small errors result in large losses as the errors go undetected for a long time. Thus, processes are run based on somewhat erroneous data for a long time.

The tools available currently only detect large (or gross) errors. Pothina, a researcher at the University of Alaska, presented a concept that uses a set of heuristic algorithms to detect calibration errors on temperature sensors in carbon strip vessels in the carbon-in-pulp (CIP) process at Pogo Mine (Pothina, 2017). In this document, Pothina refers to Pothina (2017).

Pothina's algorithm utilizes data from independent sensors that are process-related. In a given process, the numerous elements that exist between a different set of sensors can also be monitored and used to detect when a specific relationship, within a particular set of sensors, seems to be off. Pothina tested the algorithm assuming that errors were a fixed 2%.

This thesis analyzes the performance of Pothina's algorithm by designing additional error scenarios. It also analyzes the strip circuit data to better understand the performance of the algorithm.

1.2. Scope of Research

The present research focuses on select sensors in the CIP circuit in Pogo mine. Like Pothina, this research focuses only on small errors (2% in magnitude). Large errors are not part of the analysis.

In order to test the algorithm, faulty data will be generated and added to the real data. This will allow a known point of error to be analyzed by the algorithm. The time to detect the error will be measured.

1.3. Objective

This research focuses on assessing the robustness of Pothina's algorithm. While Pothina's work focused on errors in only one of the vessels, this work expands the algorithm to both vessels. Pothina's work also assumed that the two strip vessels were operated in an unbiased manner, i.e. their temperature and heat exchanger characteristics were similar. The algorithm relied on this assumption. This thesis will do a statistical analysis of the results of the algorithm to test this assumption. This thesis also uses two additional years of data (2017, 2018).

1.3.1. Challenges

The data provided was for the four-year period 2015 to 2018. As can be expected from any real process data, shutdowns and operational situations have resulted in sparse data on some occasions. The algorithm will be tested even in these occasions to verify its robustness.

Since there is no information if the sensors were calibrated, it is important to look at the patterns within the data. Therefore, this research proposes a full statistical analysis of the performance of the algorithm on different parts of the data.

1.4. Thesis Layout

This thesis is structured in chapters and each chapter will focus on the following themes:

Chapter 1: Introduction.

This chapter will present the problem statement and the structure of the thesis.

Chapter 2: Literature Review

Literature review for the themes covered in this thesis will be presented in this chapter. It will include an introduction to Pogo Mine, its industrial layout, description of the CIP circuit, sensor layout and its process. Pothina's algorithm will also be presented in this chapter.

Chapter 3: Research Methodology

A full description of tested algorithm will be presented in this chapter, along with the various testing scenarios.

Chapter 4: Results

Results of the testing are presented in this chapter, along with a discussion and interpretation of these results

Chapter 5 Conclusion

This chapter contains the conclusions reached from the work. Steps that are essential prior to testing the algorithm at the mine are also discussed.

Chapter 2. Literature Review

This section will present relevant aspects of the mine site, the algorithm to be used and literature on faulty sensors. It will also include an overview of the processing plant.

2.1. Research Location

The data comes from the Carbon in Pulp processing (CIP) circuit in Pogo Mine. The mine is located 70 miles (112 km) southwest of Fairbanks, near Delta Junction, on the Goodpaster River (Konigsmann et al., 2017). Pogo is an underground gold mine. The elevation range varies from 4,003 ft (1.220m) at the top of Pogo Ridge to 1,229 ft (396m) at the Goodpaster River. It is located in a Sub-Artic climate that has relatively mild summers and dry, cold winters (Konigsmann et al., 2017). The location is shown on Figure 2-1.

Figure 2-1 Location of Pogo Mine (source: Open Street Map)

Pogo is located at an environmentally sensitive area as it is adjacent to a salmon river habitat (Konigsmann et al., 2017). Therefore, to minimize environmental impacts, water is recycled as much as possible in their processes. This is also reflected in the cyanidation process, where the water utilized in this process is handled separately, in order to minimize the contact with cyanide.

An overview of the area can be seen on Figure 2-2, where observe the overall terrain and its infrastructure can be observed.

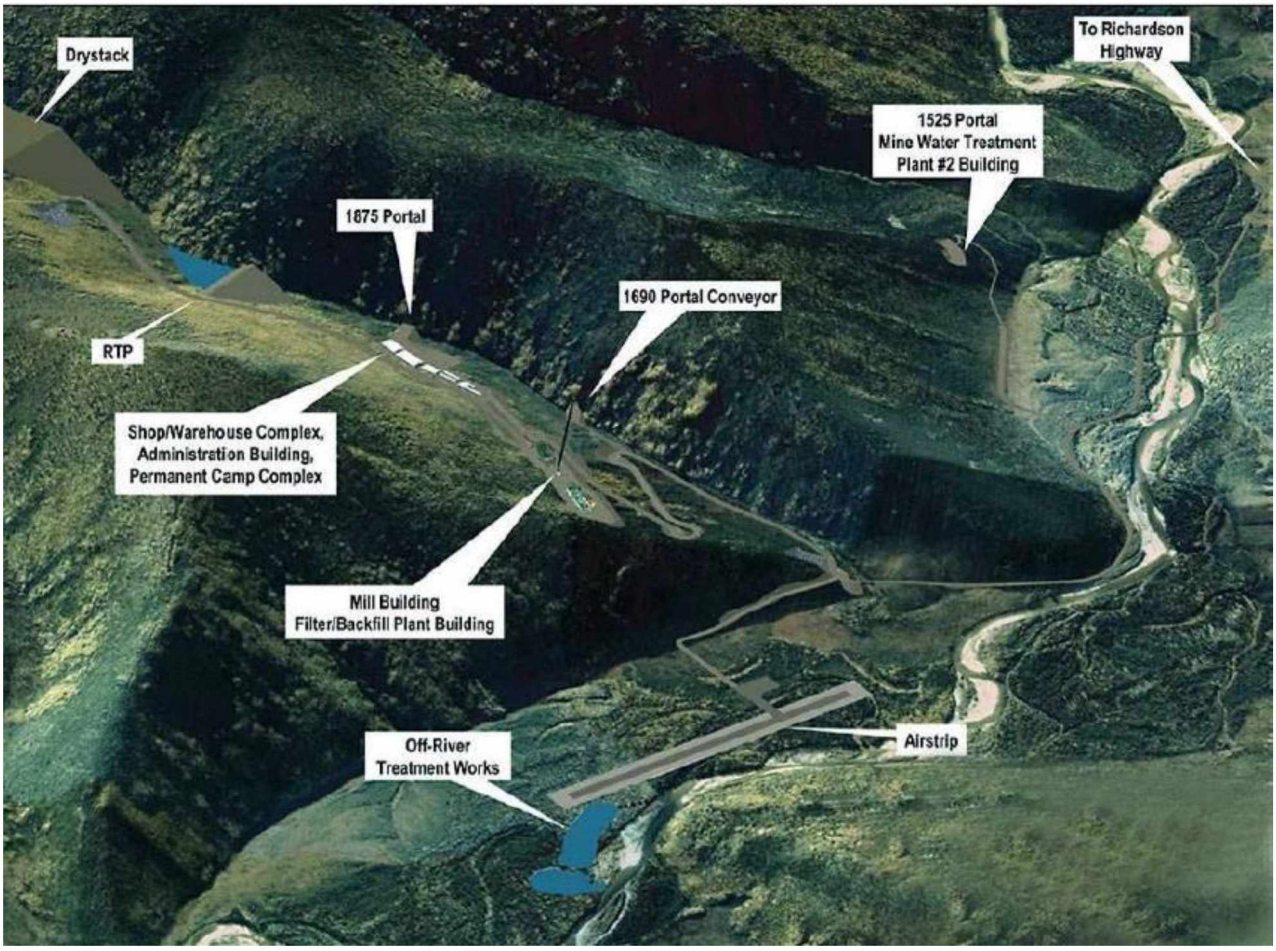

Figure 2-2 Pogo Aerial view (Konigsmann et al., 2017)

2.2. Mine and Processing Plant

Pogo Mine started operation in January 2005 and has extracted more than 3 million ounces of gold since its operation has begun (Pogo Mine, 2018a). It's a quartz vein deposit with an ore grade of 0.5 troy ounces of gold per ton of ore. This qualifies the deposit as high-grade (Pogo Mine, 2018b). The total probable reserves are estimated in about 5 million ounces of gold (Pogo Mine, 2018a). The deposit has two main veins (Figure 2-3).

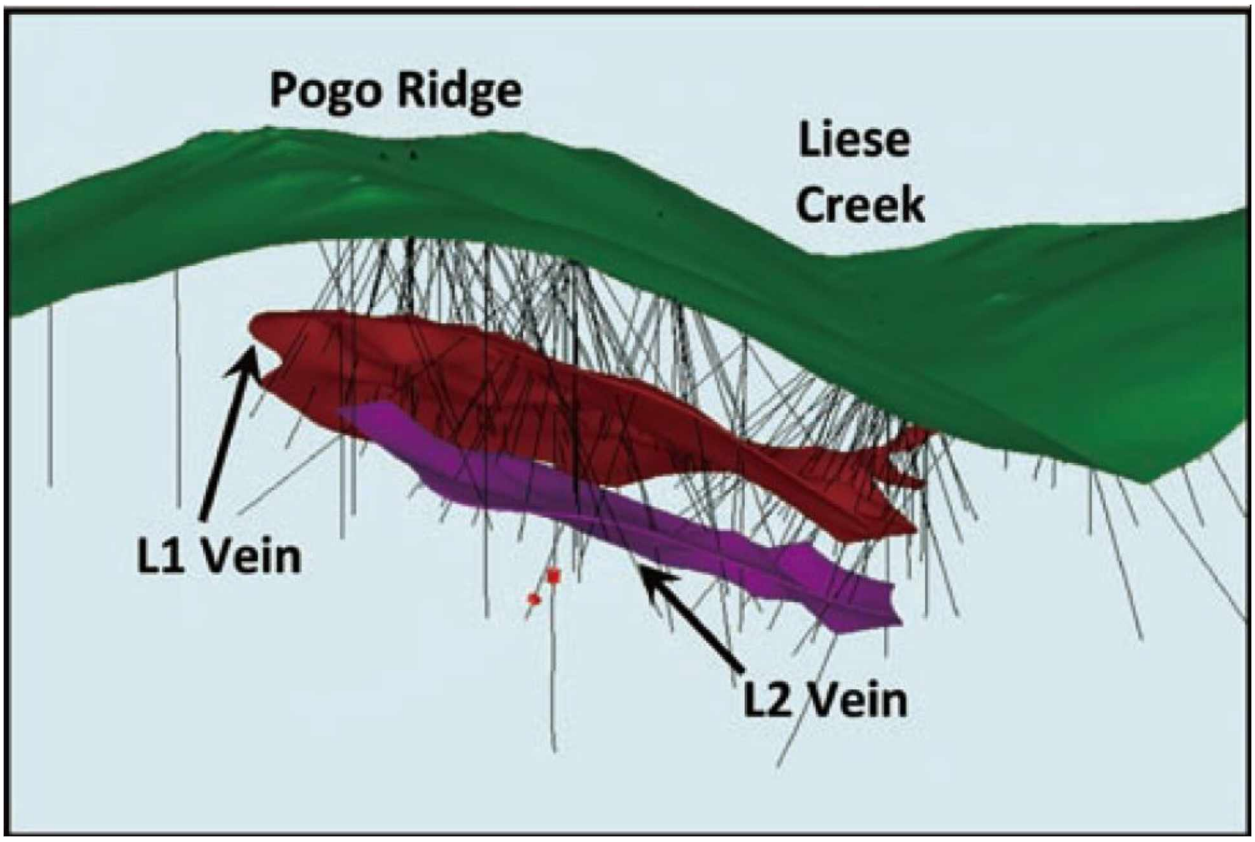

Figure 2-3 Three dimensional deposit model with drill holes at the time of the construction completion (Looking North).(Abe, 2010)

The mining method is *Drift and Fill*(Abe, 2010); the filling is done with a mixture of tailings with cement (paste backfill). A schematic for the method can be seen in Figure 2-4.

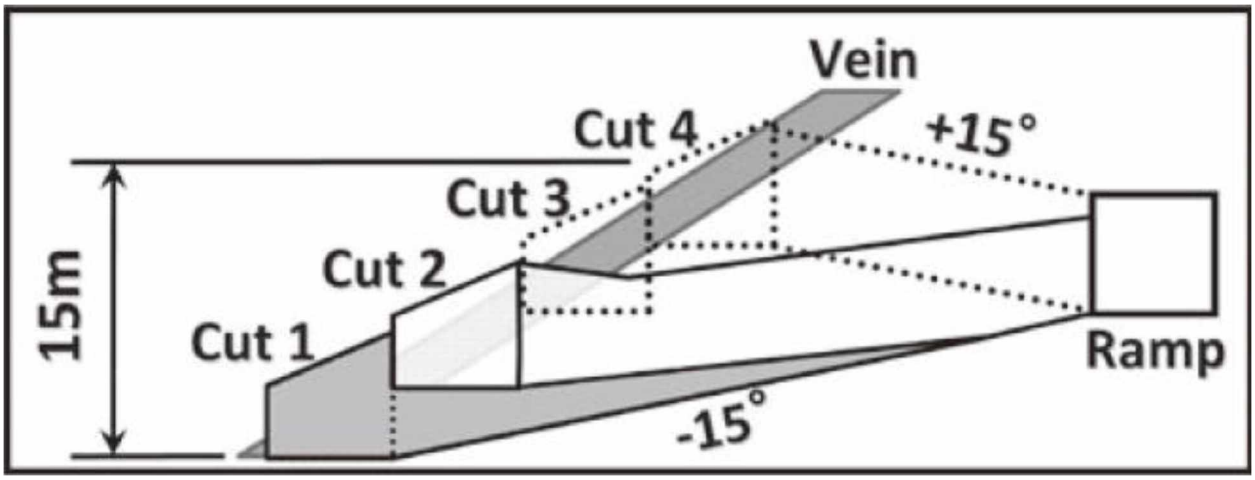

Figure 2-4 Drift and fill mining method (cross section view). (Abe, 2010)

Mineralization and Metallurgical studies have indicated that the gold present is contained in fine particles, with sizes ranging from 50 to 1 microns. Initial tests point out that up to 80% of this gold could be recovered by gravitational methods. During the pilot plant testing an average 67% of the gold was recovered through a similar circuit to the one installed on the mill (Konigsmann et al., 2017).

Due to these characteristics, the chosen comminution process is different from the usual. In order to maximize the recovery of all the material in the 50-1 micron range, the entire ball mill circulating load goes through the gravitational circuit. This was designed aiming an overall recovery rate of 60% of gold just by gravity concentration (Konigsmann et al., 2017).

The process flowsheet is presented on Figure 2-5. The flow displays a grinding circuit, a gravity concentration circuit, a flotation circuit, a leaching circuit, CIP and carbon handling circuit, a tailing and paste preparation circuit and a refining circuit. Gravity concentration also includes an intensive cyanidation phase that is done in batches. Each one of these circuits are going to be examined in more detail in the following sections.

Pogo Mine operates as a closed-circuit system. All water is reutilized and recycled. It is treated and cleaned before being released back into the pond. This is done in order to minimize potential environmental impacts that might arise from the operation. The mill recovery rate is around 67% (Pogo Mine, 2018a).

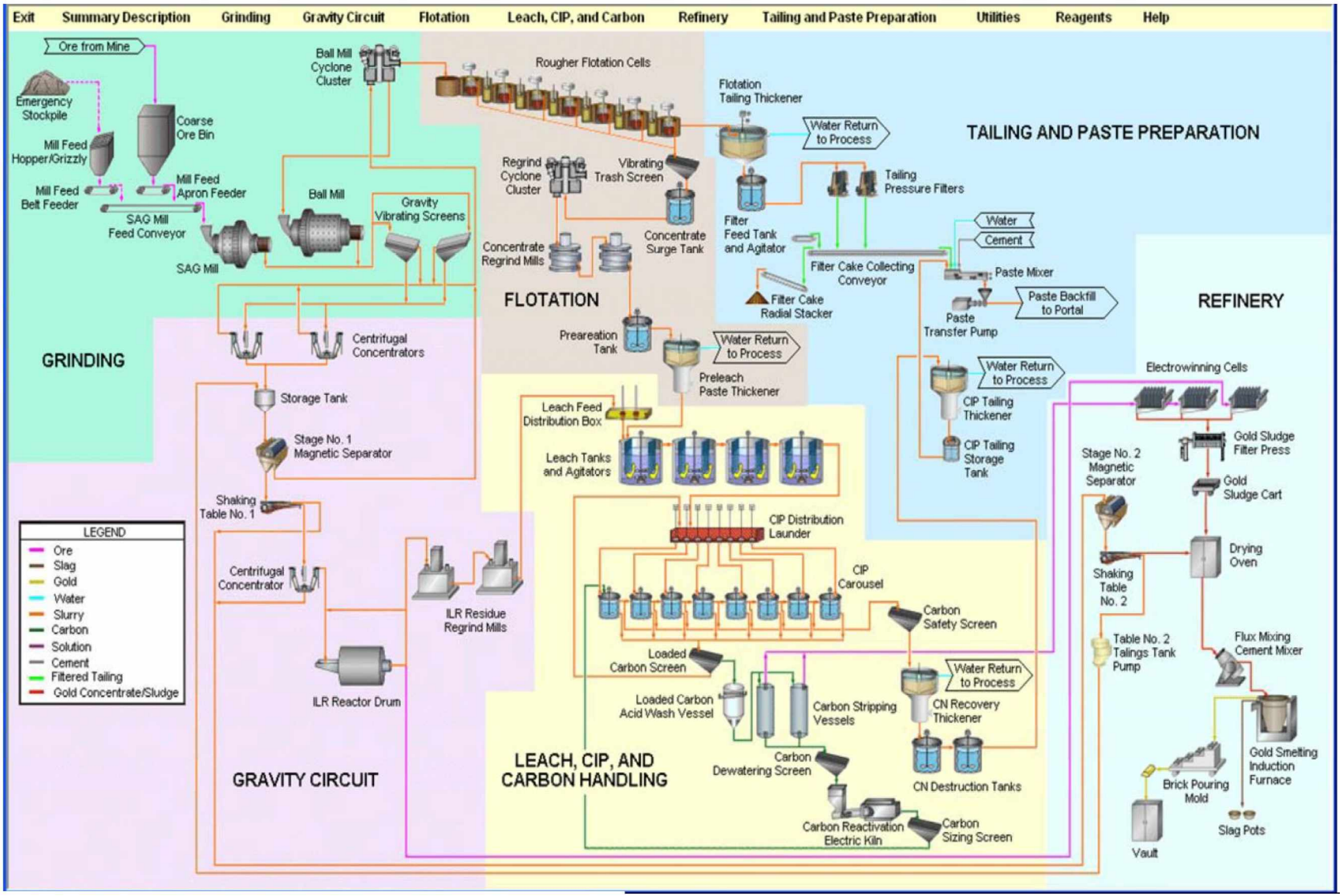

Figure 2-5 Pogo Mine Circuit Source:(Pothina, 2017)

2.2.1. Grinding Circuit

A scheme of the grinding circuit is presented in Figure 2-6. The Run of Mine (ROM) is fed to a SAG Mill, with the resulting material then screened at one (1) millimeter. The oversized material is directed to the Ball Mill Cyclone while the under size is sent to the gravity concentration circuit (A1 entrance in the Gravity circuit flowsheet).

The Cyclone cluster receives the oversized material and the material sent from Gravity Concentration (B1 entrance in the Gravity Concentration Circuit flowsheet). The fines are directed to the Flotation Circuit (A2 entrance in the Flotation Circuit flowsheet.

The Ball Mill receives the coarser material from the cyclone and directs it back into the vibrating screens.

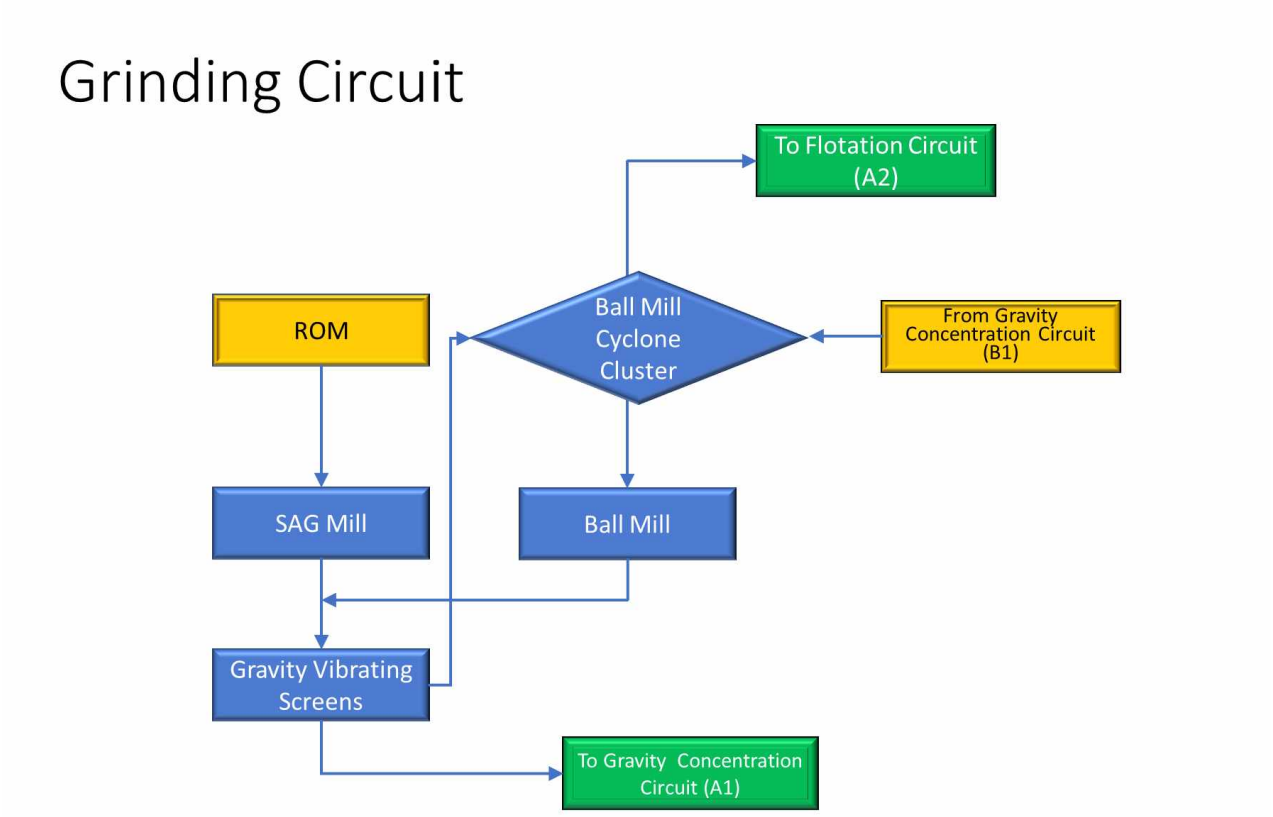

Figure 2-6 Pogo Grinding Circuit

2.2.2. Gravity Concentration Circuit

A Gravity Concentration Circuit schematic can be seen in Figure 2-7. It receives the finer portion of the vibrating screens. Two 48-inch centrifugal concentrators, working in parallel,

separates them by low and high density. High density material is directed to a storage tank, where it is mixed with material from the Refining Circuit (F2 entrance int the Refining Circuit flowsheet). Low density materials are sent back to the grinding circuit.

