CORRECTION OF ERRORS DUE TO RADIAL RUN-OUT IN ABSOLUTE ROTARY ENCODERS

A Thesis Submitted to
the Graduate School of Engineering and Sciences of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in Electronics and Communication Engineering

by
Mehmet Onur CİRİT

December 2018
İZMİR

We approve the thesis of **Mehmet Onur CİRİT**

Examining Committee Members:

Prof. Dr. Mehmet ENGİN
Department of Electrical and Electronics Engineering
Ege University

Assoc. Prof. Dr. Şevket GÜMÜŞTEKİN
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

Assist. Prof. Dr. Barbaros ÖZDEMİREL
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

24 December 2018

Assist. Prof. Dr. Barbaros ÖZDEMİREL
Supervisor, Department of Electrical and Electronics Engineering
İzmir Institute of Technology

Prof. Dr. Enver TATLICIOĞLU
Head of the Department of
Electrical and Electronics Engineering

Prof. Dr. Aysun SOFUOĞLU
Dean of the Graduate School of
Engineering and Sciences

ACKNOWLEDGMENTS

To begin with, I would like to express my sincere gratitude to my supervisor Assist. Prof. Dr. Barbaros Özdemirel for his patience and support throughout my studies. When it comes to decision making, I am always extra careful and I take my time even with minor decisions. He always respected that but also kept me in pace so that I can finish my work on time. He provided his guidance whenever I am doubtful and pushed me to do better by inspiring me with new questions and ideas. Furthermore, he was always generous when it comes to providing his valuable feedback and his recommendations were always tremendously accurate. I am and always will be very grateful for all the effort he had put into my education.

Moreover, I would like to thank my dear friends and colleagues Cihan Alp, Çağın Ekici, Önder Yılmaz and Aslı Taşçı for their helpful recommendations and their support. Special thanks to Berna Derya Deniz for helping me with drawing some of the complicated figures.

Last but not least, I would like to thank to my parents Nurten Şimşek and Ali Cengiz Cirit and my brother Mert Haydar Cirit for their endless support throughout the entire educational life of mine.

ABSTRACT

CORRECTION OF ERRORS DUE TO RADIAL RUN-OUT IN ABSOLUTE ROTARY ENCODERS

This thesis proposes a novel, general purpose correction algorithm for analog absolute rotary encoders to eliminate errors due to radial run-out of the shaft. Unlike traditional quadrature encoders, four sensors were employed that produce four quadrature signals instead of two. The radial variation of the field intensity was exploited to identify the direction and extent of radial run-out and correction was applied to poorly identified position values accordingly. A simulation environment was created from the scratch to simulate encoder signals under the influence of shaft run-out in order to verify the performance of the algorithm.

The numerical results were collected in each milestone of the development and results were presented both for overall performance and for a number of special cases. The issues occurring in the first iterations of the algorithm, such as error mismatch and singularities, were identified and resolved in a stepwise manner. The final version of the algorithm has shown significant improvement and successfully reduced the mean error in angular position down to 12% of the initial value.

ÖZET

MUTLAK DEĞERLİ DÖNER ALGILAYICILARDA RADYAL BOŞLUKTAN KAYNAKLANAN HATALARIN DÜZELTİLMESİ

Bu tezde, mutlak değerli döner kodlayıcıların mekanik bozukluklarından bir çeşidi olan radyal boşluğun açı ölçümü üzerine etkisi incelenmiş ve bu sebeple oluşan hataları düzeltmek için bir yöntem öne sürülmüştür. Söz konusu yöntem, iki algılayıcı kullanan klasik döner kodlayıcılardan farklı olarak dört algılayıcıdan faydalanmakta ve açı ölçümünü bu dört algılayıcının oluşturduğu dört sinüs biçimli dalgadan sağlamaktadır. Bu doğrultuda, radyal boşluk sebebiyle şaftta oluşan kaymanın yönü ve miktarı, ölçülen alanın radyal değişiminden yararlanılarak hesaplanmış, bulunan değerler yardımıyla yanlış hesaplanan açı düzeltilmiştir. Yöntem temelde belli açılar ve izin verilen maksimum kayma miktarında sistematik bir şekilde veri toplanmasına dayanmaktadır. Şaftta oluşan kaymanın yönü ve miktarı yalnızca toplanılan verilerin benzerlikleri kullanılarak saptanmıştır.