The material then goes through a magnetic separator. The magnetic material is directed back to the grinding circuit, and the non-magnetic is directed to the shaking table. The shaking table makes two partitions in the flow: (i) primary concentrate that is directed to the Intensive Leaching Reactor (ILR) and (ii) secondary concentrate that is directed to the Refining Circuit.

Originally, the tailings from the shaking table were directed to the intensive leaching reactor. The results were less than ideal and, so, the circuit was reconfigured to the present set-up. The need for the tabling circuit comes in part from the gold at Pogo being in the form of maldonite, a insoluble gold and bismuth alloy (Konigsmann et al., 2017).

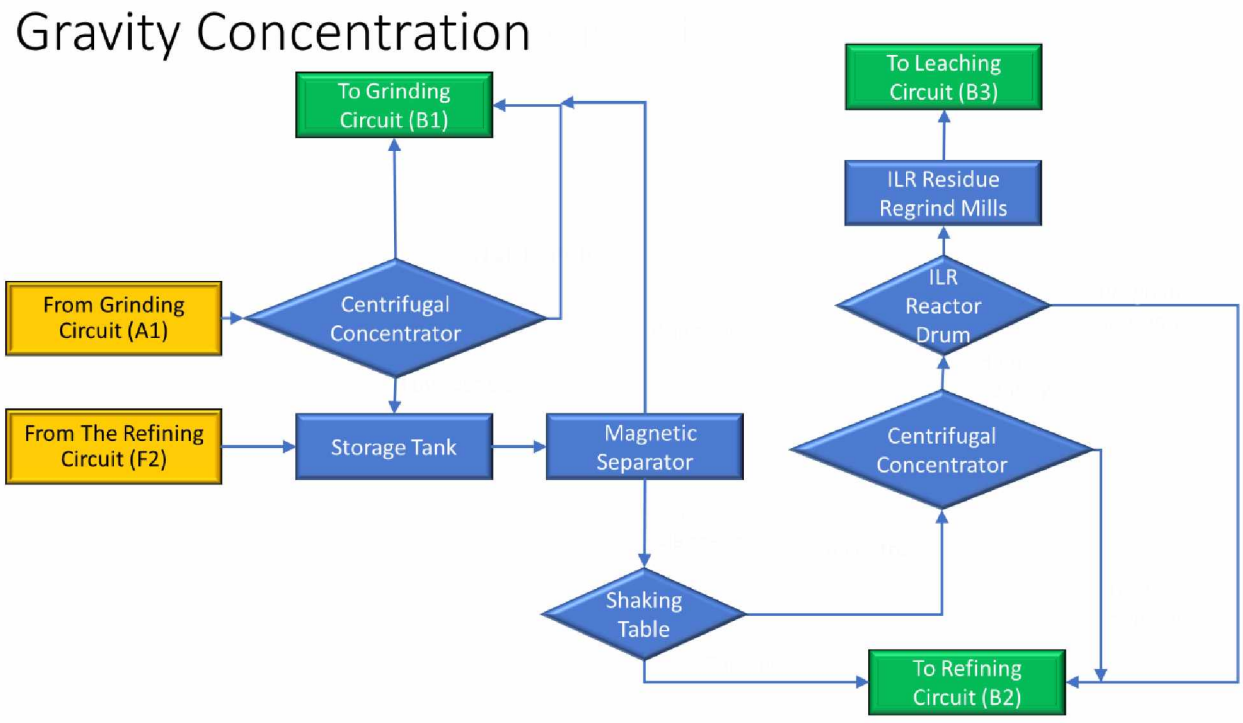

Figure 2-7 Gravity Concentration Circuit

The concentrate from the shaking table is directed to another centrifugal concentrator from where (i) high density material is directed to ILR leaching and (ii) low density is directed to the Refining Circuit. The high-density material is directed to the ILR Reactor Drum. The intensive

leaching process utilizes cyanidation, operating in batches. The drum is filled with the concentrate and then closed. A solution of cyanide at 2% circulates at a high flow rate to ensure a constant supply of oxygen and cyanide. The drum is then rotated as a common bottle roll technology. The rotation and high flow rates ensure a high shear on the particles surfaces, that improves the cyanidation. This allows for a high recovery rate. The pregnant solution is then clarified and directed to the refining process (Longley, McCallum, & Katsikaros, 2003).

The solid partition material is then reground and directed to the leaching circuit.

2.2.3. Flotation Circuit

Schematics for the flotation circuit can be seen in Figure 2-8. Ball Mill Cyclone Cluster overflow is directed to a sulfide rougher flotation circuit (A2 entrance int the Grinding Circuit flowsheet). This flotation aims to produce 10%, in weight, concentrate. The reject material is directed back to the tailings circuit for reprocessing and water recovery.

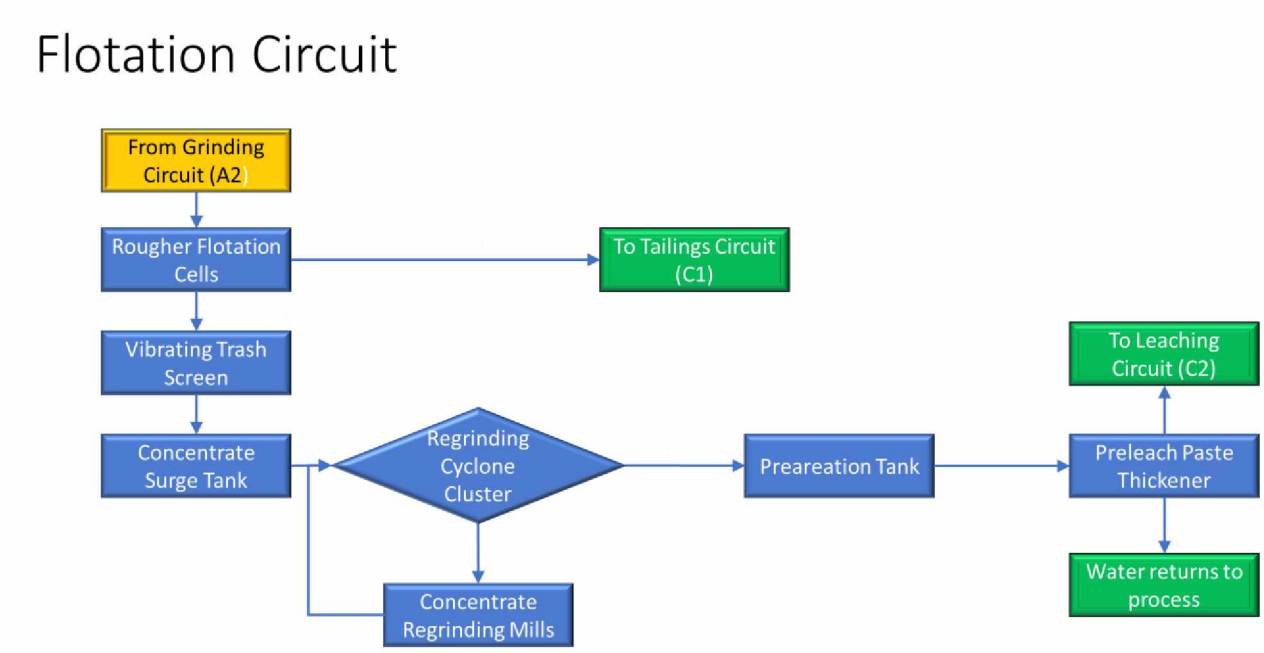

Figure 2-8 Flotation Circuit

The concentrate is pumped to a Cyclone cluster with a P_{80} of 10 microns. The overflow is directed to a pre-aeration tank and then a thickener, to recover the water. The resulting paste is directed to the Leaching Circuit.

2.2.4. Leaching Circuit

A schematic for the leaching circuit can be seen in Figure 2-9. The material directed from the Gravity Concentration Circuit and from the Flotation Circuit is fed to a distribution box. It will go through the cyanidation process in agitation tanks, where they are pumped through towards the CIP Circuit.

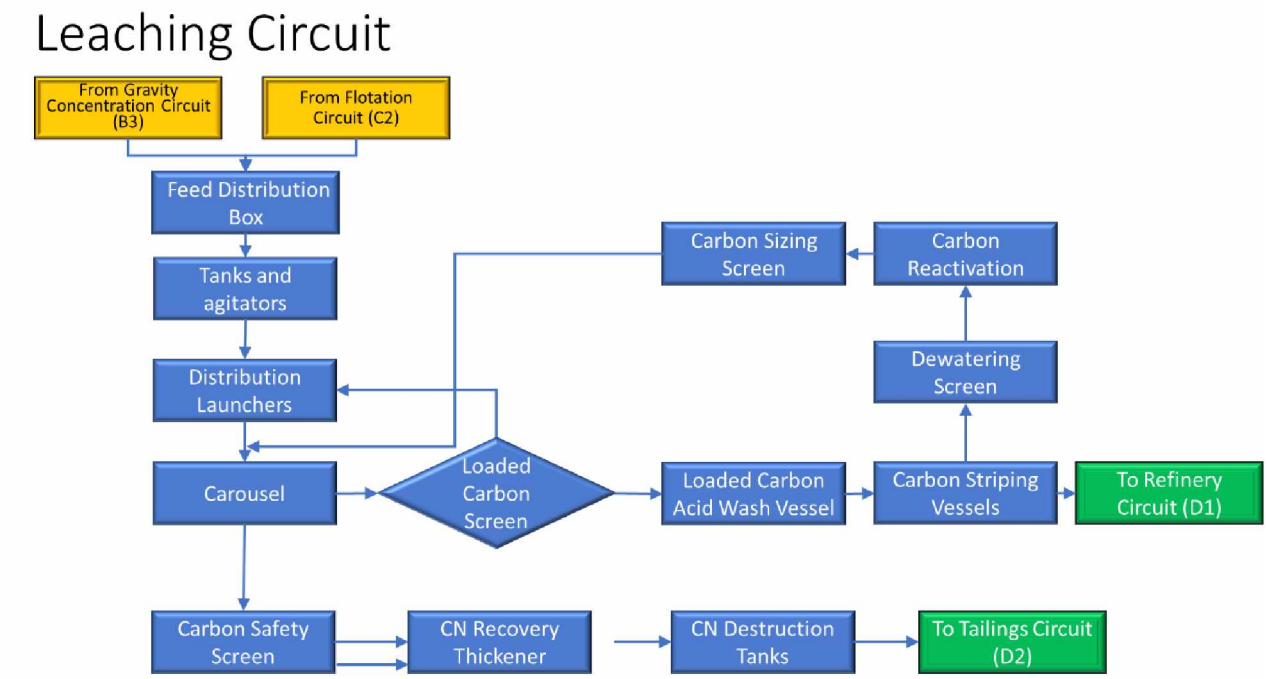

Figure 2-9 Leaching Circuit

2.2.5. Carbon in Pulp Circuit

The CIP process starts at the eight-cell carousel present in the plant. The CIP process aims to capture any gold present on this material. It does it by adsorbing gold particles into activated carbon. Activated carbon is nothing more than processed charcoal; the processing is made in such way that the resulting material has a high specific surface area. This surface area increases the chances of contact with fine particles and increases adsorption rates (Pothina, 2017).

The flotation concentrate goes through eight tanks, staying for 30 minutes in each one, for a total of 4 hours. This residency is necessary for adsorbing gold into the carbon. The amount of

gold adsorbed into the carbon ranges from 300-600 ounces of gold for every ton of activated carbon used. (Pothina, 2017).

The material goes through a screen that captures the activated carbon, with the underflow being redirected to the carousel. The liquid fraction goes through a carbon safety screen (to recover any accidental material directed there) and to a Cyanide (CN) Recovery Thickener where the water is salvaged for reutilization and the cyanide destroyed.

After it's residency in the CIP circuit, the carbon, now with gold adsorbed into it, is first acid washed and then directed to the stripping circuit.

2.2.6. Carbon Stripping Circuit

As shown in Figure 2-10, the stripping circuit works with two independent strip vessels connected in parallel to the main circuit. The process is made in batches. While one vessel processes a loaded batch of gold containing activated carbon, the other vessel is emptied and loaded with a new batch of material. The loading consists of filling the vessel with carbon that has adsorbed gold.

The stripping process requires the solution to be circulated for a cycle at high temperature and under pressure. Therefore, the solution is heated by Heat Exchangers 1 and 2. This processing operates in cycles.

A given batch usually takes roughly 11 hours to be processed, though this time is quite variable. It consists of the circulation of the heated solution in the vessel, which lasts about 8 hours, followed by loading and unloading of the material in a given batch, which lasts 3 hours. This process is called "Pressurized Zadra Stripping."

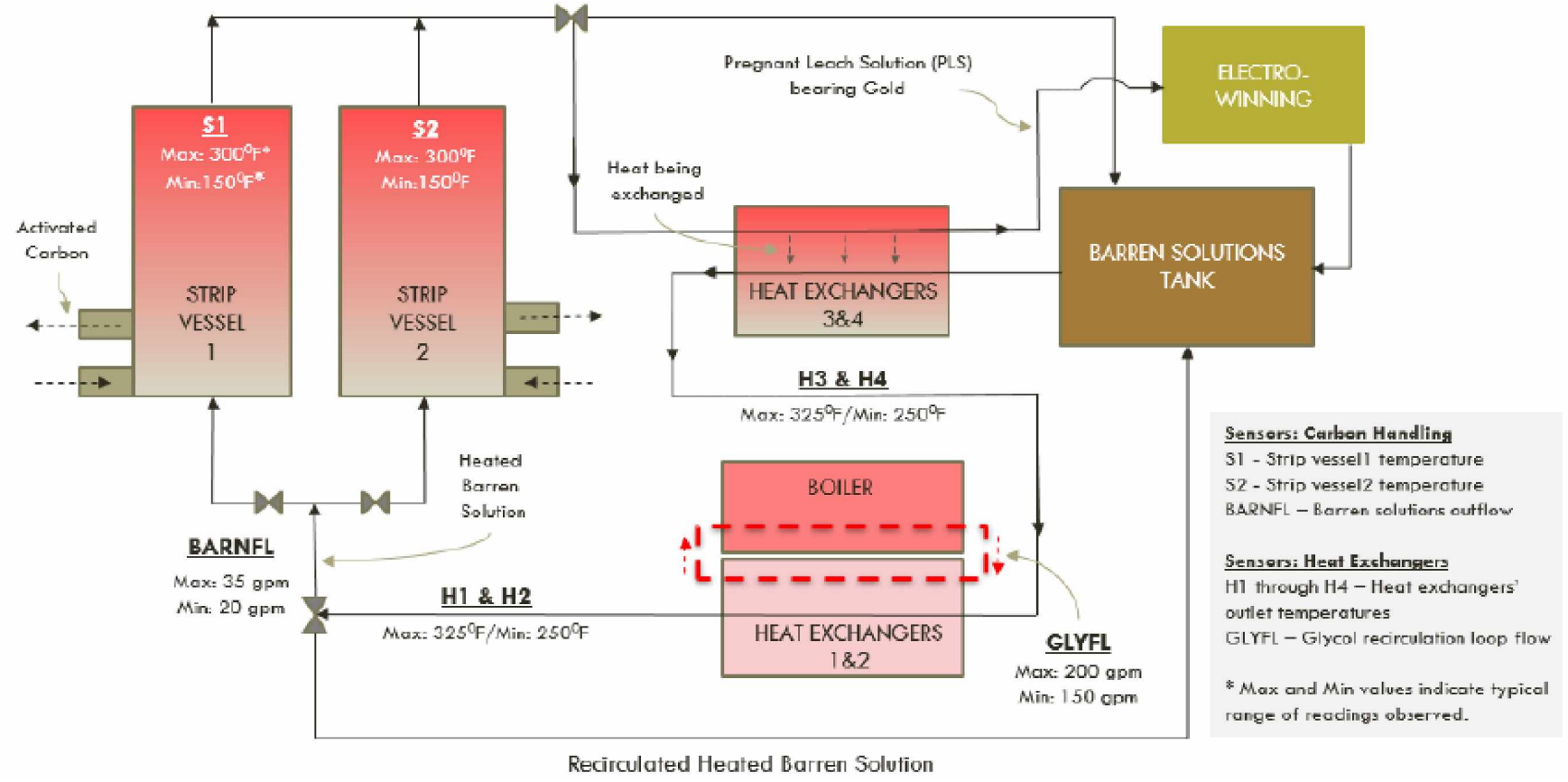

Figure 2-10 Leaching Circuit (Pothina, 2017)

2.2.7. Pressurized Zadra Stripping

The Zadra process was proposed by Zadra (Zadra, 1952), and it consists in a hot, caustic cyanide solution, originally at atmosphere pressure, being used to extract gold or silver particles from activated carbon. The hot solution, at 95-100°C, of 1% weight/volume (w/v) sodium hydroxide and 0.2% w/v sodium cyanide is recycled through the carbon loaded with gold in order to desorb it into $Au(CN)_2$. This process took 72h at these specifications. (Soleimani & Kaghazchi, 2008).

Given the maximum temperatures water-based solutions can achieve at room pressure, a derivation of this process has been created, where the vessel is put under pressure during the process, allowing a higher operational temperature, resulting in higher thermodynamic interaction and therefore less residence time (Muir, Hinchliffe, Tsuchida, & Ruane, 1985).

For the pressurized process, the usual solution is composed of 1% sodium hydroxide and 0.1% sodium cyanide (Fast, 1989). The usual temperature and pressure used are 280°F and 65 PSIG, allowing maximum liberation of gold particles. The active solution is then pumped through the loaded vessel. The heated and pressurized conditions allow the solution to reverse the adsorption, freeing the gold into the solution. This is made in a continuous form during the cycle.

In order to have optimal gold liberation, it is fundamental to keep the temperature within a certain range (270-280°F); otherwise the gold recovery will drop, and this will result in gold being discarded on the tailings(Pothina, 2017).

The solution, now full of gold (colloquially named "Pregnant Solution") is then pumped through Heat Exchangers 3 and 4. This is made to recover part of the heat, as it is not required in the electro-winning process. This allows the partial recovery of the heat invested in the process.

Afterward, the "Pregnant Solution" is pumped to the Refining Circuit, for the electro-winning process. The electro-wining circuit strips the solution of the gold particles. The resulting solution, now called "Barren Solution," is then pumped to the Barren Solution tank (Pothina,

2017). The electro-winning process also allows the recovery of the sodium hydroxide and sodium cyanide solutions, as it strips gold content and permits this solution1 to be used again.

Then, the "Barren Solution" is pumped through the first set of Heat Exchangers (3 and 4) for a first heating and pumped through the second set (1 and 2) for a final heating before being pumped one more time into the new vessel. The second set of heat exchangers is heated by a glycol solution that flows through them and the boilers.

Simultaneously with this process, the second vessel is filled with the carbon-gold solution. Once it's filled, the "Barren Solution" is pumped into a new batch. This periodic behavior can be gathered from the temperature data in the vessels.

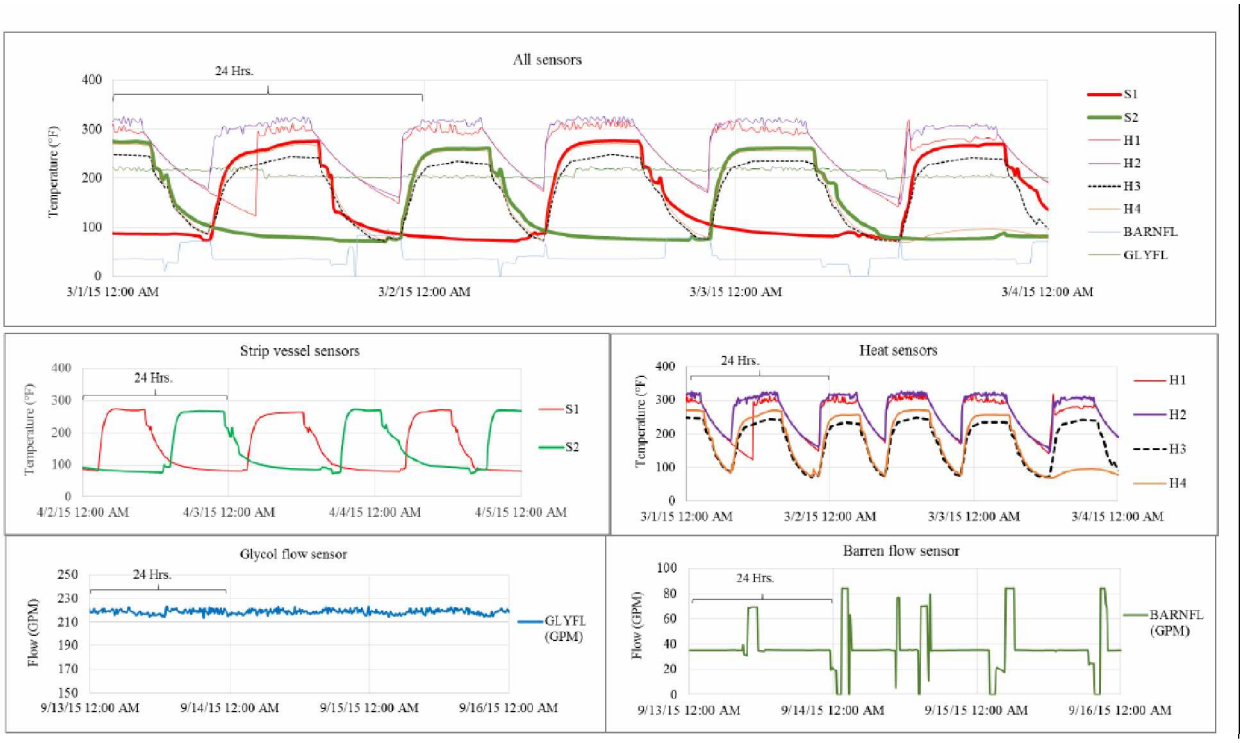

Figure 2-11 Sensor Behavior on the Strip Circuit(Pothina, 2017)

As seen in Figure 2-11, a cycle of the Strip Vessel 2 (S2 in green on the graph) starts only after the end of a cycle in Strip Vessel 1 (S1 in red on the graph), this periodic behavior, alternates between one active vessel to the other allowing a continuous stripping process.