Önerilen yöntem, geliştirme sürecinin her aşamasında kapsamlı olarak test edilmiş ve sonuçlar hem genel başarım hem de önemli senaryolar nicelendirilerek raporlanmıştır. Yöntemin yetersizlikleri, geliştirme sürecinin her aşamasında incelenmiş ve bu yetersizliklerin aşılması için çözümler sunulmuştur. Düzeltme yönteminin son hali belirgin bir gelişme göstererek radyal boşluktan kaynaklanan açı ölçüm hataları ilk değerlerinin ortalama %12'si düzeyine düşürülmüştür.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

θ_z ..... Correct Angular Position of the Shaft
ϕ_{ro} ..... Shaft Run-out Direction
R_{ro} ..... Shaft Offset from the Center
R_{roM} ..... Maximum Shaft Run-out Range
I_f ..... Field Intensity Function
I_{fR} ..... Radial Field Variation
R_{fm} ..... Middle Radius of the Field Ring
w_{fh} ..... Half Width of the Field Ring
I_{fA} ..... Angular Field Variation
I_{fos} ..... Field Intensity Offset
S ..... Received Data
S_c ..... Center Calibration Data
S_{ro} ..... Run-out Calibration Data
E_{ro} ..... Error Map
ΔS_c ..... Sensor Error Between Received Data and Sensor Calibration Data
ΔS_{cro} ..... Sensor Error Between Run-out Calibration Data and Sensor Calibration Data
D ..... Total Absolute Distance of Normalized Errors
ϕ_{roE} ..... Run-out Direction Estimate
W_c ..... Weight of Center Calibration Data
W_{ro} ..... Weight of Run-out Calibration Data
θ_{zC} ..... Corrected Angular Position

LIST OF ABBREVIATIONS

ANN . . . . . Artificial Neural Network
DLL . . . . . Dynamic Link Library
EMI . . . . . Electromagnetic Interference
GUI . . . . . Graphical User Interface
LSB . . . . . Least Significant Bit
LUT . . . . . Lookup Table

CHAPTER 1

INTRODUCTION

Providing accurate position measurements to motion control systems is an important task since the control applications essentially rely on the feedback from these measurements. Even though the current rotary encoders are capable of providing high resolution measurements, electrical and mechanical deficiencies can cause deterioration in the signals that jeopardize the accuracy and precision of the measurements by introducing position calculation errors. To overcome the errors due to mechanical deficiencies, couplings and ball bearings are commonly incorporated. However, these mechanical devices also introduce disadvantages such as lag, system complexity and cost. Besides, they are susceptible to wear issues, they require routine maintenance, and their ability to compensate mechanical errors are limited to their precision. Therefore, developing adequate error correction methods to eliminate errors due to mechanical deficiencies is important to minimize the dependence on these mechanical components and thus reducing the overall cost while increasing the reliability.

1.1. Error Sources in Rotary Encoders

Rotary encoders are susceptible to both electrical and mechanical deficiencies due to being composed of components of both kinds. Generally, electrical components are superior to mechanical components in terms of accuracy, precision and repeatability. To begin with, high precision mechanical components are hard to machine and even if machined properly, mechanical movement wear the components out after a certain amount of time. Furthermore, the effects of wear compromise repeatability because damage done to the component can not be recovered in most cases. Besides, they need maintenance to function properly. Mainly due to these weaknesses, mechanical parts fail to match the quality requirements to be able to cope with their electrical counterparts.

As a natural consequence of being composed of mechanical parts, the accuracy of a rotary encoder is affected by mechanical deficiencies that result from component imperfections, improper installation, and dynamic effects of the motors. The main interface to mechanical measurand is established via a shaft which is also equipped with some distinctive pattern to encode angular position information. The connection between

shafts is formed with a coupling mechanism for precise alignment with the mechanical input. Furthermore, ball bearings are utilized in the shaft ends to minimize the play in the shaft. However, even with the best equipment, there is still some play, which is called run-out, in the shaft end that deteriorates the signals leading to faulty determination of the angular position. These interfacing components accommodate mechanical errors up to a degree which is defined by the precision of the equipment. As a rule of thumb in all manufacturing technologies, cost of components increases with their precision. Also, these mechanical components suffer from wear issues and they need to be lubricated periodically to function properly.