The Operating schedule from a usual Zadra pressurized stripping is summarized in the following table.

Table 2.1: Operating Schedule-Pressure Zadra Stripping

Operation	Solution	Time
Load Column	Transfer water	90 minutes
Elution	0.1% NaCN, 1% NaOH	480 minutes
Carbon Cooling	Fresh Water	60 minutes
Unload Column	Transfer Water	30 minutes
Total		11 hours

Source (Fast, 1989)

2.2.8. Refining Circuit

The final gold processing is made in the Refinery Circuit. Since the gold is in two forms by the end of the process, there are two main flows in the refining circuit. The gold that comes from the gravity concentration circuit is already in a suitable form for the smelting process. The other, comes from the leaching process, and is an elute. The flowchart is presented in Figure 2-12.

This results in the need for an intermediary process before the smelting. This is made by electro-winning. Since the elute comes from a Zadra process, it has to be recirculated until the eluted carbon is low enough to be reactivated and returned to the adsorption circuit. The electrode in the electrowinning circuit allows the reduction of gold from $Au(CN)_2$ to Au and 2 CN^-. The pH is critical in this process, as the anode is oxidizing the OH^-. It is known that if the pH goes acidic this might result in the oxidation of the chromium present in the anode and the subsequent reduction of it in the cathode. This is known to inhibit the deposition of gold. Therefore, the pH needs to be closely monitored (Adams, 2005). After the electrowinning process, the gold sludge is taken to a drying oven.

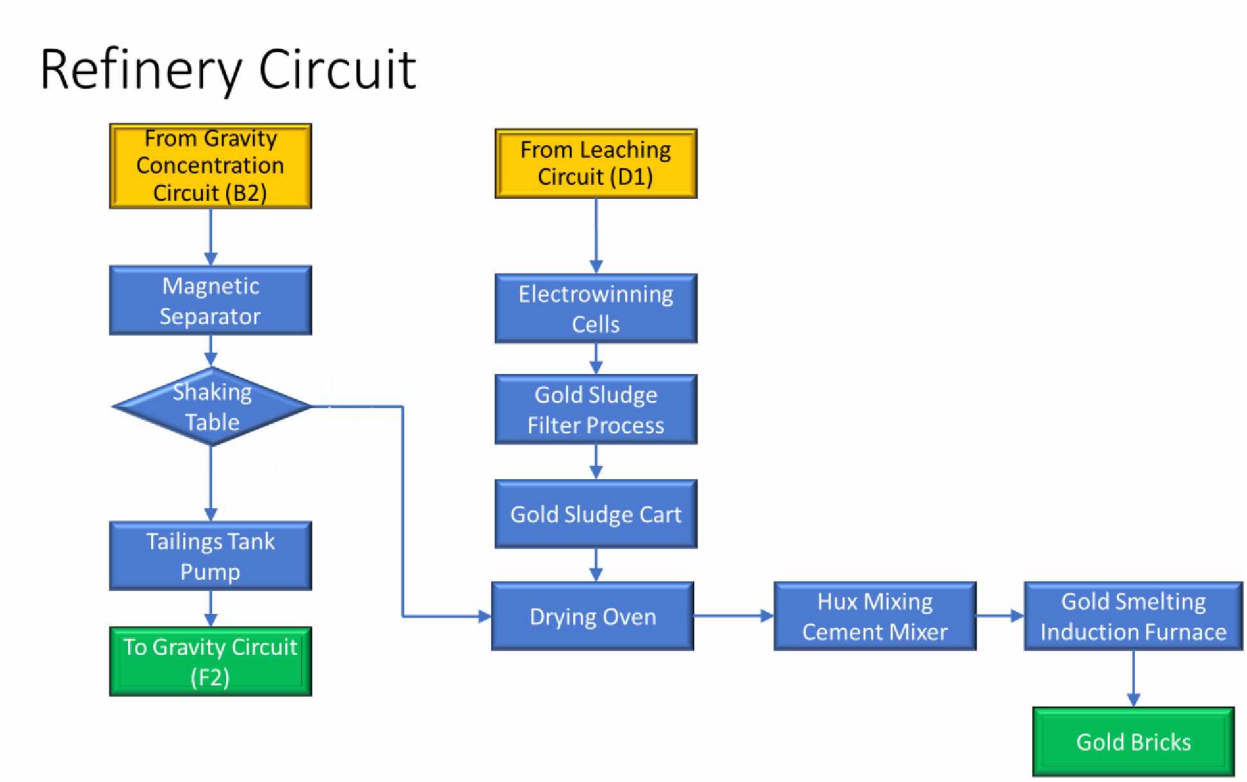

Figure 2-12 Refinery Circuit

The Gravity Concentration circuit feed is processed in parallel. First, it goes through another magnetic separation; this usually results in no magnetic material. This material goes into another shaking table, then, high density is directed to the drying oven, and low density is redirected to the Gravity Circuit.

The gold is mixed in the Drying oven and then, directed to the induction furnace, where the resulting gold bricks are recovered.

2.2.9. Tailings Circuit

In order to minimize environmental impact, all water is recirculated within the process, and cyanide tailings are processed with special care.

This is shown in the Tailings Circuit flowchart, as can be seen in Figure 2-13.Tailings from cyanidation are processed in separate, the flotation tailings are pressured, filtered, and the tailings are either stored in dry form or reutilized in the mining process.

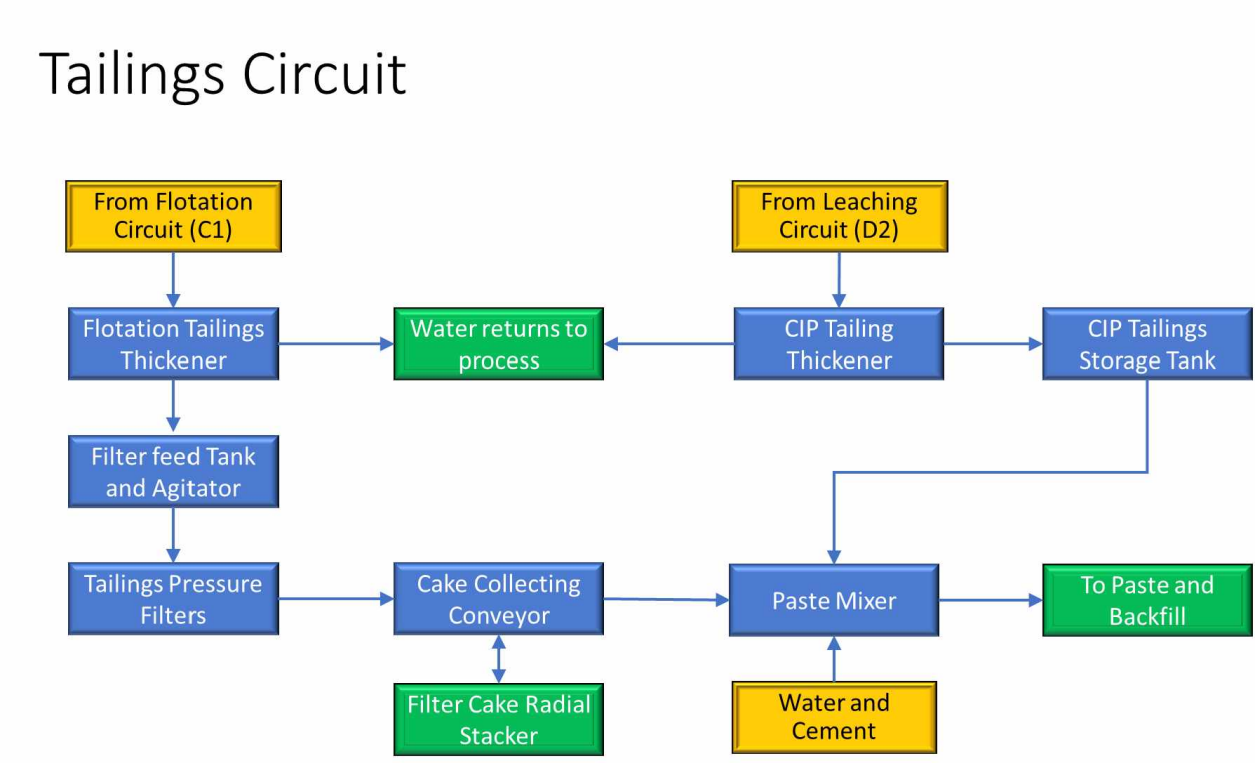

Figure 2-13 Tailings Circuit

The rejects from the flotation are first thickened, and the recovered water is returned to the process. The resulting pulp is the fed to tanks and agitators for conditioning and then pressured filtered.

The resulting cake is staked and used in the paste for the backfill in the mining process. The tailings from the Leaching Circuit are separately processed, and the recovered water is used only in the cyanide circuit (Konigsmann et al., 2017). The resulting pulp is used in the backfill paste.

2.3. Sensors Overview

A technical definition of a sensor, for this research, is taken from the Institute of Electrical and Electronic Engineers (IEEE). They define sensor as a device that responds to biological, chemical, or physical stimuli (such as heat, light, sound, pressure, magnetism, motion, and gas detection) and transmits the resulting signals or data by providing measurement, operating a control, or both (IEEE, 2010).

For example, in a piezoelectric sensor for pressure, the material's crystalline structure, usually quartz, is deformed when subject to strain. As the pressure increases, the crystal deformation generates an electrical potential within the crystalline lattice, which is measured and converted into an electronic signal that is transmitted into a monitor, this mechanism can be seen in Figure 2-14. How the sensor is built up and how the structure works can be seen on Figure 2-15. An example of real life sensor with this configuration is shown in

Figure 2-16. These sensors are located in internal combustion engines since they have no moving parts and are composed of high resistance materials (Gautschi, 2002; Rupitsch, 2017).

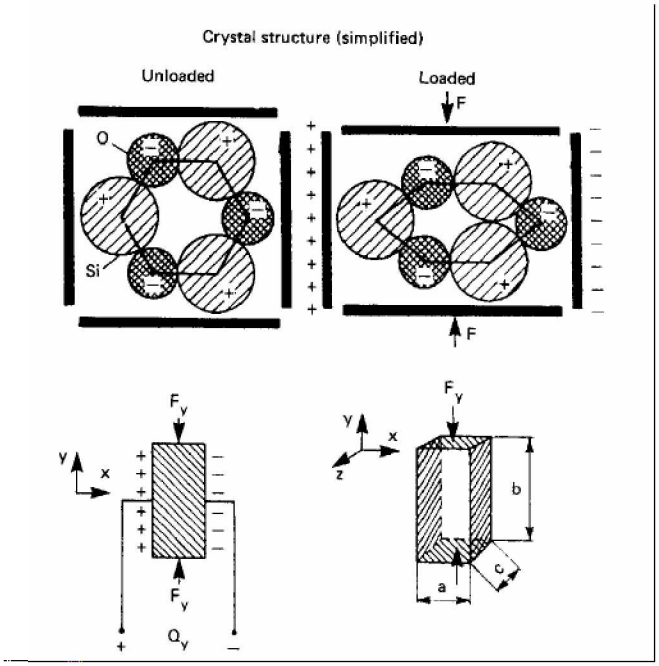

Figure 2-14 Principle of transverse piezoelectric effect (Boyes, 2003).

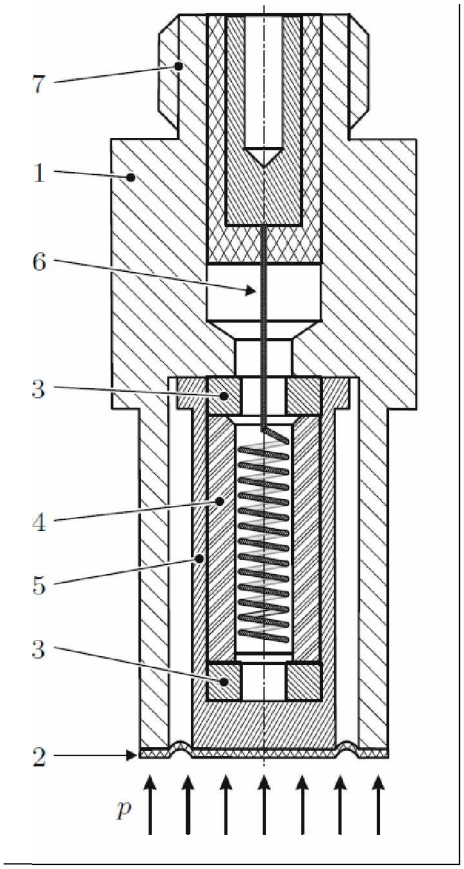

Figure 2-15 Internal structure of membrane-based piezoelectric pressure sensor; 1 cylindrical housing; 2 membrane; 3 transfer plate; 4 piezoelectric elements; 5 preloading sleeve; 6 helical-shaped spring; 7 electrical connectors (Rupitsch, 2017).

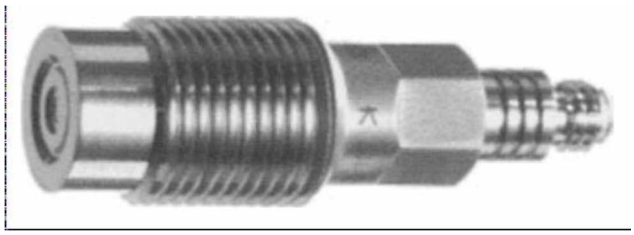

Figure 2-16 High Pressure System Range of 0 to 10 kbar (Gautschi, 2002).

Sensors can be simplified by the following diagram, Figure 2-17:

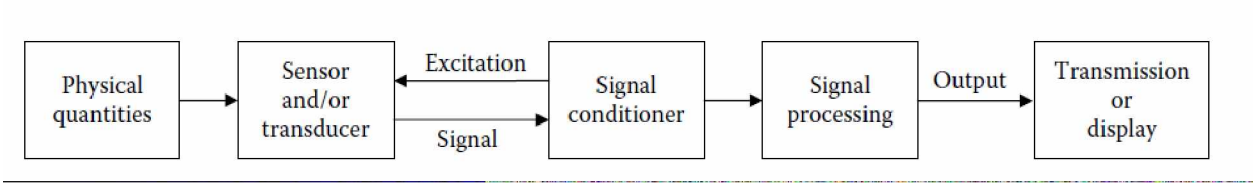

Figure 2-17 Construction structure of a typical instrument (Webster & Eren, 2014).

This diagram is valid for analog and digital sensors or a combination of both. In any of these, the structures is still the same. What is measured many sensors is not the changes in physical quantities, but their signals.

Measuring a sign depends on how the sensor is built. What is measured by the sensor is not the quantity itself, it measures a change in a parameter that is measurable and this parameter change has relation to the property that is wanted. A piezoelectric crystal generates a change in frequency of the current passing through it when the crystalline lattice is disturbed. A pressurized sensitive film, when presented with the change in its form will develop an equivalent change in its conductance. While both might be used to in a sensor that aims to measure pressure, they have a different relating measurement. While the film sensor measures the voltage difference that can be detected on the change, the piezoelectric sensor measures the change in frequency. While they measure the same physical quantity, pressure, the signal they produce will be different (Fraden, 2016).

Sensors can be analog, digital or a combination of both. Analog sensors transmit continuous signal stream, which is fed into a circuit, then amplified, filtered and transmitted to a display. Digital sensors work in a similar form, but the information is converted to digital form before it is passed on, resulting in an intermediary station that converts the information to digital. The importance of this relies on the fact that the information flow is no longer continuous, now being sampled, quantified and encoded. This means that its value is no longer continuous, but discreet. This is not a major concern since modern sensors do the sampling at a frequency much higher than the usual changes in industrial mining process (Webster & Eren, 2014).

2.3.1. Sensors in Mineral Processing

Mineral Processing has some features that require specific sensors. Given the rough nature of mineral processing, the sensor needs to be extra rugged, especially during the comminution phase.

Every operation in mineral processing requires information in order to be evaluated. Sensors are not only needed for this, but also have to be properly placed for a better understanding of the mineral processing.

2.3.2. Sensors Accuracy

Sensors are a measuring tool and like all measuring tools, they have inherent errors. The accuracy of a sensor is a key factor in its functionality. A sensor inaccuracy represents the highest deviation of a value from the ideal or true value of stimulus input (Fraden, 2016). This sort of inherent error emerges from a myriad of sources that are part of the engineering process of creating a sensor. An error such as these is part of the expected scenario.

The most desirable characteristic of a sensor is as much accuracy as possible (National Institute of Intrumentation Starndards, 2018). This comes with a tradeoff; highly accurate sensors work on a smaller range of measurements. They are more sensitive to mechanical disturbances, and usually, not as rugged as the less accurate counterparts.

The sensor reading is only meaningful if it can be understood. A sensor detects a disturbance in a given situation and translates that disturbance measurement into intelligible information by calibration. In metrology, calibration process function as a measurement procedure made with the tool in order to be calibrated and with a piece of reference equipment, the calibrator. The calibrator is kept in a setting as close as it can to the ideal reference value. The references found on the tool to be calibrated are then set with the ones found on the calibrator (Webster & Eren, 2014).

Although limited, the best way to exemplify this subject is presented as a system of equations (Webster & Eren, 2014):

$$\text{Ideal value Measured} = \text{Value} + \text{Bias} + \text{Error} \qquad \textbf{(Eq. 2.1)}$$

Ideal reference value = Reference Value + Bias + Error **(Eq. 2.2)**

Deficiency = Ideal measured Value − Ideal reference Value **(Eq. 2.3)**

This deficiency is the overall inaccuracy of a given sensor. It is caused in part by the calibrator and tool of inherent measurement bias.

It should also be noted that calibration is very costly as it can be disruptive to industrial processes. The sensor must be taken off-line, potentially stopping critical processes and incurring losses. Appropriate personnel must also be mobilized potentially disrupting other processes. To minimize these losses, it is best to schedule calibrations periodically.

The calibration process should be done periodically, regardless of the detection of errors. Small errors can build up and make the information less useful. When there are periodic calibrations, the main issue becomes detecting the errors that happen in between the calibrations.

2.3.3. Sensors Accuracy and Precision

Precision and accuracy do not mean the same. Accuracy is the uncertainty of a given sensor. Precision is the proximity of measured values when the measurement is repeated. This means that any given measure might be accurate but not precise and vice versa. Figure 2-18 brings that difference in a visual form.

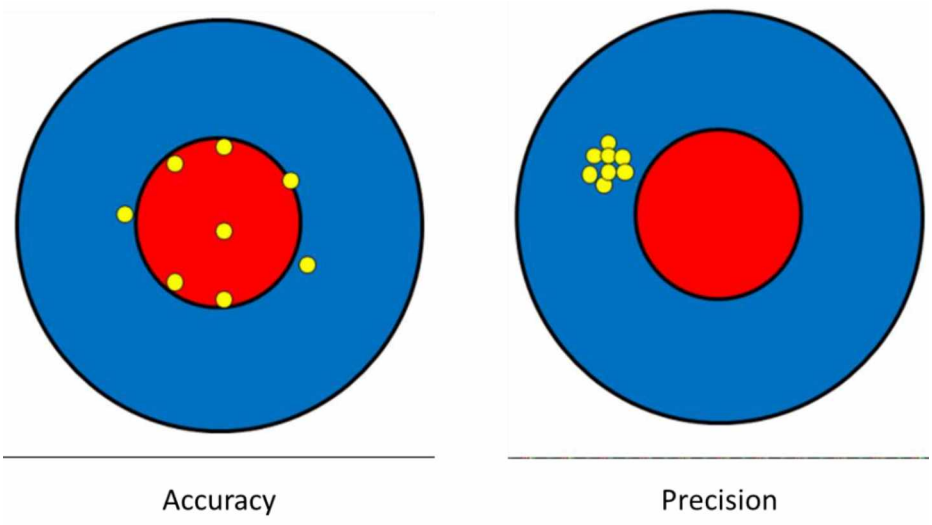

Figure 2-18 Accuracy and Precision (CHIEREGATI & PITARD, 2018)

In the above figure, the yellow dots on the left indicate an accurate measurement as the measurements (yellow dots) are centered around the true value (red circle). However, they are not very precise. The measurements on the right side are precise as the measurements are tightly clustered. However, being far from the red circle, they are not very accurate.

2.3.4. Sensor Faults

Having defined an inaccuracy, henceforth referred to as uncertainty, there is the need to delve on the issue of the faulty sensors. Faults differ from simple errors because they are higher than the tolerated deficiency of a sensor. The literature points to different behavioral fault classifications. Some researchers (Balaban, Saxena, Bansal, Goebel, & Curran, 2009) divide faults into five categories as shown below:

Bias – constant offset in the measured value, positive or negative, resulting in the following:

$$\text{Ideal value Measured} = \text{Value} + \text{Bias} + \text{Error} + \beta \quad \textbf{(Eq. 2.4)}$$

Where β is the constant offset value. This fault β classification is not time dependent. When β can be parametrized as a function of time, this results in another kind of failure. β can be caused by a faulty calibration, a change in sensor placement or any other physical change experienced by the sensor.

Drift – a time-dependent offset in the measured value. Literature accounts that there are only linear drifts examples being modeled. This, however, does not disprove the possibility of nonlinear drifts being observed. This results in the following:

$$\text{Ideal value Measured} = \text{Value} + \text{Bias} + \text{Error} + \delta(t) \quad \textbf{(Eq. 2.5)}$$

Where $\delta(t)$ represents the varying factor of offset.

Scaling (or gain failure) – the magnitude of the measured value is scaled by a factor of $\alpha(t)$. The sensor signal and the waveform does not change. This can be represented by the following:

$$\text{Ideal value Measured} = \text{Value}^* \alpha(t) + \text{Bias} + \text{Error} \quad \textbf{(Eq. 2.6)}$$

Noise – A series of time-dependent random observations. Usually, it is considered to have a mean of zero, being the one random variable that adds up to the measured value. It differs from the error because it is time dependent.

Hard Fault – the sensor output is stuck at a constant level, this might be due to loss of signal, where sensor data gives continuous zero value or a Stuck Sensor, where the sensor gives a continuous non-zero value.

Intermittent – deviations from normal readings that appear and disappear from the sensor signal. It can appear as any of the previously stated failures. Due to the intermittent and random behavior, they are especially challenging to track and identify.

Other researchers (Baljak, Tei, & Honiden, 2012) classified errors into two main groups, Continuous and Discontinuous. Continuous errors are either Malfunctions or Random errors. Malfunctions are what the previous classification considers Intermittent errors and Hard Faults. Continuous errors include Drifts and Bias, as classified under the first one.

2.4. Data Mining for Sensor Errors

Data-mining techniques tools are for relating data streams that can be associated, where correlations are identified between what's being used to analyze and understand the data streams (Brown, 2012). They can classify the internal behaviors of a given data stream, creating classes where attributes of each given set are studied. They can cluster, creating sets that groups together a particular part of the stream. They can recognize patterns, where reoccurring situations are identified and regular trends can be observed.

This listing is by no means exhaustive. Many applications will use a combination of one or more techniques to achieve the results needed. Prediction tools are for classification, pattern recognition and association, aiming to be able to predict, within the bounds of the previously gathered data, the future behavior of a given variable. So, any day-to-day issues will require this combination of tools in order to solve the problems (Pothina, 2017).

Data-mining works as a tool to understand large data sets. It is its own scientific field and tries to understand the behaviors in these situations. Regular statistical tools are a big part of the field, but they sometimes fall short on the need to process wider understanding. (Leek, 2015).

The use of regular statistical processes usually results in high production of false alarms (Kusiak & Song, 2009). The work done by Kusiak and Song used clustering algorithms to direct faults in power plant boilers. Other authors (Hou, Lian, Yao, & Yuan, 2006) and (Yang, Liu, Zhang, Nepal, & Chen, 2015) have used different data mine operations using different algorithms.

As all of this comes forward, the focus is to find gross errors on the sensor data. The algorithm proposed by (Pothina, 2017), differs from them because it focuses on finding small calibration issues on the data. His work focused on temperature sensors, which directly impacted energy costs and gold recovery. The importance of energy costs cannot be underestimated since 39% of all energy used in mining goes toward mineral processing (U.S Department of Energy, 2002).

2.5. Multiple Ratio Function Analysis with Automation (MRFAA) Algorithm

Pothina's research started as a Peak Reading Count and Sensitive Analysis (PRSCA) algorithm. In this algorithm, the number of peak readings (nP), above a given threshold (Th), of a given cycle is stored and compared to the readings of the next cycle. As the two vessels work in alternate cycles, if one of these vessels is out of calibration, the number of peaks will increase in magnitude and can be flagged.

The tests ran by him demonstrates that once bias is introduced, this results in a relevant change on the number of peak readings. The main drawback of this algorithm is its inability to function without a parallel set of "clean" data set, i.e., true measurements. This is not a realistic demand.

In order to overcome this issue, the algorithm developed by Pothina (2017) uses data from other independent sensors in the system. This algorithm is focused on solving small-scale deviation on sensor data, and how to detect them.

Data from the stripping circuit was gathered on a 10 min average interval. It included temperature data from both strip vessels, the flow rate of the barren flow sensor, the temperature data from the four heat exchangers and the flow data from the glycol circuit. All this data information was gathered through the period of 12 months, from January 1st, 2015 to December 31, 2015.

The data was filtered by a threshold value so that it reduced the entire data collected to only those that represented a stripping cycle. A cycle is assumed to start when the temperature in a stripping vessel exceeds the threshold value. The cycle is assumed to stop when the temperature drops below this threshold. The threshold temperature has to be sufficiently high so that spurious "cycles" are not created by thresholding.

A descriptive statistic from the data used by Pothina can be seen in Table 2.2. Pothina's assumption, regarding calibration errors, is that this sort of deviation comes in the form of an offset value, that constantly deviates from the true values. Since this algorithm only focuses on this sort of error, other errors assortments, that have more complex behaviors are not targeted by the algorithm. Those other errors can have larger magnitudes and, so, for the detection of its issues, common statistical analysis is more suited and effective.

Table 2.2: Descriptive statistics for the sensor data collected (Pothina, 2017)

Sensor	Measuring Parameter/ Variable	Unit	Maximum	Minimum	Mean	Median	Standard Deviation	Data Cleansing Cut-Off Value ($Th_{cleanse}$)
S1	Strip Vessel - 1 Temperature	°F	32.7	52.1	149	104.1	78.2	150
S2	Strip Vessel - 2 Temperature	°F	304.3	57.7	418	104.2	77.6	150
H1	Heat Exchanger - 1 Outlet Temperature	°F	334	42.9	126	77.1	89.8	150
H2	Heat Exchanger - 2 Outlet Temperature	°F	336.2	56.3	266	297.9	60.2	150
H3	Heat Exchanger - 3 Outlet Temperature	°F	287.6	48.4	191	223.5	71	150
H4	Heat Exchanger - 4 Outlet Temperature	°F	299.5	48.5	145	99	79.2	150
BARNFL	Barren Solution Flow	GPM	84	0	39.6	35	17.3	20
GLYFL	Glycol Flow	GPM	230	0	210	214.6	18.4	150

Pothina's research defines bias as: "Bias is expressed as a percentage over the true reading value" (Pothina, 2017). In order to validate the results an artificial ±2% bias was induced on the database. Pogo Mine requested a value of ±2%and, therefore, identifying such deviations as soon as possible was the main concern. The analysis focuses on data from the Strip Vessels (S1 and S2), though the error was inserted only in S1. It was inserted systematically in varied locations of the database and tested multiple times in order to guarantee that the results were consistent.

The pressurized Zadra Process used in Pogo has an optimal operation temperature between 270°F and 280°F. Since the circuit uses a batch system, the temperature will increase towards this range as the system is heated. Later, at the end of the stripping process, the temperature will again drop. Therefore, the cycle can be characterized by the rise and eventual fall of temperature seen in this data. As the algorithm searches for displacement in these values, the data can be reduced to the only data around the peak temperature. This results in an analysis that only shows itself at the

peak of the system. The peak threshold for the data gathering was 250°F; Figure 2-19shows the appearance of this data for S1.

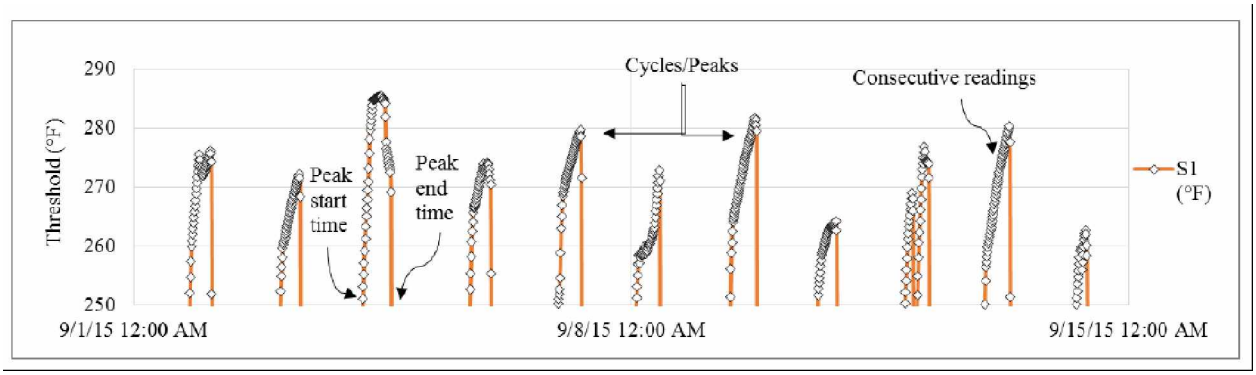

Figure 2-19 Peaks vs Thresholds in cleanser of S1 Sensor Data (Pothina, 2017)

In order to develop an algorithm that can be used in an industrial setting, Pothina proposed that there was a need for going beyond the usual peak on reading algorithms. Considering the heating system presented in Pogo, and the direct relationship between the heat provided by heat exchangers to the stripping vessels, there should be some relationship between the data from the heat exchangers, the strip vessels and the material circulating in between (Barren and Glycol flow). He proposed using these relationships for the algorithm. Figure 2-20 presents a schematic for these relations.

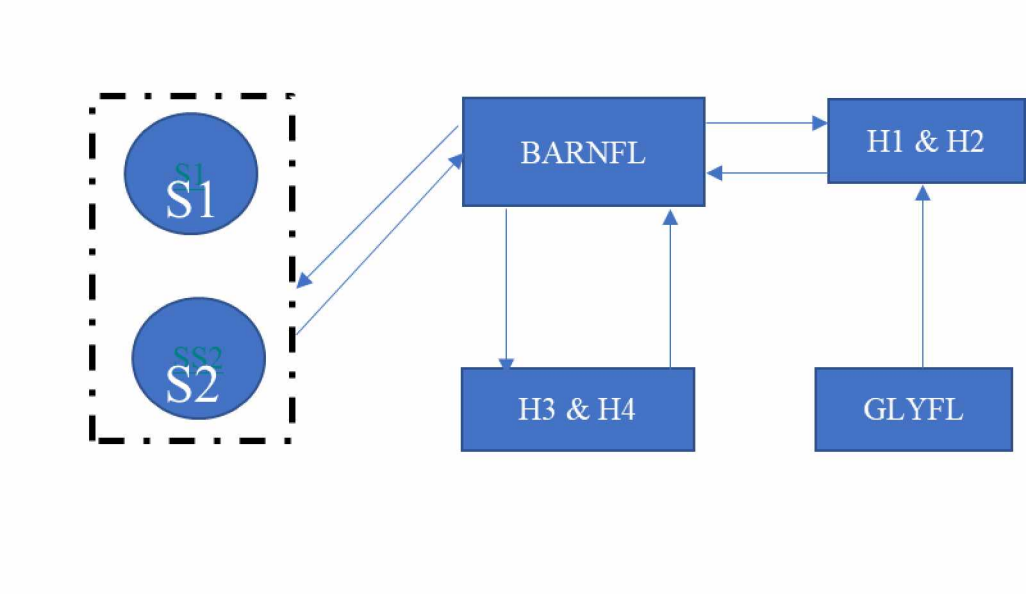

Figure 2-20 Various Sensors and their interrelations

As was stated before, the strip vessels work on alternate cycles. The heat provided to each, when being used, is provided by the heat exchangers. The total heat provided affects the measured temperature on the vessel. The glycol flow also is a measure of the provided heat as it connects the boilers to the heat exchangers. Barren flow is also a factor as it is the medium that conducts the heat from the exchangers to the vessels. For naming purposes, when in reference to a vessel active in a cycle, i.e., the vessel being heated, the name Active Vessel (S_x; x=1 or 2) will be used.

Several techniques, such as neural networks and Fourier Transforms were applied to detect the applied error. However, the ±2% error was too little to be detected by these techniques. The approach that proved to be successful relates both temperatures in the vessel and the temperature of the heat exchangers. The successful relationship could be represented as it follows:

$$\text{Ratio} = f[T_{\text{peak-ave}}(\text{Heat}),\ T_{\text{peak-ave}}(S_x)] \qquad \textbf{(Eq. 2.7)}$$

where x=1 or 2, for strip vessels 1 and 2 respectively. The equation describes a S_x:H ratio, i.e. ratio of temperature-based statistic for a stripping vessel with the same temperature based statistic for the 4 heat exchangers. Thus, there were two ratios, S1:H, and S2:H.

Ideally, the S1:H ratio would be compared to S2:H ratio for the same cycle. Since the vessels work in alternate cycles, that is impossible. Therefore, the S1:H ratio was compared to S2:H ratio for the S2 cycle that immediately followed the S1 cycle and was within 24 hours of the S1 cycle. This was called matching "forward".

To compute the Sx:H ratio, temperature data from the heat exchangers are gathered at the same time as the data from the active vessel. The mean temperature of the active strip vessel during a cycle is used for the ratio, along with the mean temperature of the four heat exchangers during the same cycle. A threshold is used for every heat exchanger, and if any data is below this threshold, the data from this individual heat exchanger is expunged and the mean is calculated without it. If none of the data on the heat exchangers are valid, data from the vessel matching this time period is also discarded.

So, in order to gather a representative data, the minimum amount of data points needed are 6, as this represents a peak cycle of at least 1 hour (each reading is 10 minutes apart). Cycles shorter than 1 hour are assumed to be unimportant or spurious,

After collecting the data, the average ratio of a given cycle is calculated and compared to its subsequent cycle in the other vessel. If the S1:H ratio is larger than S2:H ratio, then a value of +1 is added to the cross-ratio comparison. Otherwise, a value of -1 is added. Figure 2-21 shows the flow chart for the algorithm.

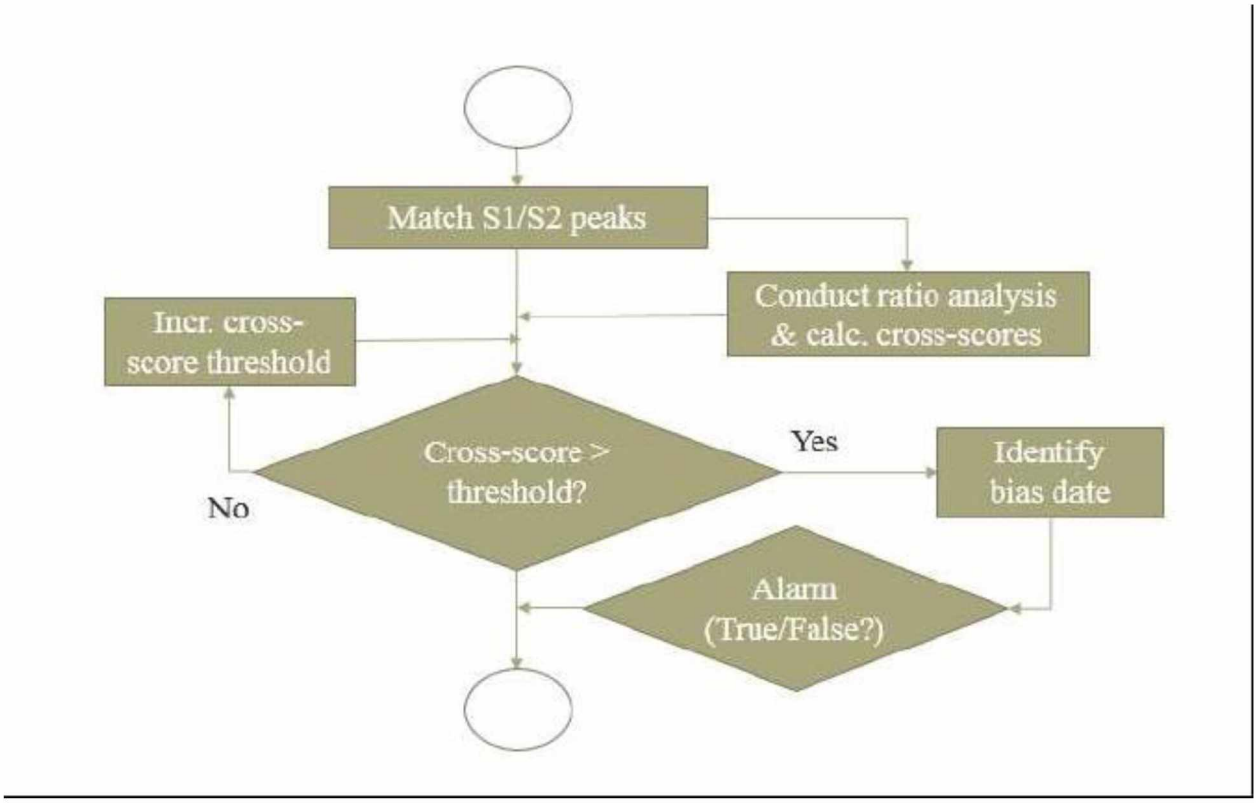

Figure 2-21 Flow chart for the multiple ratio function analysis (MRFAA) algorithm. (Pothina, 2017)

In a non-biased situation, S1:H ratio is not continuously higher or lower than the S2:H ratio. Therefore, the cross-scores, which are a cumulative sum of a series of 1s and -1s, remain small. Similar to a Wilcoxon two-sample rank sum test, if the data are similar, the value of the cross-ratio comparison should be close to 0. Considering the data comes from two different sources, they are similar enough to be considered equivalent, the pairing should trend towards the value 0. The cross-scores stray significantly from zero when one strip vessel runs consistently hotter or colder compared to the other one. This is usually a sign of a sensor error and, therefore, used to identify errors.

Chapter 3. Research Methodology

Pothina's research focused on a strategy of using data mining to establish a relationship between the already-available systems to detect deviations on the temperature reading of the vessels. The research started as a peak-reading algorithm; therefore, only the data related to the values around the peaks of the cycles were relevant.

As described in the previous chapter, the strip circuit works in batches. While Vessel 1 is stripping gold, Vessel 2 is being loaded with new material. As soon as Vessel 1 is done, the stripping operation moves to Vessel 2. During Vessel 2's strip operation, Vessel 1 is unloaded and loaded with new material.

The original data gathered was acquired from January 1st, 2015, to December 31st, 2015. This procedure is done in ten-minute intervals. It consisted of data from the temperature sensors in Vessel 1 (S1);

- the temperature sensor in Vessel 2 (S2);
- the flow measurement from the barren tank (BARNFL);
- the flow measurement of glycol from the boiler to Heat Exchangers 1 and 2 (GLYFL);
- the temperature sensor in Boiler Outlet 1 (HX1);
- the temperature sensor in Boiler Outlet 2 (HX2);
- the temperature sensor from Heat Exchanger 3 (HX3); and
- the temperature sensor from heat exchanger 4 (HX4).

A key element of the algorithm is comparing the behavior of strip vessels with one another at the same point in time. However, since the vessels operate in alternate cycles, a direct comparison is impossible. Therefore, it was decided that the vessel cycles could only be compared if they occurred within 24 hours of each other. If the time difference was more, operational

conditions may very well be quite different, as things can change quite a bit within a day in a mineral processing plant.

3.1. Key Concepts

A few concepts are used throughout the dissertation. In order to clarify meanings, this section will define each one.

3.1.1. Cycle

For the purposes of this dissertation, a cycle is a period between the start of the stripping process in a given vessel and the end of this process. This is delimited in the data by the fluctuation of the vessel temperature. As the cycle starts, the vessel temperature rises to a maximum point; after that, it falls down to the point that allows the vessel to be unloaded.

Figure 3-1 shows a cycle.

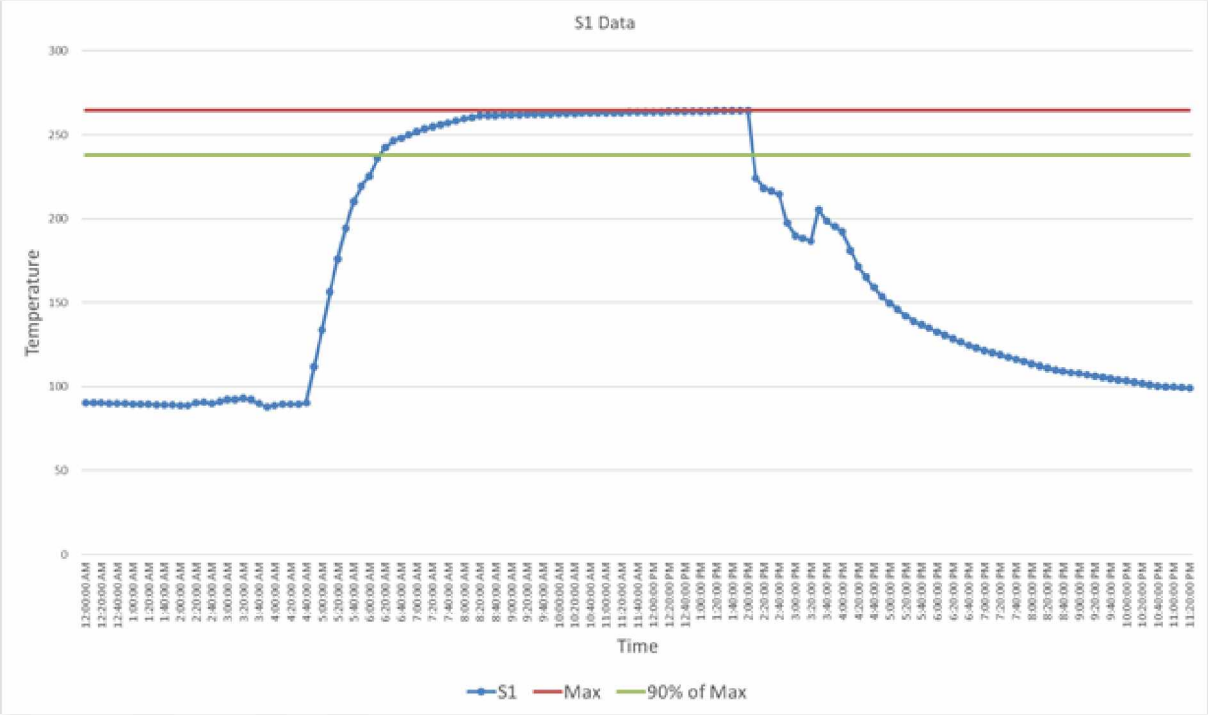

Figure 3-1 Vessel 1 Cycle Example

For this project, the values of a cycle are the values between the maximum temperature and 90%, i.e. the top 10% achieved around the peaks. In Figure 3-1 this represents the data points above the green line. This is the data that will be gathered in the algorithm to compare the cycles.

3.1.2. HX Average

Previously, the heat exchangers' data was used to create a ratio of heat provided to the system. This was done by gathering the mean value of the valid data from the heat exchangers. Valid data means values that are above a certain threshold. That way, if one of the values in a heat exchanger is below the threshold, it will not be used to calculate the mean heat-exchanger point.

Figure 3-2 shows how this calculation is done: the discontinuous red line represents the calculation of the mean, considering only the values from the heat exchanger that are above the threshold. The line is discontinuous because at some points there is no valid data above the threshold.