1.2. Effect of Run-out in Position Calculation

A numerical example can be helpful to understand the effect of run-out in position calculation. Figure 1.1 shows a standard encoder configuration where sensors are placed to specific locations to provide quadrature signals. Both sensors are normally R_{sns} away from the center, but since there is ΔR run-out in $-x$ direction, sensors are shifted by ΔR in $+x$ direction relative to the shaft position. In this case, shaft run-out can affect the signals provided by the S1 and S2 sensors differently.

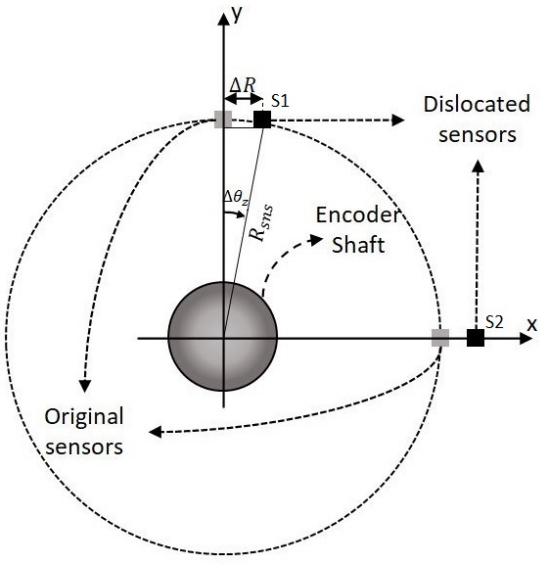


Figure 1.1. Shaft and sensor positions in the presence of run-out

The error in the position registered by S1 is directly affected by the run-out distance and it can be approximated by the corresponding angular rotation. The sine of the

angular error can be approximated by eq 1.1;

$$\sin\left(\Delta\theta_z\right) \approx \frac{\Delta R}{R_{sns}} \tag{1.1}$$

where $\Delta\theta_z$ is the angular position error, ΔR is the run-out distance and R_{sns} is the distance of the sensors from the center. This equation yields the position error $\Delta\theta_z$ as;

$$\Delta\theta_z \approx \arcsin\left(\frac{\Delta R}{R_{sns}}\right) \tag{1.2}$$

A normal class bearing has about a ball-gap of $25\mu m$ and a ring run-out of $15\mu m$ which makes a total of $\pm 40\mu m$ run-out ΔR. If the sensors are placed at $R_{sns} = 10mm$ away from the center, the error in angular position, $\Delta\theta_z$, is approximately $\pm 0.23^\circ$. In poorly specified bearings, ΔR can easily go up to $\pm 100\mu m$ which can result in up to $\pm 0.57^\circ$ error in angular position.

The second sensor, S2, is indirectly affected by the run-out. The coding field exerted by the shaft moves away or closer to the sensor as the shaft moves in $\pm x$ direction. The radial variation in the field intensity causes amplitude variation in the signal registered by S2 which in turn results in error in angular position. Normally this variation will add onto the error in θ_z, but it can also be utilized to identify the shaft motion.

1.3. Correction of Run-out Errors

Correction of position errors caused by run-out opens up opportunities for devices that suffer from mechanical deficiencies to reach higher accuracies, and it also describes a starting point to completely eliminate the need for mechanical alignment components, flexible couplings and precision ball bearings. The aim of this thesis is to develop a method that will correct the errors that are caused by radial run-out of the shaft of the encoders to improve the accuracy and precision.

The correction methods in this thesis attempt to correct the errors caused by radial run-out utilizing the radial variation in the field. The correction algorithms were developed to estimate the run-out direction and correct the position error with two additional sensors. The main idea was to collect calibration datasets with known run-out configurations and utilize these datasets to estimate the run-out direction and apply correction for

the angular position accordingly. The algorithm was developed considering the following real-time requirements of the encoder system;

1. Exhibit deterministic behavior to guarantee the response in a timely manner
2. Straightforward enough to run on an embedded system

We have built necessary tools to simulate encoder signals under the influence of run-out and tested the algorithm for different field types and different run-out configurations. Simulation, position calculation and error correction algorithms were implemented in C programming language and compiled as Dynamic Link Library (DLL) functions. A GUI was designed in LabVIEW to host the DLL functions, to adjust function parameters and to visualize the results.