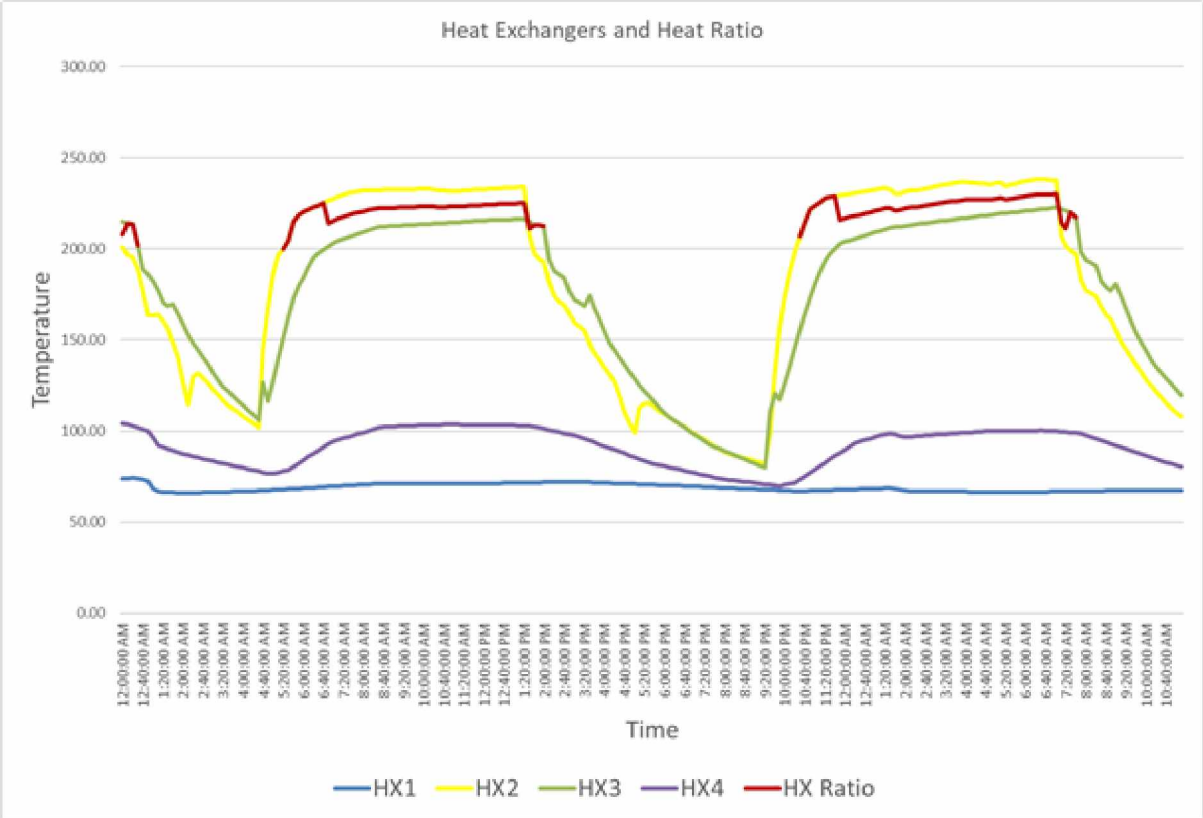

Figure 3-2 Heat Exchangers and HX Average

3.1.3. Matching Forward

A cycle of Vessel 1 is compared to the cycle that occurs immediately afterwards in Vessel 2. This is called matching "forward" since the match occurs with a cycle that is "forward" in time.
Cross Score Threshold

When comparing cycles, a value is attributed to every comparison. If the averages of a Vessel 1 cycle are higher than those of a Vessel 2 cycle, a value of 1 is attributed to that comparison. If not, the value attributed to the comparison is -1. At any given time, a running total is maintained for the last several comparisons (moving window). The running total is computed as soon as there are enough matching cycles to form a window of comparison.

"Cross score threshold" is the point where this sum has risen above the regular, established pattern. This is the value for which there is enough indication that there is a bias regarding the data.

3.1.4. Window of Comparison

When gathering the total score for the comparisons, it will only consider some past comparisons. This is done in order to limit the effect of previous data, as going too far back would include data that is no longer relevant. In Pothina's research, this was a value of 10, so it considered only the last ten valid comparisons.

3.2. Algorithm

The algorithm works using the data provided by Pogo Mine. This was provided in an Excel worksheet. The algorithm imports the data as a csv file. This creates a table with all the data for the period provided.

3.2.1. Identifying the Cycles

First, the algorithm calculates the value for the HX Average ratio — this calculation is described in key concepts; the value is calculated for every line of data. This is done for the whole data base before the start of the cycle identification. This process is described in Figure 3-3.

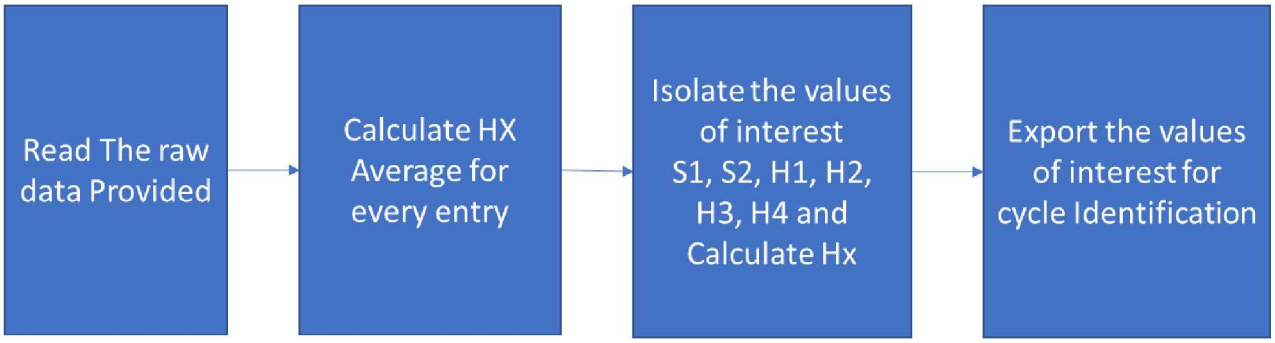

Figure 3-3 Flow chart for calculation and analysis of Raw Data

Following this, it reads the imported table line by line. It scans the value for Vessel 1's temperature (S1), and if it is above 200 °F, it starts to save the values from S1 and the H average in a new table. It keeps going line by line until the value falls under 200 °F. As soon as this happens, it calculates the maximum value achieved and uses only the data above 90% of this maximum value. The "cycle" is the time span between when the temperature first exceeds, and then drops below, the time 90% of the highest value. The temperature values of S1 during the cycle are then averaged and divided by the temperature values of the heat exchangers (during the cycle). This value is called the "Heat Ratio".

The mean value for the Heat Ratio is calculated for the cycle, and this is stored with the cycle number and the date the cycle started. If the cycle has under five valid data points, the cycle is discarded.

This procedure is repeated for Vessel 2, resulting in two data frames that gather all the mean Heat Ratios of each cycle and the date they started.

This process is summarized in Figure 3-4 where the cycle identification flowchart is presented.

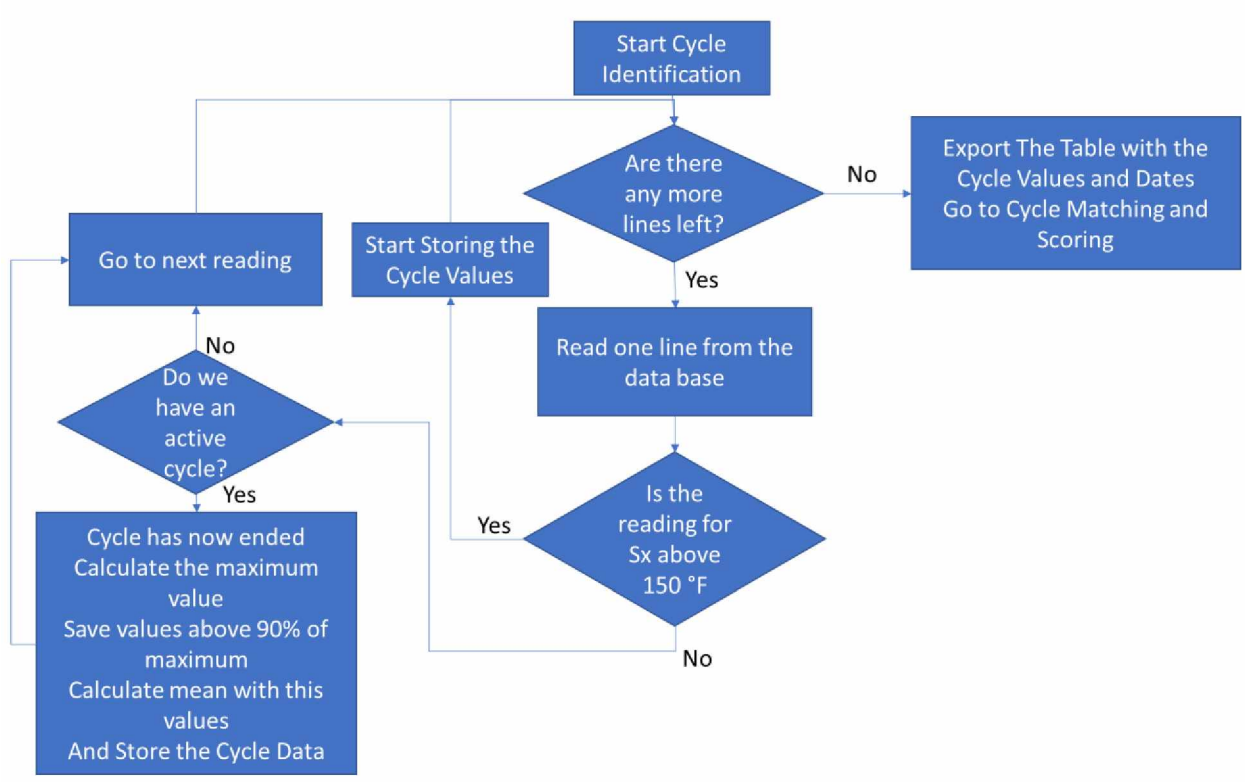

Figure 3-4 Flow chart for Cycle identification

Figure 3-4 exemplifies this process; this is done on the whole data set. The original algorithm did not export this data for further analysis. This change allows the storage of the analyzed cycles for further study, allowing the study of any pattern present in the cycles.

3.2.2. Comparing cycles

When all cycles have been gathered, the algorithm starts comparing them. Since it only does forward-matching, it considers the first valid cycle of S1, then matches it with S2 if they are 24 hours apart. If this criterion is not met, it goes on to the next cycle of S1.

If the cycles are less than 24 hours apart from each other, the algorithm compares the value of the Ratio. If the value of S1 > S2, then a score of 1 is stored. If S2 > S1, then a score of -1 is stored.

As soon as the algorithm has enough comparisons to form a Window of Comparison, it sums up the scores and evaluates the total value. If this score is above the threshold, it saves the date of occurrence and moves forward.

This process is repeated until there are no more cycles to compare.

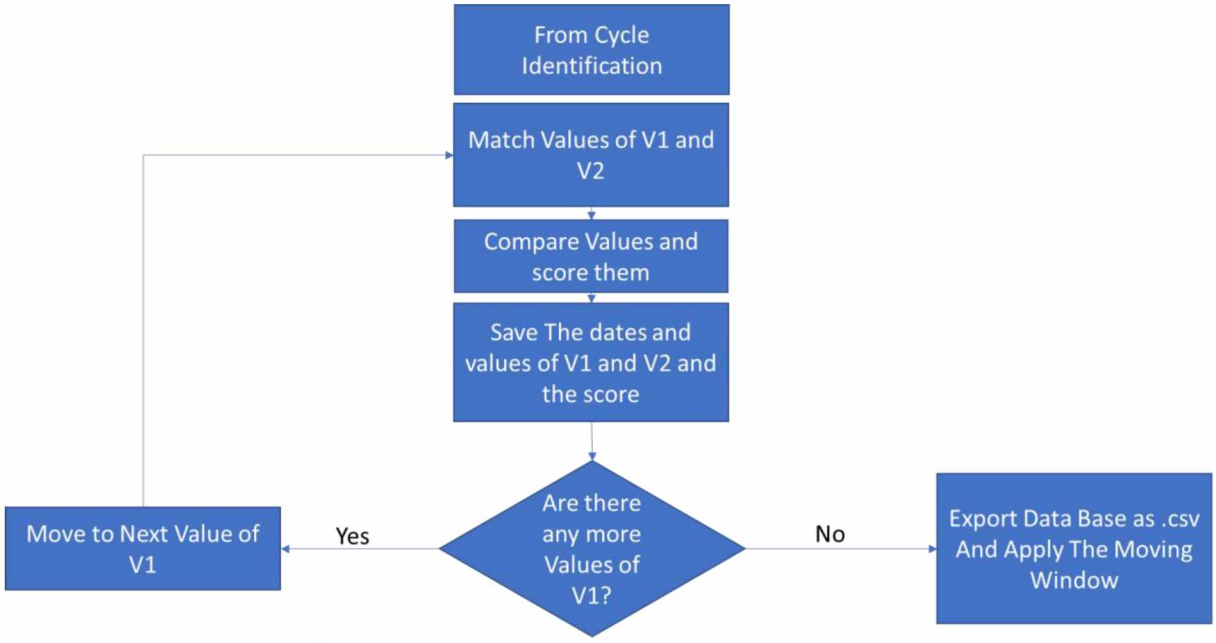

Figure 3-5 Flowchart of cycles comparison

Figure 3-5 Shows the flow chart for this part of the algorithm. There are a couple of deviations from the original algorithm; these were done to allow a better understanding of the overall behavior of the data.

The original algorithm only focused on the values that had risen above the threshold on the positive side. This means that it only focused on S1 having consistent values above S2. Since there is no thresholding prior to comparison in the current method, it allows a more complete comparison of S1 and S2. Thus, new situations where S2 is consistently above S1 may be included if they were below threshold in Pothina's algorithm. It also allows for a histogram analysis of the data behavior.

Also by saving the database, the amount of cycles in the moving window can be tailored to test the algorithm with smaller or larger windows without having to repeat the whole process again.

The algorithm still focuses on the dates when there was continuous abnormal behavior in the sensors. The reasoning behind the importance of these results comes from the following: since the material from both vessels is roughly similar, they should behave, within a small comparison window, in a similar way. So, any continuous abnormal behavior will be the fault of the sensors, not of the material.

3.3. Assumptions for the Tests

The algorithm gathers a derived value from each cycle and compares it to the corresponding value in the next cycle in another vessel.

Since the data consists of the mean value in a 10-minute span, there is no way to ascertain any assumptions about the underlying distribution of the data. However, by using the 24-hour window, pairs can be matched, and so the values from S1 and S2 cycles can be matched and compared. Since the material processed within this time frame is roughly the same, and the process is the same, these values should not deviate from each other unless the following happens: a) the material changes, b) the process changes, and c) the measuring is in error.

Issue A is resolved using the forward matching within 24 hours, i.e. material changes are not likely to be dramatically different within 24 hrs so as to require a different process setting. Issue B cannot be analyzed with the cycle data, but will be known to the operator; therefore, if a deviation is observed, it should come from C.

The proposed test is very similar to the non-parametric test called the sign test. The values are paired and matched, and a score is calculated (either +1 or -1). The underlying assumption is that if the paired sets are equal, any random pair of measurements are equally likely to be larger than the other.

Considering the last ten valid matches to be the window of comparison, if the underlying distribution is indeed binomial, the cumulative scores in an unbiased data set should follow the results found in Table 3.1:

Table 3.1: Binomial distribution for 10 matches

Binomial distribution for 10 matches			
S1 Successes	**S2 Successes**	**Cumulative Score (last 10)**	**Chance**
0	10	-10	0.10%
1	9	-8	0.98%
2	8	-6	4.40%
3	7	-4	11.72%
4	6	-2	20.51%
5	5	0	24.61%
6	4	2	20.51%
7	3	4	11.72%
8	2	6	4.40%
9	1	8	0.98%
10	0	10	0.10%

This means that, for example, cumulative scores of +6 should be found in 4.4% of the cases and results over 8 points should be only found in less than 1% of all the matches observed.

The original research used a heuristic analysis to achieve the cross-threshold value. It assumed that the process data from 2015 and 2016 was inherently unbiased, i.e. that there were no biases in how the strip vessels were operated. This assumption was not tested.

In order to test this assumption, the baseline database was scrutinized with respect to cumulative scores of the last ten cycles.

3.3.1. Database analysis to test for bias in strip vessel operation

The original data histogram for the year 2015, without any errors added, had 267 cumulative scores (Figure 3-6). Of these, there were 48 instances (18% of total) when the cumulative score was exactly zero. For a binomially distributed process, 24.6% of the comparisons would yield a zero cumulative score. A slight bias (cumulative score of +2) existed 41 times towards S1 (15.3%). A similar bias existed, however, 63 times (23.5%) towards S2 (cumulative score of -2). Neither of these match binomial distribution for cumulative score of ±2 (20.5%).

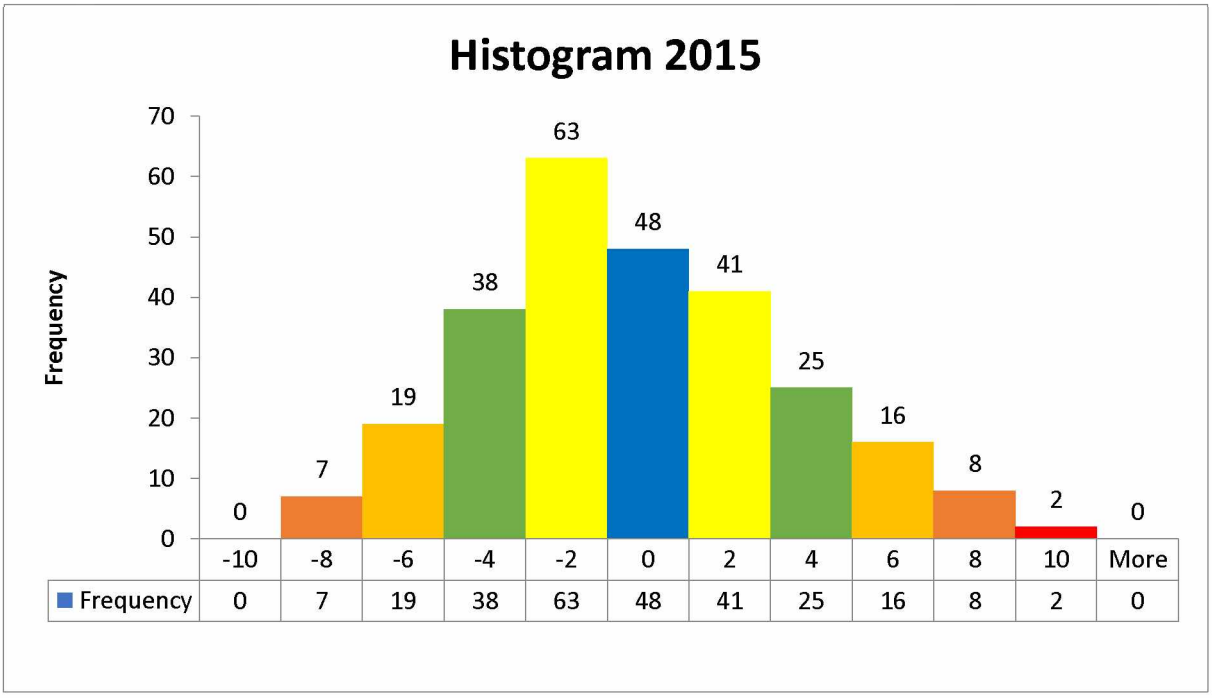

Figure 3-6 Histogram for 2015

Comparing the rest of the data to binomial distribution, the achieved cumulative scores are more skewed than expected. It has more values on the higher ends of the distribution, and it shows a preference towards S2's having a more consistent higher value.

The threshold value for warning in the initial algorithm was established as +8 (using S1 as the basis), The equivalent for S2 would be -8. What can be seen is that the dataset has almost the same number of -8 and +8 values. Also, there is the fact that there are only two occurrences of the value 10. Its equivalent, -10, is not present.

Using the same procedure for 2016, these results have a small difference in the overall distribution, as can be seen in the Figure 3-7:

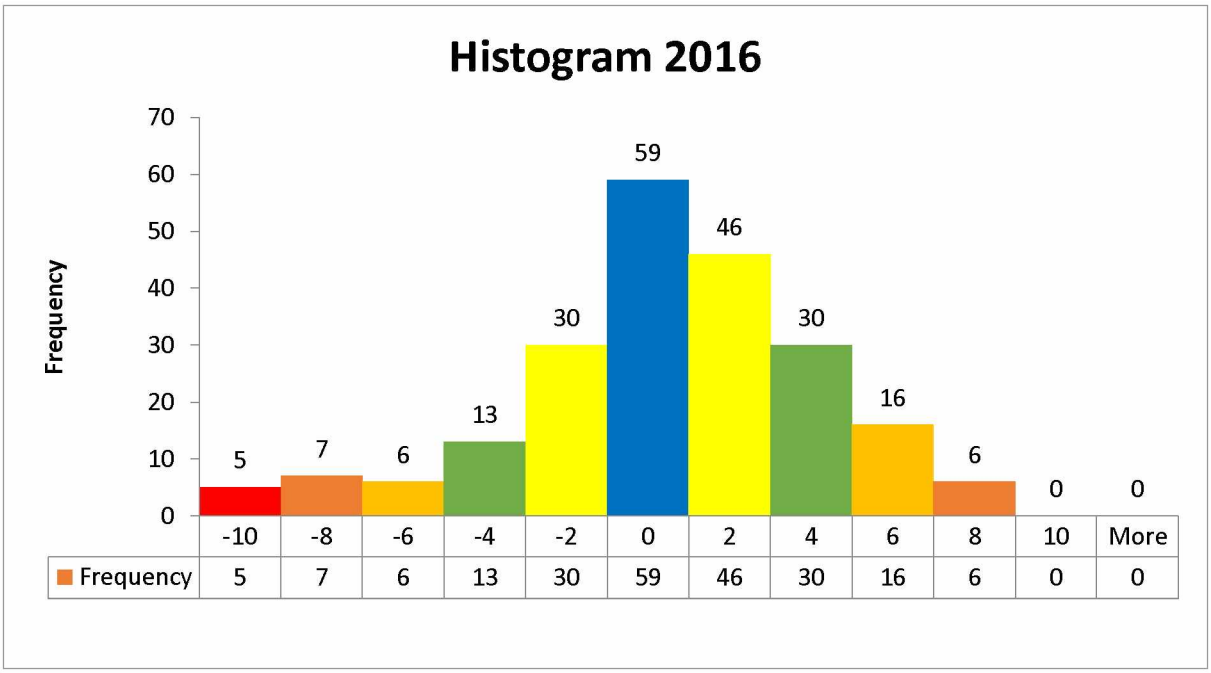

Figure 3-7 Histogram for 2016

Note that in 2016 the overall results do not favor S2 as much as they did in 2015, but the extreme values are much more prevalent this year for that Vessel. While the values of +8 and its equivalent, -8, are similar, the values of -10 have a significant presence in five circumstances, and there are no values for+10. Considering its overall behavior, the 2016 data set is more centered around 0. It is, however, not entirely symmetrical considering the center values of -2 to +2, but it is more so than that of 2015.

Before presenting the 2017 results for this procedure, there are a few warnings that should be noted. The values for H1 and H2 during 2017 were not present: the sensor data presented itself as either stuck on value, with a "No Data" tag, or with a warning saying that the sensor is offline. This issue resulted in a data set that has fewer data to work with, and it also resulted in a scenario where the data from the primary heat source, the boiler, is missing. The only data present comes from the recycling heat and the vessel temperature. The resulting histogram is seen in Figure 3-8.