The chapters are organized to provide necessary and sufficient information in the following order: Chapter 2 introduces the fundamentals of rotary encoders, then explores angular position extraction procedure for absolute rotary encoders. Moreover, it discusses mechanical problems and provides a survey on error correction methods found in the literature. Chapter 3 constructs the basis of the work by providing information on field generation, sensor placement and effects of run-out on the measurements. Chapter 4 describes the correction method and summarizes the development process and how the algorithm matured to solve problems. Chapter 5 gives overall results obtained for each milestone and shows accomplishments and weaknesses of the algorithm on different case scenarios. Chapter 6 concludes the study and discusses the future work.

CHAPTER 2

ANGULAR POSITION MEASUREMENT

2.1. Rotary Encoders

The task of a rotary encoder is to produce reliable angular position information. A typical rotary encoder consists of a shaft, a PCB and a cover as illustrated in Figure 2.1. The shaft is equipped with a distinct pattern that can be used to resolve the position information inside the encoder and in the other end it is used to connect mechanical input to the device. The PCB consists of sensors to work in cooperation with the shaft pattern along with related electronics and data acquisition components. Lastly, the cover protects the PCB and the pattern on the shaft from the outer effects, such as dust, dirt or interfering stimuli to sensors.

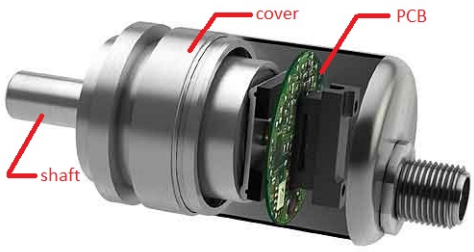


Figure 2.1. Major components of a rotary encoder

Rotary encoders can be categorized by the sensor technology or the physical phenomenon they utilize to accomplish their task. Currently available devices are capacitive, magnetic and optical encoders. There is also a family of rotary encoders that rely on mechanical contact using resistance or conductance as a feature, thus they can grouped based on their contact type and then the sensor technology as shown in Figure 2.2.

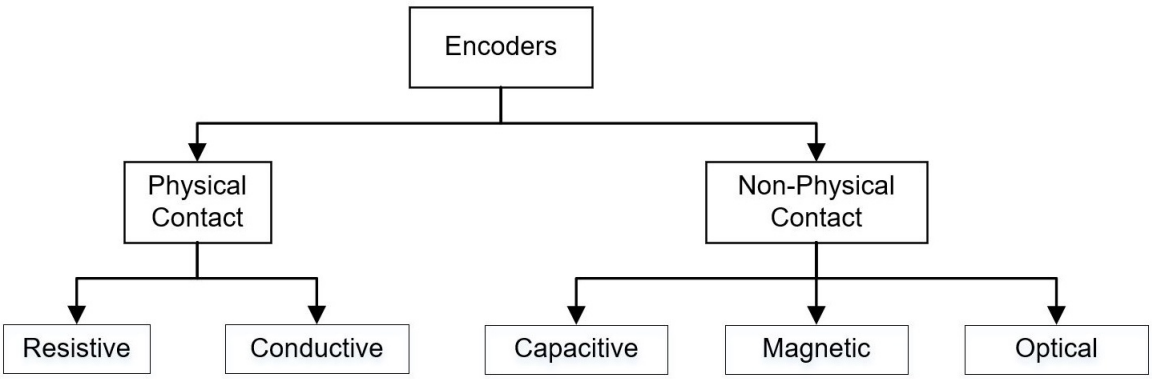


Figure 2.2. Encoder types categorized by the sensor technology

2.1.1. Contact Encoders

Contact type encoders have to maintain contact with the rotating shaft to resolve the angular position. They both utilize a sliding contact, which is called wiper, to track the rotation of the shaft by utilizing either analog or digial signals. Although their operation mechanism is very simple, they are hard to machine and assemble. These former devices are mostly abandoned, at least for quantitative applications, due to several reasons such as low resolution, compromised repeatability, hysteresis and wear.

2.1.1.1. Resistive Encoders

Resistive encoders are devices that utilize Ohm's law to determine the angular position of the mechanical input. The resistive encoder is commonly referred as a potentiometer. One of the terminals of the potentiometer tracks the mechanical input by changing the location of the sliding contact on the resistive track between the other terminals where a constant voltage potential is applied. The voltage potential on the terminal that tracks the mechanical input then changes in accordance with the linear or rotary position of the input.

There are major problems with potentiometers that are limiting their usage as rotary encoders in quantitative applications. The most important problem is the fact that most potentiometers cannot span the full 360 degrees for a rotary encoder application. Three terminal structure requires a gap between the two terminals where the voltage potential is applied. Without this physical gap, the voltage potential on the track can not be maintained without a short circuit. To overcome this problem, various methods are de-

scribed, such as using multiple tracks to compensate for dead bands (Karg, 1960) or using a spring-loaded wiper on an exponentially shaped track (Mayer, 1960). These methods have their flaws and none of them was good enough to survive the evolution of encoders. The exponential track method for example, was only meant to turn in one direction as the contact can not pass over the end of the spiral in the reverse direction. Moreover, the track shape introduced a non-linear behavior due to its exponential structure which is the reason why these devices are called exponential potentiometers. Despite these improvements in compensating for the dead band, there are other limiting factors such as low precision, compromised repeatability, and susceptibility to mechanical stress. Most importantly, the contact element wears out after a certain amount of time.

2.1.1.2. Conductive Encoders

Conductive encoders consist of a series of alternating conductive strips along with insulators that will be in contact with a wiper or a series of wipers to create rectangular pulses encoding the rotary position information. These types of encoders produce a digital output either indicating the absolute rotary position information or relative rotary position information incrementally. If the wiper makes a contact with conductive strips it shorts the circuit and a voltage indicating high logic level can be read from that strip. If the wiper makes a contact with the insulator, the circuit is open and a voltage indicating low logic level can be read from that strip. In absolute encoders, multiple wipers, which are placed in parallel, are committed to track different concentric circles. Here, the output of every wiper represents a bit in a sequential binary series. Following equation shows the achievable resolution when N number of bits are represented in concentric circles.

$$Resolution = \frac{360^\circ}{2^N} \tag{2.1}$$

Thus, the resolution of these devices is limited to the pattern segmentation on the disk and the size of the wiper. Figure 2.3 shows the tracks on a rotary encoder that is equipped with the common binary representation of the decimal sequence. The outermost track represents the least significant bit and the innermost track represents the most significant bit.

Some of the problems of potentiometers persist in the conductive encoders as well. The structure is susceptible to mechanical stress like vibration and shock because the wiper contact must be maintained all the time. The parts of the device must be carefully

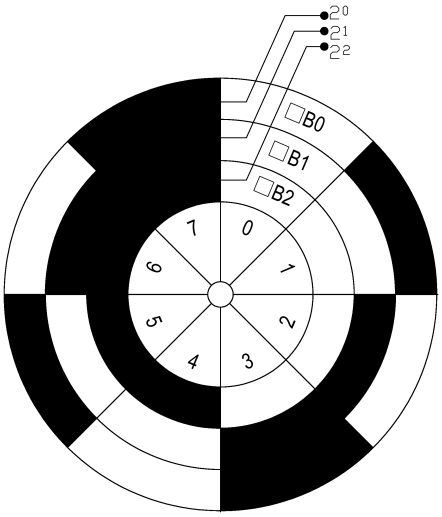


Figure 2.3. Shaft extension of a simple 3 bit absolute encoder

designed, machined and assembled to avoid mechanical failures. Moreover, the wiper wears out after a certain number of revolutions.

2.1.2. Non-Contact Encoders

Non-contact encoders are the current state of the art devices. The notion of contact should not be confused with contactless encoders. The encoder itself still has to be coupled with mechanical input even though the sensing mechanism inside the encoder do not rely on any form of physical contact with the encoder shaft.

2.1.2.1. Capacitive Encoders

Capacitive Encoders rely on the fact that two metal surfaces facing each other create a capacitance depending on the surface area, the dielectric constant of the medium and the distance between the surfaces. In its most fundamental form, the capacitive encoder employs the linear relationship between the capacitance and the surface area, if the dielectric coefficient and the distance are kept constant. The change in the capacitance is detected by observing frequency and phase shifts of sinusoidals (Baxter, 2000).