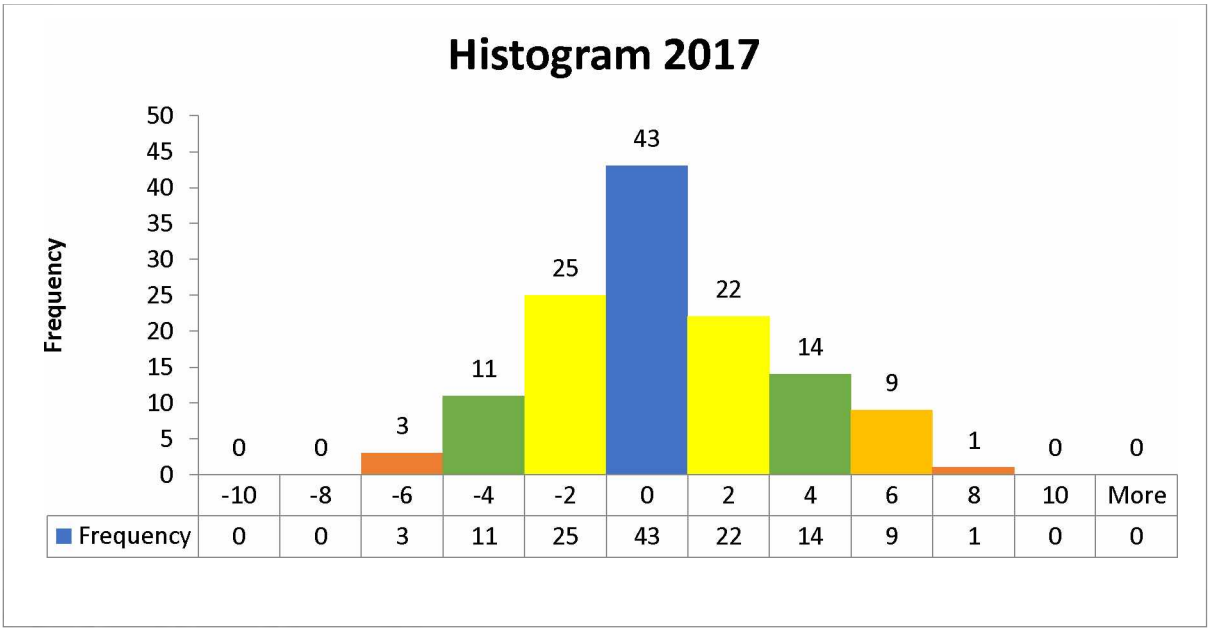

Figure 3-8 Histogram for 2017

The data from 2017 is much more centralized and symmetrical with only a few outliers; only one extreme value was detected during the year. The distribution is centered around the 0 value. This situation resembles the binomial distribution much closely, the values in a bracket being much closer to the equivalent value in the other bracket.

As mentioned above, 2017 had only one extreme value, favoring S1, but from all data sets, this is the one with a more even distribution.

Using this procedure on 2018 data shows a year where the distribution is skewed towards S1. 2018 data did not have the same issues observed in 2017: the values for H1 and H2 were present and valid during the year. The resulting histogram is presented in Figure 3-9.

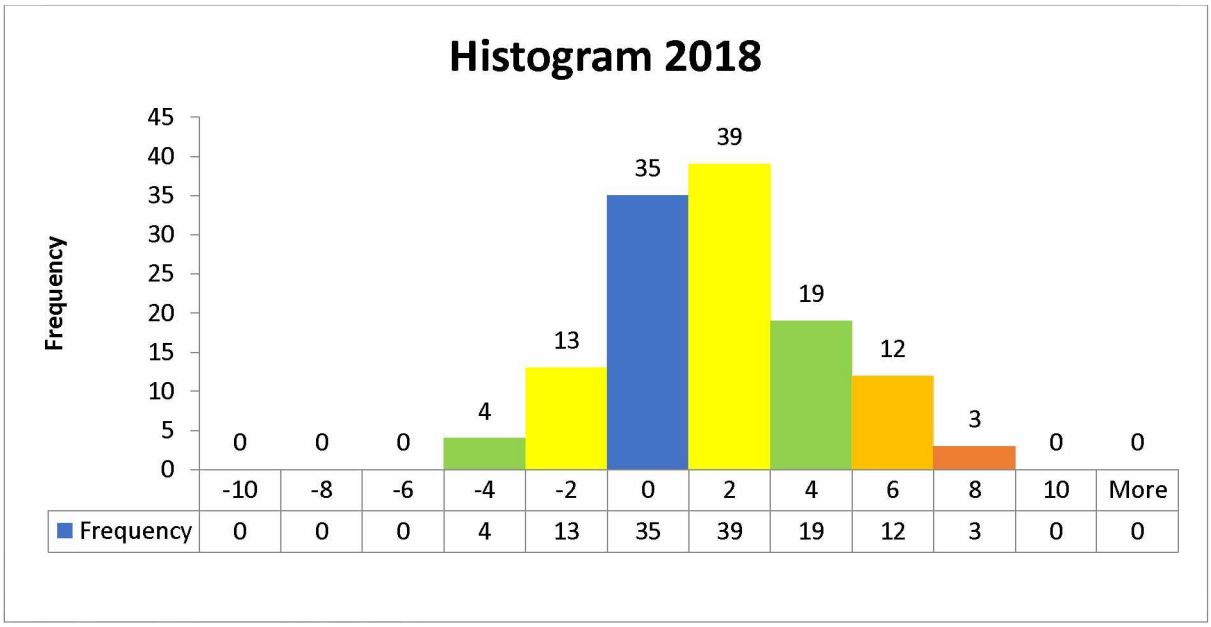

Figure 3-9 Histogram for 2018

The year 2018 presented three extreme values in favor of S1, and the overall values do favor S1, as most of the summations resulted in a +2 result. This shows a slight bias toward S1, as it consistently runs better results.

The histogram analysis reveals the overall behavior of the data. Even though the data does not follow the binomial distribution as was expected, there is an overall trend of symmetry, especially around |8|, i.e. there is not a big difference between +8 and -8 in most years. Also, there are not that many occurrences beyond |8|. This indicates that setting a threshold of |8| may be appropriate.

Considering these results, the cumulative scores (sum of last 10) were inspected more closely. Using the year 2015 as an example, the following patterns can be seen emerging:

Plotting the results for the Heat Ratio value for S1 cycles in blue and the Heat Ratio value for S2 in cycles in orange, the operating pattern emerges. By graphing the sum of the last ten cycles in green bars, the resulting interaction is explicated. These graphs are presented in Figure 3-10 and Figure 3-11.

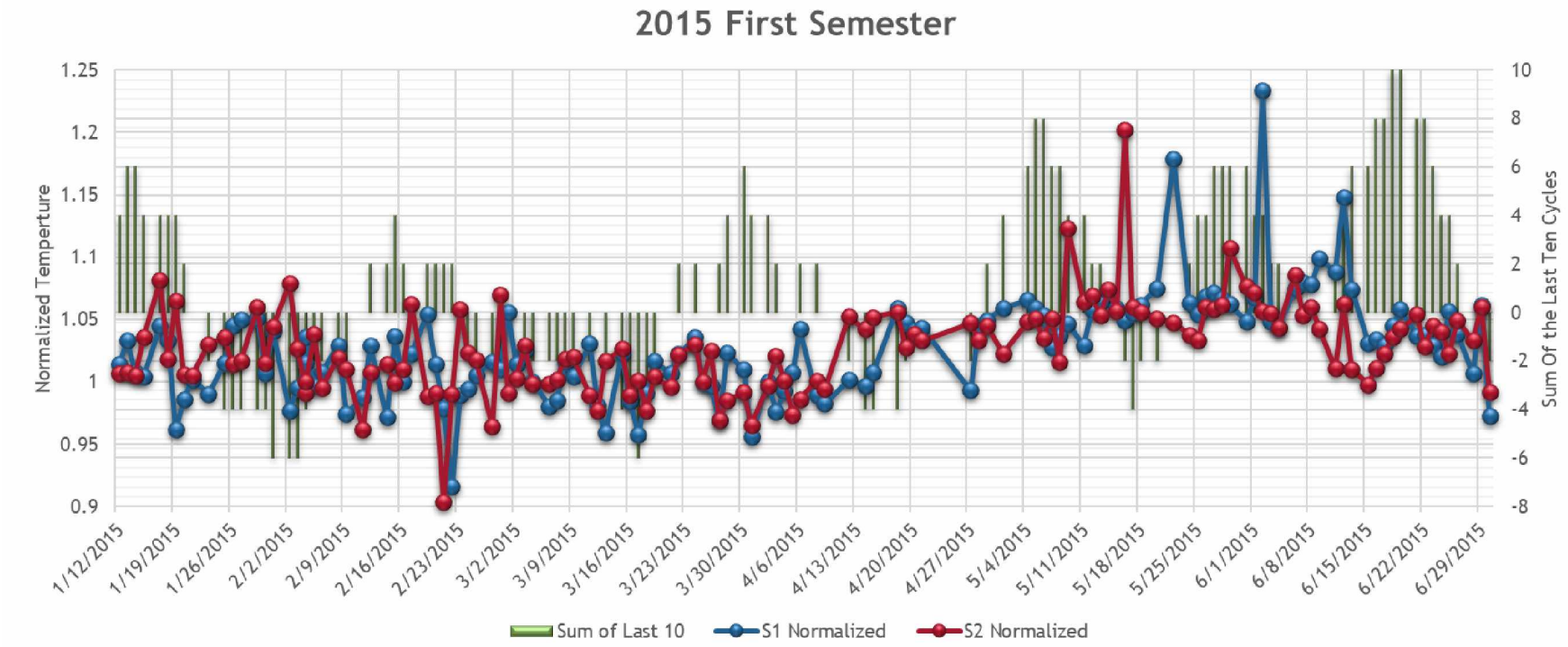

Figure 3-10 2015 First Semester

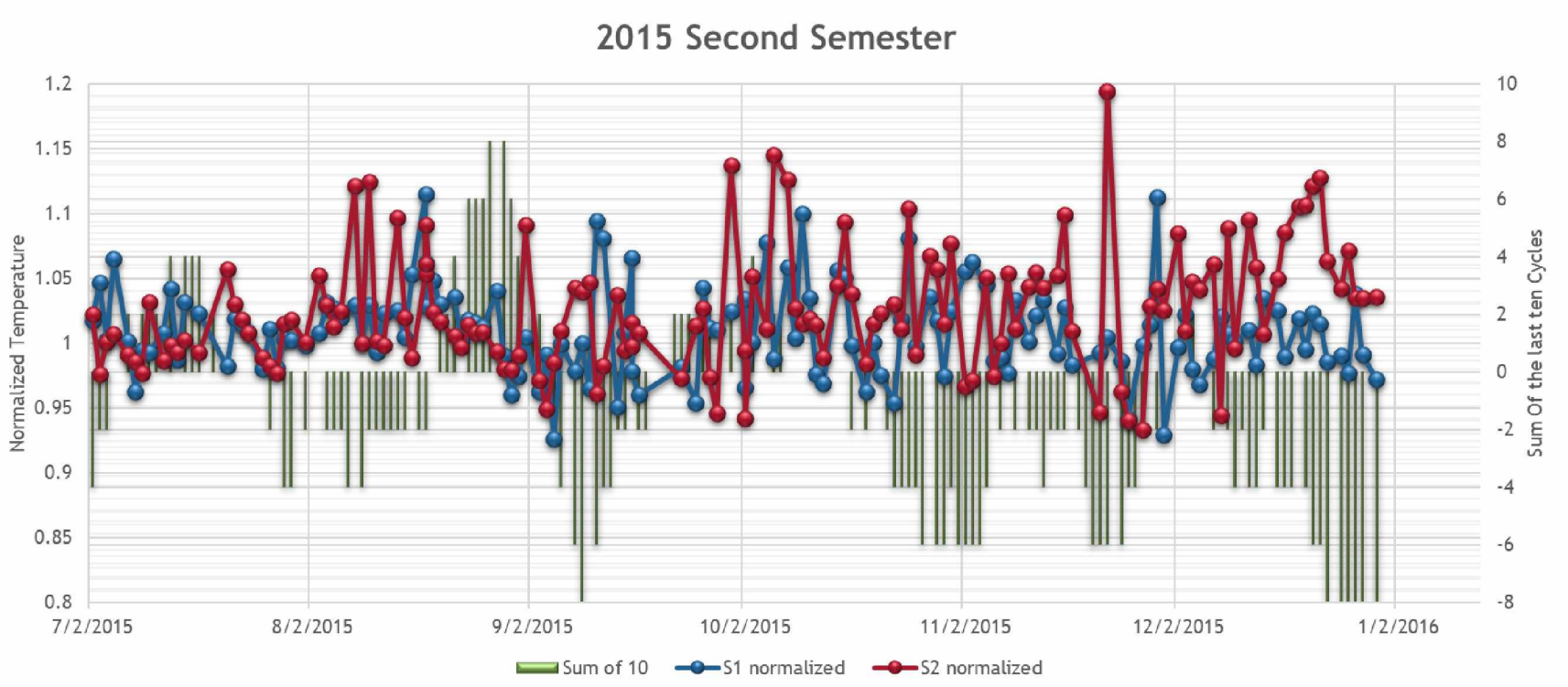

Figure 3-11 2015 Second Semester

The operating parameters of S1 and S2 are independent as there is no operational connection between them. What can be observed is that the sum of the ten last comparisons has momentum, as it approaches one of the extreme values. The cumulative score for the last 10 comparisons does not change dramatically as it can only change by +1 or -1 from the last value. A cumulative score reaches a threshold faster if there a winning streak ("momentum") for one of the strip vessels. The cumulative score does not move from positive to negative unless the opposite streak happens. This momentum is the key operational factor of the algorithm, because a bias will help it build up fast, allowing for a fast detection.

Note that in the first semester of 2015, S1 consistently beats the values of S2. Most of the high values occur in the first semester, while the second semester has all occurrences of low values. If there were an error in S1 during the second semester, the persistence of high values of S2 would hide this, as the comparisons would not build up to high scores unless the error is substantial. The algorithm would be defeated in such a situation, unless the error was substantial.

Analyzing 2016, there is a similar pattern of concentration of extreme values in a given semester. It is as extreme as in 2015. All extreme low values are present in the first semester. All extreme high values are present in the second semester. Up to February, there are consistent high values for S2, a trend that continues from the year before. As presented in Figure 3-12.

The second semester presents a gentler pattern; although all the extremely high values are present in this semester, they are of a small quantity and only trend at the very end of the year. As can be seen in Figure 3-13.

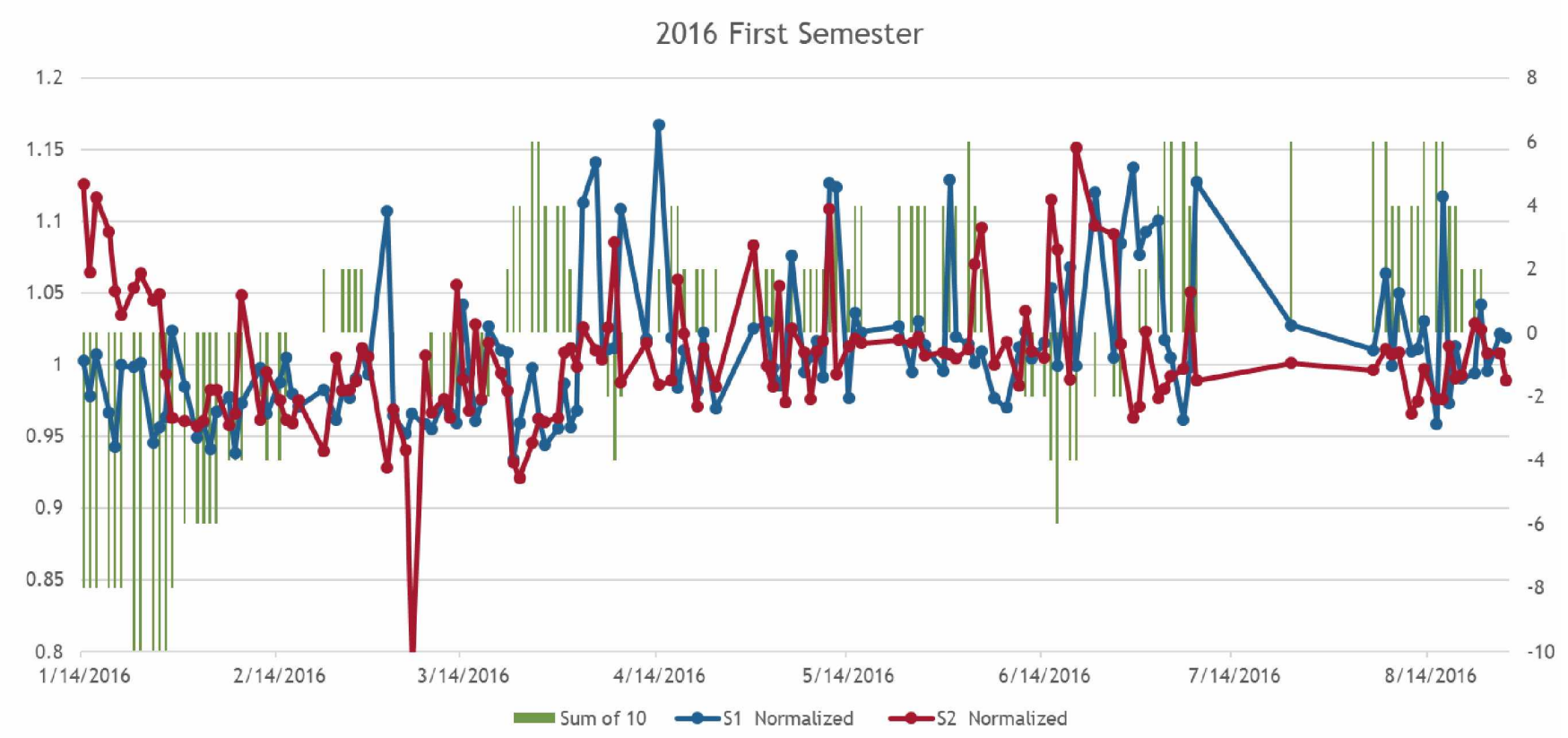

Figure 3-12 2016 First Semester

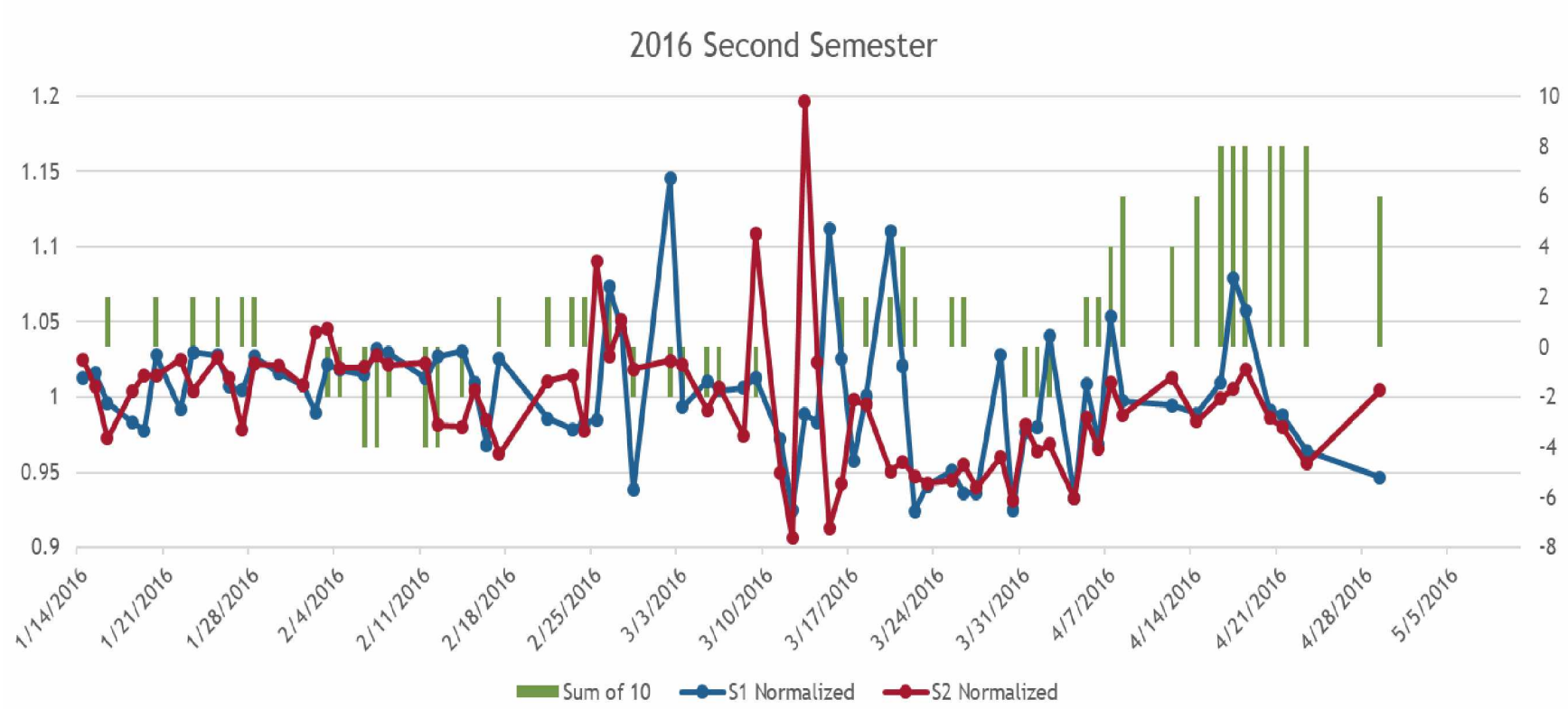

Figure 3-13 2016 Second Semester

The same analysis was carried out for 2017, although since this year has less data, the resulting graph is discontinuous at some points. The lack of data for specific periods of the year as well as lack of data from the primary heating sources result in a graph with very high points; there is no prevailing trend. This results in spikes in values presented, so there is not enough heat data to properly normalize the temperature. As a result, the Heat Ratio value ends up swinging between low and high values.

2017 had fewer extreme values (only one) and only at +8. This resulted in a year with an overall behavior that did not carry as much momentum as the years before. However, the scarcity of valid data has prejudiced the possibility of coming to any proper conclusions. The plots for the year of 2017 are presented in Figure 3-14 and Figure 3-15.

Analyzing the year of 2018, we had a very well-behaved year; the first semester had no overall extreme values, only a slight trend of high values for S1. The second semester did have three extreme values in close sequence, but they trended down very quickly. It was the only year where no underlying trend heavily skewed the results toward one end or the other during a significant time frame. This year is presented in Figure 3-16 and Figure 3-17.

When seeding errors to test the algorithm, these trends will hide the error at some points, as a substantial trend towards one end builds up and takes time to be weighted towards the bias inserted.

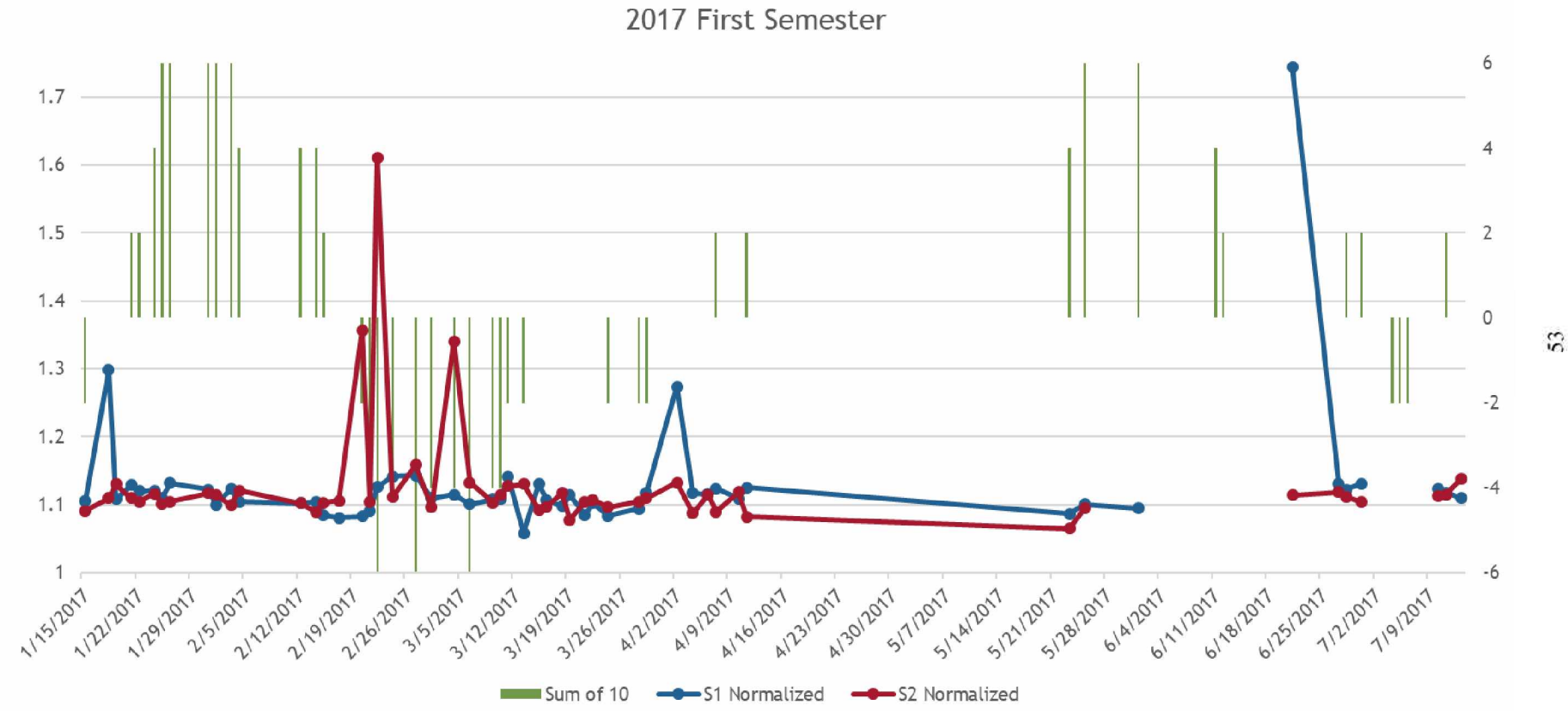

Figure 3-14 2017 First Semester

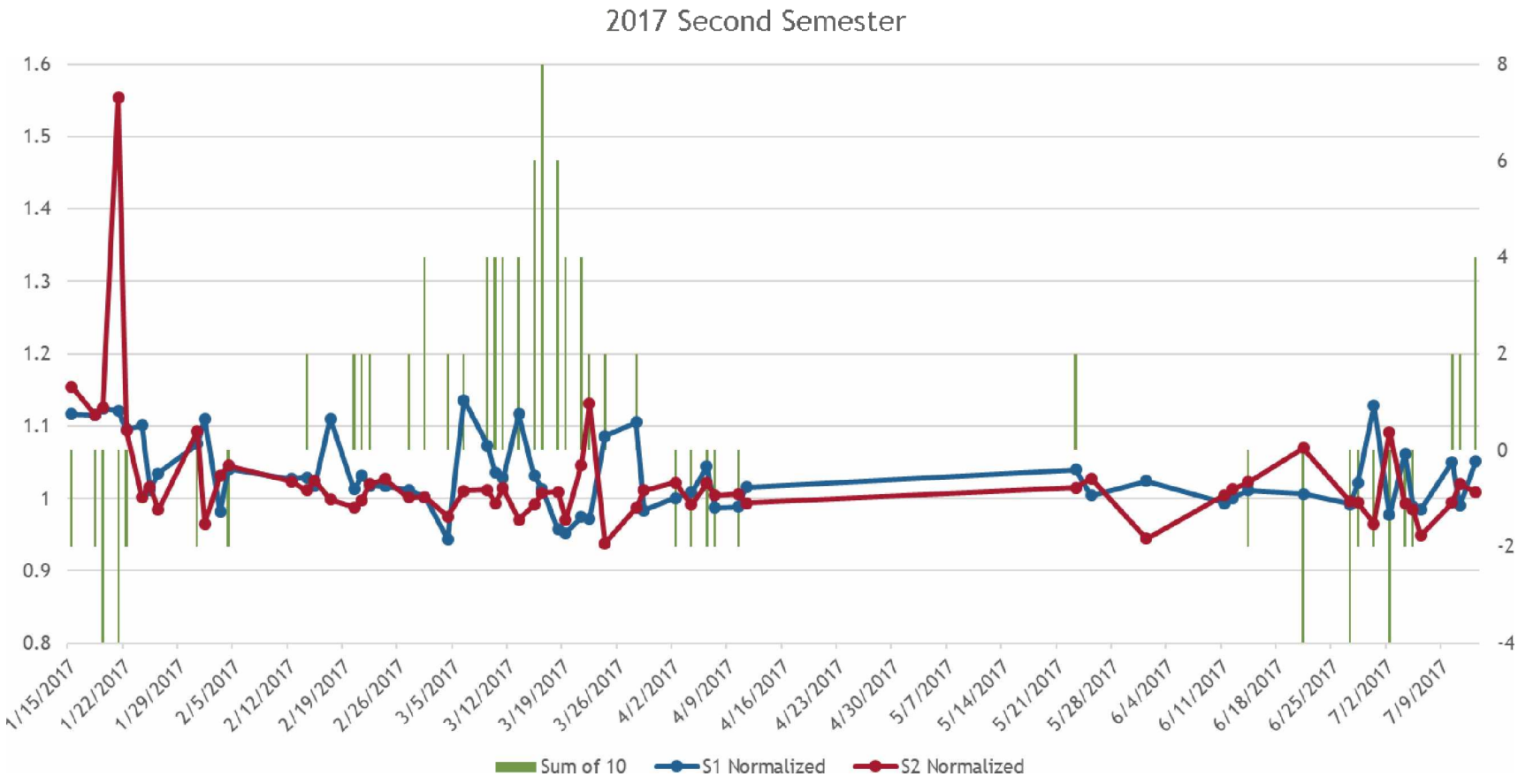

Figure 3-15 2017 Second Semester

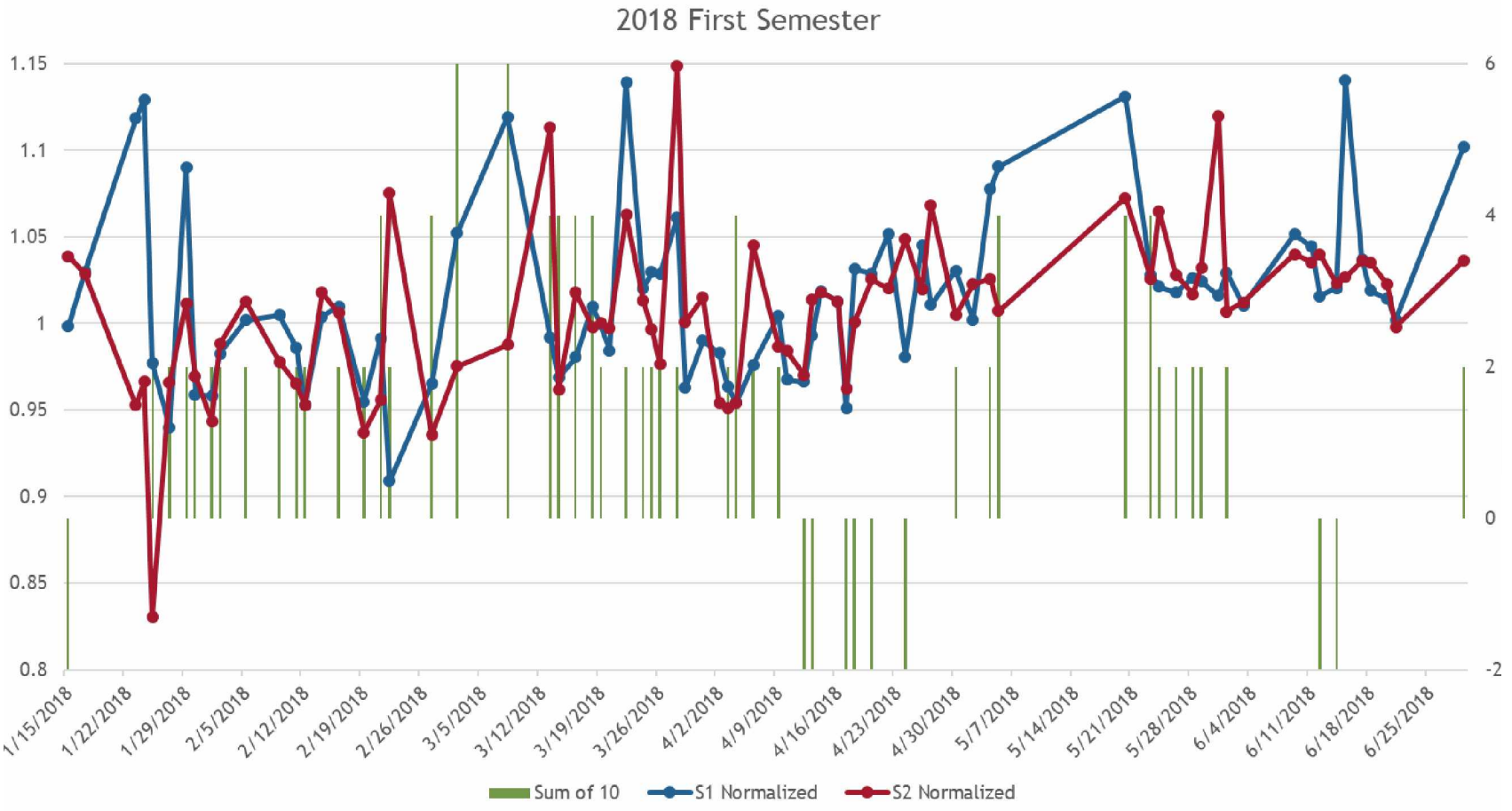

Figure 3-16 2018 First Semester

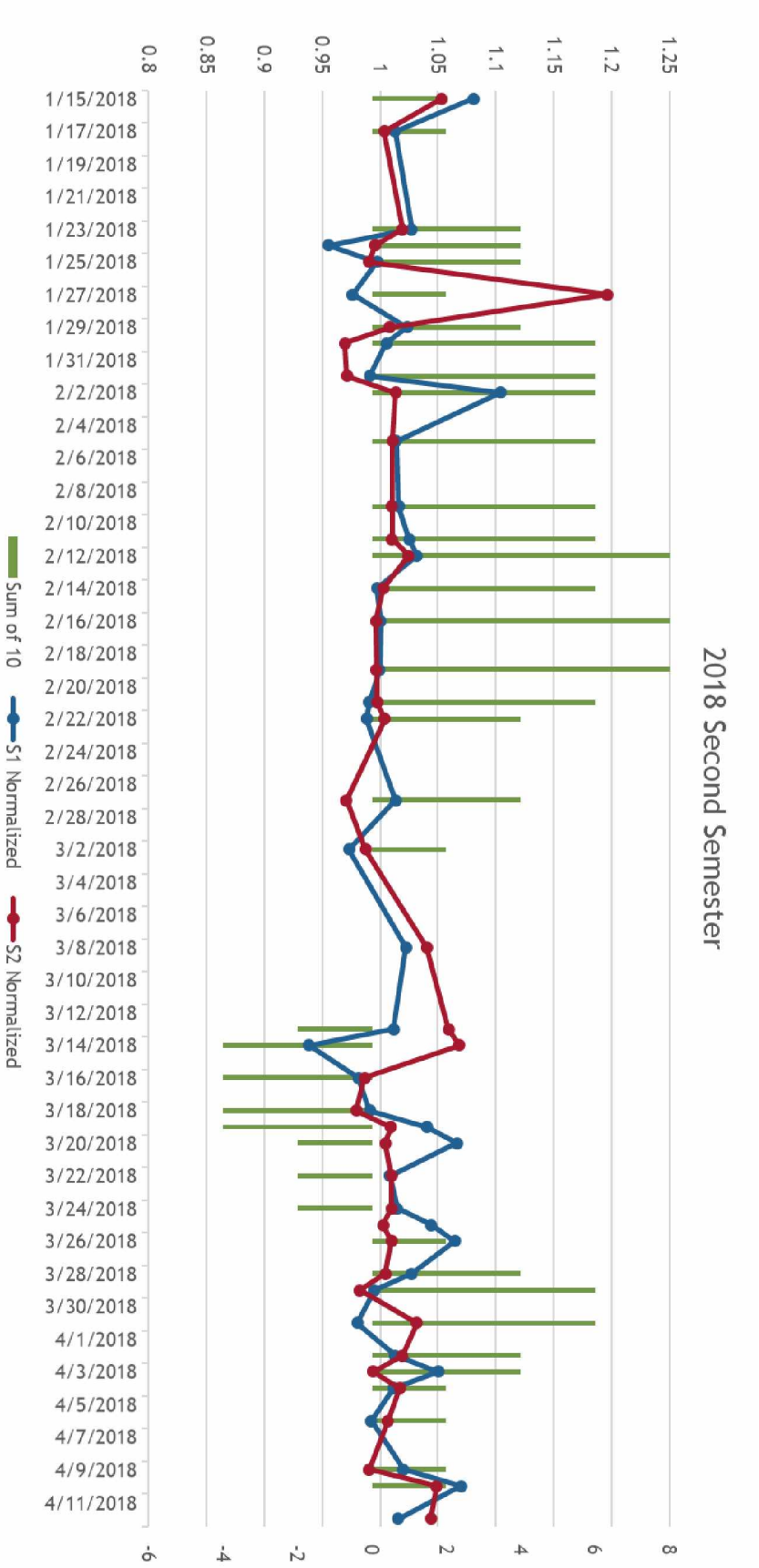

Figure 3-17 2018 Second Semester

After analyzing these patterns, it can be confirmed that the data is skewed in some time frames. This skewness works against the algorithm. The error detection at the opposite end of the skewness is hampered and may not even be detectable depending on how long the trend goes.

The operators from Pogo Mine shared that they did not have the data on the last calibration of the sensors. Lacking this precise point where an unbiased data set could be started hinders the optimization of the algorithm but, at least, the data that is present can be used to indicate an overall trend in error. Though the algorithm needs an unbiased process to succeed, an understanding of the bias can help improve the algorithm.

Chapter 4. Results

Four years of raw data, from 2015 to 2018, was available for testing the algorithm.

Pothina's research only had data from 2015 and 2016 to test the algorithm. He seeded the error by adding 2% to S1 temperature in a day and every single day after that, using all the days of the year, resulting in 365 scenarios in each year. He also did the same with -2% to S1. Obviously, this approach only tested for one possibility: when S1 had errors.

The new tests focus on trying to understand the validity of the algorithm on a longer dataset, and also on the possibility of the error being originated on values of S2.

4.1. Error Seeding

Having access to data for more years, this research applies this methodology of error seeding to the years not covered in Pothina's work. It also applies a similar seeding to S2. Additionally, in order to test the sensitivity to the algorithm, it will seed a Normal error of 2% (i.e. with a mean of 2%, instead of a fixed 2%) to S1 and S2 in all years. This is called the "noisy error" scenario. Since normally distributed errors have more noise than fixed errors, these tests aim to verify the robustness of the algorithm.

In order to generate values with a Normal distribution error of 2%, the algorithm created does the following:

1) seize the temperature value to have an error seeded;

2) create a normal distribution with 1000 values with a mean of 102% of the original value and a standard deviation of 1% of the original value; and

3) pick one of these values randomly and assign it as the new biased temperature value.

This is done systematically starting on the first day of the year and going until the end of the year. In the next iteration, error is seeded on the second day until the end of the year. This system results in 365 or 366 possibilities for each error scenario (2016 had 366 days).

The following scenarios are faced by the algorithm every year:

a) +2% of baseline S1 Temperature ("fixed" increase error scenario);

b) +2% of baseline S2 Temperature ("fixed" increase error scenario);

c) -2% of baseline S1 Temperature ("fixed" decrease error scenario);

d) -2% of baseline S2 Temperature ("fixed" decrease error scenario);

e) +2% of baseline S1 Temperature under a Normal Distribution (noisy error); and

f) +2% of baseline S2 Temperature under a Normal Distribution (noisy error).

This results in a total of 24 different experiments since there are six experiments for each year available. There is also the baseline scenario consisting of the raw data without the addition of any artificial error. The algorithm may declare some values in the baseline scenario as errors. These are called false or baseline alarms. These alarms occur in the baseline datasets because the true process behaved in a way that was different from what was defined in the algorithms. In order to differentiate these baseline errors from the seeded errors, the baseline/false alarms are filtered out from the analysis, as they were always present no matter when the seeding happened.

For clarity purposes the "fixed" increase in error scenario will be named as "FEIS", in the same way, the "fixed" decrease error scenario will be named "FEDS" and the noisy error scenario will be named as "NES".

The main results consider the time it took for the error to be detected since error is first inserted. A temperature value is called an "error" when the cross-score threshold is +8 (when error is in S1) or -8 (when error is in S2).

4.2. The Year of 2015

The year of 2015 has 17 errors on its baseline scenario; Table 4.1: describes these errors, the date they happen and the summation value of the last ten matches that trigged it.

Table 4.1: 2015 Baseline Scenario Errors

2015 Baseline Scenario Warnings			
Event Number	Date	Sum of 10	Scenario
1	Tuesday, May 5, 2015	8	Baseline
2	Wednesday, May 6, 2015	8	Baseline
3	Tuesday, June 16, 2015	8	Baseline
4	Wednesday, June 17, 2015	8	Baseline
5	Thursday, June 18, 2015	10	Baseline
6	Friday, June 19, 2015	10	Baseline
7	Sunday, June 21, 2015	8	Baseline
8	Monday, June 22, 2015	8	Baseline
9	Saturday, August 29, 2015	8	Baseline
10	Sunday, August 30, 2015	8	Baseline
11	Thursday, September 10, 2015	-8	Baseline
12	Wednesday, December 23, 2015	-8	Baseline
13	Friday, December 25, 2015	-8	Baseline
14	Saturday, December 26, 2015	-8	Baseline
15	Sunday, December 27, 2015	-8	Baseline
16	Monday, December 28, 2015	-8	Baseline
17	Wednesday, December 30, 2015	-8	Baseline

4.2.1. Error Seeding in S1 for 2015

To understand how the algorithm performs when error is seeded to S1, it is first important to look at the baseline errors in Table 4.1:. There are several times before August 30, 2015, where the cross-score naturally reaches 8 even when error is not seeded. This means, the statistics for S1 consistently beat out the statistics for S2 naturally without any addition of error. When error is added to S1, it does not have any effect on the cross-score, as statistics for S1 was already more than S2. Thus, the added errors are "masked" by the natural process.

When a positive error is seeded to S1, it's ratios consistently beats S2 ratios. The algorithm 1declares an "error" when the cumulative cut score exceeds the threshold of +8. The threshold of +8 may also be exceeded, when a negative error is seeded to S2, as even in that situation, the S1 ratios consistently beat S2 ratios. Therefore, when results are presented, the 102% error scenarios of one strip vessel are presented alongside 98% error scenarios for the other strip vessel.

Table 4.2: summarize the results on the performance of the algorithm when error is seeded to S1.

Table 4.2: 2015 Test Results When Threshold >8 triggers error

Year	2015		
Error Insertion Form	102% of S1 (NES)	102% of S1 (FEIS)	98% of S2 (FEDS)
Average time to Detect	23.1	24.4	21.6
75% Quartile of detection Time	31.4	35.8	29.6
Maximum time to detect	83.2	61.5	61.5
Average cycles to detect	19.8	18.7	18.8
75% quartile of Cycles to Detect	25.0	26.0	26.0
Max number of cycles to detect	74.0	57.0	57.0
Number of Detections	216.0	241.0	243.0
Number of Scenarios	365.0	365.0	365.0
Number of False Detections	10.0	10.0	10.0
Last day of detection	242.6	268.2	242.6
Seed Day of Last Detection	208.0	207.0	243.0

As can be seen, the maximum time to detect error is higher for noisy error scenario (NES) compared to fixed error scenario (FEIS and FEDS). The percentage of errors detected was about 66% (~242/365) for FEIS and FEDs, while it was 59% (216/365) for NES. Note that the success rate, computed by assuming 365 scenarios per year, is a simplistic measure of success. Since valid data points are not always available, the total scenarios possible in a year is usually less than 365. In some years (2017 and 2018), the total scenarios in a year is a lot lower than 365 due to large data gaps.

Other than that, there is not much of a difference between the two scenarios. The reason is that the randomized values might, in some cases, create streaks of values higher than 2% values and this allows, in some situations, for early detection.

It should also be noted that in the year in question (2015) errors are not detected after 208 days after the first day of the year. This is due to the original data having a lot of negative streaks

in the second semester, as can be seen in Table 4.1:. "Negative streaks" are where the statistics for S1 are continuously less than those for S2. "Positive streaks" are the opposite of negative streaks. These streaks, along with data gaps, explain when only 60-66% of errors were detected.

The results for decreasing S2 by 2% are like those of an increase of 2% in S1. This shows that the algorithm performs in a consistent form regardless of the direction of the error.

Another important information is that the amount of cycles needed to find the seeded error are similar in all the scenarios.

4.2.2. Error Seeding in S2 for 2015

For the errors seeded on S2, the last seven warnings of error in Table 4.1 are important to note as they would mask errors seeded to S2. In the later part of the year, statistics for S2 were naturally higher than those of S1. Therefore, when error is seeded to S2, a comparison of the statistics is no different than from the baseline case. Table 4.3 summarizes the results for the seeding to S2.

Table 4.3: 2015 Test Results When Threshold <-8 triggers error

Year	2015		
Error Insertion Form	102% of S2 (NES)	102% of S2 (FEIS)	98% of S1 (FEDS)
Average time to Detect	22.3	19.5	20.1
75% Quartile of detection Time	28.1	26.8	26.8
Maximum time to detect	209.3	55.9	57.9
Average cycles to detect	19.1	17.1	17.7
75% quartile of Cycles to Detect	24.0	23.0	23.0
Max number of cycles to detect	172.0	50.0	51.0
Number of Detections	322.0	342.0	343.0
Number of Scenarios	365.0	365.0	365.0
Number of False Detections	7.0	7.0	7.0
Last day of detection	354.7	354.7	354.7
Seed Day of Last Detection	344.0	345.0	337.0

As the time distribution of the errors in the baseline favors S2 at the end of the year, the algorithm can detect errors at a much later date. As with S1, time to detect is higher for NES than

FES. It is the expected behavior; as the seeded error distribution is randomized, it results in a long time to detect, as expected. The success rate of 88% and 95% for NES and FEIS/FEDS is much higher than when errors were seeded into S1.

In 2015, the number of detections is less when the error (in S1 or S2) is normally distributed.

4.3. The Year of 2016

2016 presents 18 errors in the original data; the positive streak of errors is located at the end of the year, while the negative streak happens at the beginning of the year. Table 4.4 describes the dates of these events.

Table 4.4: 2016 Baseline Scenario Errors

2016 Baseline Scenario Warnings			
Event Number	Date	Sum of 10	Scenario
1	Friday, December 23, 2016	8	Baseline
2	Sunday, December 25, 2016	8	Baseline
3	Monday, December 26, 2016	8	Baseline
4	Tuesday, December 27, 2016	8	Baseline
5	Wednesday, December 28, 2016	8	Baseline
6	Thursday, December 29, 2016	8	Baseline
7	Thursday, January 14, 2016	-8	Baseline
8	Friday, January 15, 2016	-8	Baseline
9	Saturday, January 16, 2016	-8	Baseline
10	Monday, January 18, 2016	-8	Baseline
11	Tuesday, January 19, 2016	-8	Baseline
12	Wednesday, January 20, 2016	-8	Baseline
13	Friday, January 22, 2016	-10	Baseline
14	Saturday, January 23, 2016	-10	Baseline
15	Monday, January 25, 2016	-10	Baseline
16	Tuesday, January 26, 2016	-10	Baseline
17	Wednesday, January 27, 2016	-10	Baseline
18	Thursday, January 28, 2016	-8	Baseline

4.3.1. Error Seeding of in S1 for 2016

As before, Table 4.4 shows the situations in baseline case that would mask errors added to S1 or S2. Looking at Table 4.4, masking of S1 errors is unlikely before December 23, when the cut score first naturally rises to 8. Table 4.5 Summarize the results for the seeding towards S1.

Table 4.5: 2016 Test Results When Threshold >+8 triggers error

Year	2016		
Error Insertion Form	102% of S1 (NES)	102% of S1 (FEIS)	98% of S2 (FEDS)
Average time to Detect	34.9	29.3	29.7
75% Quartile of detection Time	46.1	43.6	43.8
Maximum time to detect	251.1	74.7	75.7
Average cycles to detect	23.8	20.2	20.4
75% quartile of Cycles to Detect	32.0	30.0	30.0
Max number of cycles to detect	171.0	47.0	47.0
Number of Detections	331.0	353.0	354.0
Number of Scenarios	366.0	366.0	366.0
Number of False Detections	6.0	6.0	6.0
Last day of detection	356.8	356.8	356.8
Seed Day of Last Detection	354.0	340.0	354.0

As expected, the time to detect is higher for NES than FES. However, as in 2015, time to detect is quite reasonable for both scenarios. The success rates are also much higher than in 2015. Since the errors toward the negative side are located at the start of the year, the algorithm can detect the errors, but at a much later date.

The tardiness in the detection also influences the amount of cycles the algorithm takes to detect; this happens for the same reason as stated above. This year has the major outliers in cycles needed to detect, all the other results take similar amount of cycles to detect the seeded error. Error Seeding of in S2 for 2016

For the errors seeded on S2, the last twelve warnings of error are the important ones, as they are the errors that would mask the seeded increase. Table 4.6 summarize the results for the seeding towards S2.

Table 4.6: 2016 Test Results When Threshold <-8 triggers error

Year	2016		
Error Insertion Form	102% of S2 (NES)	102% of S2 (FEIS)	98% of S1 (FEDS)
Average time to Detect	57.9	59.4	47.5
75% Quartile of detection Time	83.7	89.7	73.6
Maximum time to detect	246.7	143.5	108.5
Average cycles to detect	38.9	43.2	34.4
75% quartile of Cycles to Detect	54.0	61.0	49.0
Max number of cycles to detect	75.0	111.0	83.0
Number of Detections	259.0	271.0	270.0
Number of Scenarios	366.0	366.0	366.0
Number of False Detections	12.0	12.0	12.0
Last day of detection	353.7	275.7	275.7
Seed Day of Last Detection	330.0	273.0	271.0

As the errors toward S1 concentrate at the very end of the year, the algorithm is not capable of detecting errors at later dates, in a similar fashion to what happened to S1 in 2015. Since the normally distributed error seed has a variance, it allows the algorithm to detect later in some cases, but in overall, up to the third quartile, the results are close to the detection for an increase made by a set increase. The average time to detect is much higher when S2 has error compared to when S1 has errors. This is since the errors in the baseline are on the beginning of the year, the whole month of January is marked by large scores on the negative side. Due to filtering, all these results are not used. So, the detection can only be truly detected after the end of this month, increasing the detection time. The success rates are lower than when errors were seeded in S1.

This also results in an increase in the number of cycles needed to detect the seeded errors.

4.4. The Year of 2017

The algorithm pronounced only one data point as erroneous in 2017 in the baseline scenario. This is due to the lack of many valid data points in this year (the heat exchangers were down quite a bit in 2017). This is not a typical operational situation. The lack of valid data points

will increase the time to detect, as the valid cycles will be less present. Table 4.7 shows the date of the error present on the baseline.

Table 4.7: 2017 Baseline Scenario Errors

2017 Baseline Scenario Warnings			
Event Number	Date	Sum of 10	Scenario
1	Saturday, November 11, 2017	8	Baseline

4.4.1. Error Seeding of in S1 for 2017

In Table 4.7, the only error "present" in S1 happens at the end of the year. Therefore, it does not mask the overall results for the test. Table 4.7 presents the results.

Table 4.8: 2017 Test Results When Threshold >8 triggers error

Year	2017		
Error Insertion Form	102% of S1 (NES)	102% of S1 (FEIS)	98% of S2 (FEDS)
Average time to Detect	73.7	65.4	65.5
75% Quartile of detection Time	114.8	108.3	108.8
Maximum time to detect	285.9	190.8	191.8
Average cycles to detect	23.8	20.6	20.7
75% quartile of Cycles to Detect	35.0	29.5	30.0
Max number of cycles to detect	106.0	45.0	46.0
Number of Detections	333.0	333.0	335.0
Number of Scenarios	365.0	365.0	365.0
Number of False Detections	1.0	1.0	1.0
Last day of detection	346.5	346.5	346.5
Seed Day of Last Detection	303.0	303.0	303.0

The success rate for detecting error is high, at about 92% (~334/365). The longer detection time can be explained by the lack of overall data and the fact that the cycles will be more sparsely distributed in this situation, as stated above.

One important result is that, even under these conditions, the average number of cycles the algorithm took is like those found in 2015. These results implicate that, for algorithm, the amount of cycles is more relevant than time in order to detect the errors.

4.4.2. Error Seeding of in S2 for 2017

The algorithm does not find any "errors" in S2 in the baseline case in 2017. Table 4.9 summarizes the performance of the algorithm when error is seeded in S2.

Table 4.9: 2017 Test Results When Threshold <-8 triggers error

Year	2017		
Error Insertion Form	102% of S2 (NES)	102% of S2 (FEIS)	98% of S1 (FEDS)
Average time to Detect	61.6	53.9	54.8
75% Quartile of detection Time	95.9	85.7	86.2
Maximum time to detect	214.3	132.2	133.2
Average cycles to detect	26.2	22.6	23.1
75% quartile of Cycles to Detect	36.0	28.0	28.0
Max number of cycles to detect	92.0	56.0	56.0
Number of Detections	238.0	189.0	191.0
Number of Scenarios	365.0	365.0	365.0
Number of False Detections	0.0	0.0	0.0
Last day of detection	349.6	220.3	220.3
Seed Day of Last Detection	339.0	193.0	191.0

Similarly, to S1, the lack of valid points in 2017 resulted in an overall more prolonged time to detect. The number of cycles to detect is like 2015, as with S1. Success rate is significantly lower due to data gaps.

4.5. The Year of 2018

2018 presents three errors in the original data. They are all located toward S1 and in the second semester. This is a small streak, so this should not result in significant issues for the algorithm.

Table 4.10 describes these points and the dates they occur.

Table 4.10: 2018 Baseline Scenario Errors

2018 Baseline Scenario Warnings			
Event Number	Date	Sum of 10	Scenario
1	Sunday, August 19, 2018	8	Baseline
2	Thursday, August 30, 2018	8	Baseline
3	Saturday, September 1, 2018	8	Baseline

4.5.1. Error Seeding of in S1 for 2018

The three errors present in 2018 are all towards S1 since the errors are in the middle of the year, they are not able to mask the seeding of errors. Table 4.11 describes these results.

Table 4.11: 2018 Test Results When Threshold >8 triggers error

Year	2018		
Error Insertion Form	102% of S1 (NES)	102% of S1 (FEIS)	98% of S2 (FEDS)
Average time to Detect	48.4	35.7	35.7
75% Quartile of detection Time	56.0	51.1	51.1
Maximum time to detect	297.7	70.3	70.3
Average cycles to detect	28.1	21.4	21.5
75% quartile of Cycles to Detect	30.0	30.0	30.0
Max number of cycles to detect	165.0	46.0	46.0
Number of Detections	302.0	319.0	319.0
Number of Scenarios	365.0	365.0	365.0
Number of False Detections	3.0	3.0	3.0
Last day of detection	341.1	341.1	341.1
Seed Day of Last Detection	320.0	320.0	320.0

2018 results are like the results from 2015/2016. However, since there were significant gaps in the data, the performance is not as good as 2015/2016. However, data gaps were not as bad as 2017. Also, the number of cycles needed to detect are closer to the results of 2015 and 2017. The success rate is 83-87% for cases when error is triggered cumulative cut score exceeds 8.

4.5.2. Error Seeding of in S2 for 2018

The algorithm does not call any errors in S2 in the baseline scenario. Therefore, there is no masking of the seeded errors. Table 4.12 summarizes the performance of the algorithm when errors are seeded in S2 in 2018:

Table 4.12: 2018 Test Results When Threshold <-8 triggers error

Year	2018		
Error Insertion Form	102% of S2 (NES)	102% of S2 (FEIS)	98% of S1 (FEDS)
Average time to Detect	49.9	44.5	46.2
75% Quartile of detection Time	65.4	61.5	51.1
Maximum time to detect	235.1	96.3	98.3
Average cycles to detect	29.4	26.5	27.5
75% quartile of Cycles to Detect	37.0	36.0	36.0
Max number of cycles to detect	137.0	58.0	59.0
Number of Detections	244.0	243.0	242.0
Number of Scenarios	365.0	365.0	365.0
Number of False Detections	0.0	0.0	0.0
Last day of detection	359.5	262.3	262.3
Seed Day of Last Detection	306.0	244.0	242.0

The results show a similar pattern to S1; they take roughly the same amount of time to detect the errors, showing a good response of the algorithm. The success rate is about 67%.

The tables presented next provide a different look at the results above.

Table 4.13: Performance of algorithm for FEIS (102%)

Seed Form	102% (FEIS)							
	S1				S2			
Year	2015	2016	2017	2018	2015	2016	2017	2018
Average cycles to detect	18.7	20.2	20.6	21.4	17.1	43.2	22.6	26.5
75% quartile of Cycles to Detect	26.0	30.0	29.5	30.0	23.0	61.0	28.0	36.0

Table 4.14: Performance of algorithm for FEDS (98%)

Seed Form	98% (FEDS)							
	S1				S2			
Year	2015	2016	2017	2018	2015	2016	2017	2018
Average cycles to detect	17.7	34.4	23.1	27.5	18.8	20.4	20.7	21.5
75% quartile of Cycles to Detect	23.0	49.0	28.0	36.0	26.0	30.0	30.0	30.0

Table 4.15: Performance of algorithm for NES (102%)

Seed Form	102% NES							
	S1				S2			
Year	2015	2016	2017	2018	2015	2016	2017	2018
Average cycles to detect	19.8	23.8	23.8	28.1	19.1	38.9	26.2	29.4
75% quartile of Cycles to Detect	25.0	32.0	35.0	30.0	24.0	54.0	36.0	37.0

Chapter 5. Summary and Conclusions

This thesis presented a new perspective on an error detection algorithm that was developed by Pothina. The algorithm was developed to detect small errors in the strip vessel temperature sensors in the carbon stripping circuit at Pogo mine. The algorithm was tested by seeding artificial errors into real data obtained from Pogo mine.

An analysis of the data, the testing conditions for the algorithm, and the results, are summarized below:

- Errors were seeded to both strip vessels, S1 and S2, and not just S1. However, errors were seeded only one sensor at a time.
- Three types of errors were seeded, fixed error and noisy error:
    - a fixed 2% error added to sensors S1 and S2 (fixed error increase)
    - a fixed decrease in 2% was applied to sensors S1 and S2 (fixed error decrease)
    - an error with a mean and standard deviation of 2% and 0.01 % of the magnitude of S1 and S2, added to the sensors respectively (noisy error)
- The algorithm was applied year by year, for each of the four years, 2015, 2016, 2017 and 2018. Overall, the average time to detect errors ranged from 19 days to 73 days. The time to detect was very high (53 to 73 days) in the year of 2017 due to scarcity of valid data points in this year.
- The algorithm performed well even under the noisy error scenario. The performance was close to that when the applied error was a fixed 2%. The increase in time for detection was 10% for most scenarios. The time to detect increased 40% only in 2018. This is simply because there were large gaps in the data set that year.

- When matching the performance for S1 and S2, the results follow the skewness present in the data. A year, like 2016, that had a large number of extreme values in only one of the strip vessels, resulted in performance differences that were significant. For example, 29 days for detecting errors in S1 compared to 59 days for S2. For years without such high concentrations, like 2018, resulted in much smaller differences. For example, 35 days for S1 and 44 days for S2.

- The original assumption that the two strip vessels operated under similar condition was tested. Statistical analysis of the results of the algorithm on baseline data highlighted how individual strip vessels maintained biases in their relationship with heat exchangers for extended periods of time even before errors were added. These inherent biases impacted when the algorithm succeeded or failed. For example, the algorithm was unable to detect +2% error in the last quarter of 2015 because the temperature ratios for S1 were generally lower than S2 during that time of the year.

- Another very relevant result is that the algorithm performance (time to detect) is consistent when expressed in 'number of cycles to detect', than when expressed in 'number of days to detect'. This is because the days to detect is impacted by data gaps.

- When tested under noisy error data (data that had a normally distributed error added), the algorithm performed almost as well as it did with fixed (2%) errors, with only the maximum time to detect being longer.

The main lesson of the thesis is that the relative difference in how the two strip vessels are operated significantly impacts the algorithm. When the temperature ratios (strip vessel to heat exchangers) are consistently higher for S1 compared to S2, the algorithm will succeed in detecting positive errors in S1. If the S1 ratios are consistently lower than S2, it will have less success in detecting positive errors in S1. This relationship between S1 ratios and S2 ratio directly stems from how the plant is operated.

The lack of calibrated data, i.e., data that has been collected after calibration, is a relevant issue. The relationships developed in this thesis assume that the data was "pure", i.e. it had no

errors. Ideally, relationships for heuristic algorithms such as those in this thesis, should be developed from data known to be reliable.

A true test of the algorithm will first require collection of reliable data. All sensors must be kept calibrated for several months while the data is collected. The cut-scores used to trigger error alerts will most likely have to be adjusted to reflect the new data. It is only after this, can the algorithm be deployed for field testing.

References

Abe, I. (2010). From Exploration and Mine Construction through Successful Production in the Pogo Gold Mine , Alaska , USA. *Journal of the Mining and Materials Processing Institute of Japan.*, *126*(8), 477–481.

Adams, M. D. (2005). *Advances in Gold Ores Processing. Developments in Mineral Processing* (Vol. 15). https://doi.org/10.1073/pnas.0703993104

Balaban, E., Saxena, A., Bansal, P., Goebel, K. F., & Curran, S. (2009). Modeling, detection, and disambiguation of sensor faults for aerospace applications. *IEEE Sensors Journal*, *9*(12), 1907–1917. https://doi.org/10.1109/JSEN.2009.2030284

Baljak, V., Tei, K., & Honiden, S. (2012). Classification of Faults in Sensor Readings with Statistical Pattern Recognition, (c), 270–276.

Boyes, W. (2003). *Instrumentation Reference Book*. (W. Boyes, Ed.) (3rd Editio). Elsevier Science Ltd.

Brown, M. (2012). Data mining techniques. Retrieved November 7, 2018, from https://www.ibm.com/developerworks/library/ba-data-mining-techniques/

Buxton, M., & Benndorf, J. (2013). The Use of Sensor Derived Data in Optimization along the Mine-Value-Chain: An overview and assessment of techno-economic significance. *142nd SME Annual Meeting and Exhibit*, (September), 324–336. Retrieved from http://gekkos.com.au/sites/default/files/documents/TechnicalPaper033TheInLineLeachReactorInstallationAtTeckPogIncPogoMine.pdf

CHIEREGATI, A. C., & PITARD, F. F. (2018). Amostragem. In A. Benvindo da Luz França, S. Cristina Alves França, & P. Fernando Almeida Braga (Eds.), *Tratamento de Minérios* (6th

editio, pp. 27–65). CETEM/MCTIC.

Engineers, I. of E. and E. (2010). "INTERNATIONAL STANDARD ISO / IEC / IEEE transducer interface for sensors and communication protocols and formats," *2010*, 82. Retrieved from https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6155051

Fast, P. E. J. L. (1989). Carbon Stripping – Denver Mineral Engineers. Retrieved August 6, 2018, from http://www.denvermineral.com/ carbon-stripping/

Fraden, J. (2016). *Handbook of modern sensors: Physics, designs, and applications*. *Handbook of Modern Sensors: Physics, Designs, and Applications*. https://doi.org/10.1007/978-3-319-19303-8

Gautschi, G. (2002). *Piezoelectric Sensorics*. https://doi.org/10.1007/978-3-662-04732-3

Hong, W. (2009). Data Driven Fault Diagnosis and Fault Tolerant Control : Some Advances and Possible New Directions, *35*(6). https://doi.org/10.3724/SP.J.1004.2009.00739

Hou, Z., Lian, Z., Yao, Y., & Yuan, X. (2006). Data mining based sensor fault diagnosis and validation for building air conditioning system. *Energy Conversion and Management*, *47*(15–16), 2479–2490. https://doi.org/10.1016/j.enconman.2005.11.010

Konigsmann, E., Metallurgist, S., Cominco, T., Abols, J., Manager, S., Systems, G., … Operations, T. (2017). The InLine Leach Reactor Installation at Teck- Pogo Inc . Pogo Mine, 1–11. Retrieved from http://gekkos.com.au/sites/default/files/documents/TechnicalPaper033TheInLineLeachReactorInstallationAtTeckPogIncPogoMine.pdf

Kusiak, A., & Song, Z. (2009). Sensor Fault Detection in Power Plants. *Journal of Energy*

Engineering-Asce, *135*(4), 127–137. https://doi.org/10.1061/(ASCE)0733-9402(2009)135:4(127)

Leek, J. (2015). *The Elements of Data Analytic Style*. LeanPub.

Longley, R. J., McCallum, A., & Katsikaros, N. (2003). Intensive cyanidation: Onsite application of the InLine Leach Reactor to gravity gold concentrates. *Minerals Engineering*, *16*(5), 411–419. https://doi.org/10.1016/S0892-6875(03)00054-2

Muir, D. M., Hinchliffe, W., Tsuchida, N., & Ruane, M. (1985). Solvent elution of gold from C.I.P. carbon. *Hydrometallurgy*, *14*(1), 47–65. https://doi.org/10.1016/0304-386X(85)90005-2

Mular L., A., Halbe, D. N., & Barratt, D. J. (2002). *Mineral Processing Plant Design, Practice, and Control. Proceedings. Mineral Processing Plant Design, Practice, and Control. Proceedings* (Vol. 1–2).

Narasimhan, S., & Rengaswamy, R. (2004). Characterization of value for sensor networks for process fault diagnosis. *AIChE Annual Meeting, Conference Proceedings*.

National Institute of Intrumentation Starndards. (2018). Concepts of accuracy, precision, and statistical.

Pogo Mine. (2018a). Our Operation – Pogo Mine Alaska. Retrieved August 6, 2018, from http://pogominealaska.com/our-operation/

Pogo Mine. (2018b). Who We Are – Pogo Mine Alaska. Retrieved August 6, 2018, from http://pogominealaska.com/who-we-are/

Pothina, R. (2017). *Automatic detection of sensor calibration errors in mining industry*.

University Of Alaska-Fairbanks. Retrieved from https://scholarworks.alaska.edu/handle/11122/8137

Rupitsch, S. J. (2017). *Piezoelectric Sensors and Actuators Fundamentals and Applications*. (C. P. Bergman, Ed.), *Topics in Mining, Metallurgy and Materials Engineering* (1st ed.). Erlanger: Springer.

Soleimani, M., & Kaghazchi, T. (2008). Gold recovery from loaded activated carbon using different solvents. *Journal of the Chinese Institute of Chemical Engineers*, *39*(1), 9–11. https://doi.org/10.1016/j.jcice.2007.11.004

U.S Department of Energy. (2002). Mining overview. *ITP Mining: Energy and Environmental Profile of the U.S. Mining Industry*, 1–26. Retrieved from www1.eere.energy.gov/industry/mining/pdfs/overview.pdf?

Webster, J. G., & Eren, H. (2014). *Measurement, Instrumentation, and Sensors Handbook, Second Edition: Spatial, Mechanical, Thermal, and Radiation Measurement: John G. Webster, Halit Eren*.

Yang, C., Liu, C., Zhang, X., Nepal, S., & Chen, J. (2015). A time efficient approach for detecting errors in big sensor data on cloud. *IEEE Transactions on Parallel and Distributed Systems*, *26*(2), 329–339. https://doi.org/10.1109/TPDS.2013.2295810

Zadra, J. B. (1952). *Process for recovering gold and silver from activated carbon by leaching and electrolysis /*. Washington, D.C. : Retrieved from http://hdl.handle.net/2027/mdp.39015078535732