The linear capacitive encoders have been used in ordinary digital calipers for more than two decades and they can easily reach 0.01mm resolution. Rotary encoders employing the same position detection scheme also gave promising results. In a recent paper

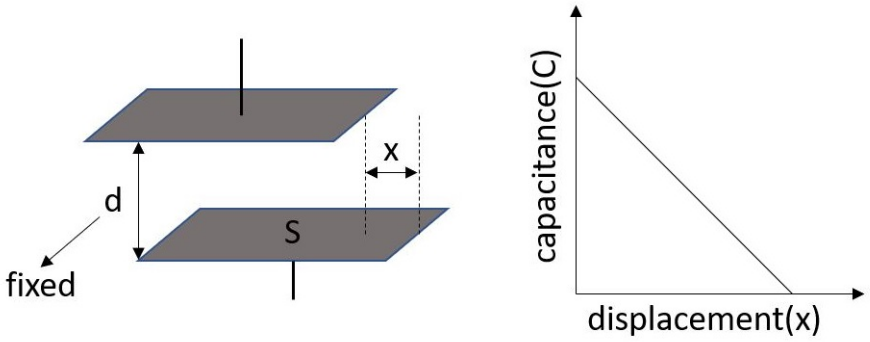
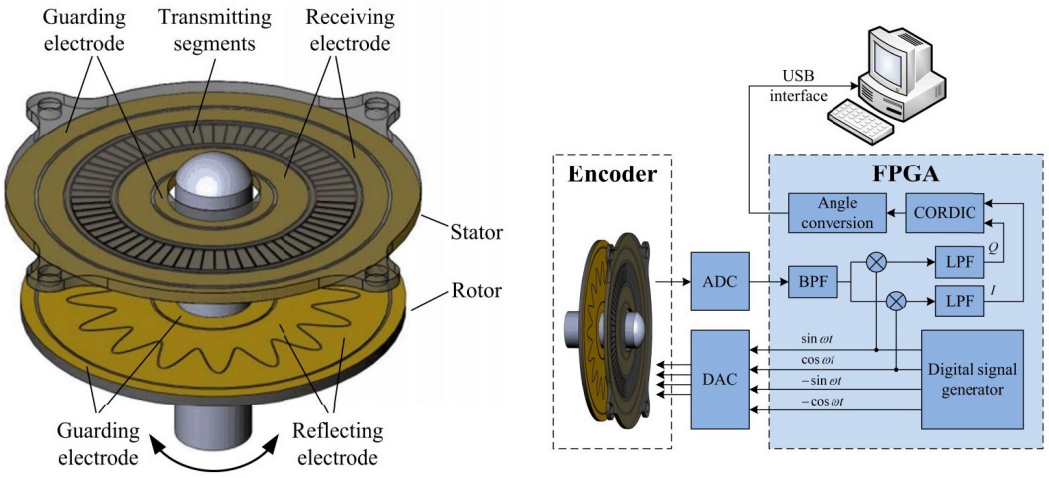


Figure 2.4. Capacitance between two parallel plates as a function of displacement

(Zheng et al., 2015), the design idea shown in Figure 2.5 with two electrode system is explored. A periodic (in this case sinusoidal) pattern is etched into the reflector part which is coupled to the mechanical input. A variable capacitance is formed between the transmitting segments of the stator and the corresponding section of the sinusoidal pattern from the reflecting electrode. The sinusoidal pattern encodes angular position information by modulating the signal in terms of phase and frequency. A high-speed digital processing system interfaces with the encoder to acquire data and to perform demodulation operation to the acquired signal to obtain angular position information.



(a) Structure of the shaft extension

(b) Encoder interface

Figure 2.5. Capacitive encoder design (Source: Zheng et al. (2015))

Although they are the least popular choice among the non-contact encoders, there have been promising improvements on capacitive encoders recently. Latest capacitive en-

coders from reliable manufacturers can make measurements with $\pm 0.2^{\circ}$ accuracy up to a speed of 8000 rpm (CUI Inc., 2017). However, these improvements yet to be enough to compete with magnetic and optical counterparts dominating the market for encoders. Capacitive encoders are susceptible to electrical interference. Even if the rotor and stator parts are equipped with guarding electrodes the mechanism still suffers from crosstalk and exhibits offset errors. Moreover, unlike other encoders, the entire encoder structure is involved in the measurement, making the process more complex. Thus, data acquisition and signal processing get complicated, requiring dedicated high-speed electronics (FPGA/ASIC).

2.1.2.2. Magnetic Encoders

Magnetic encoders are commonly employed for heavy-duty tasks due to their high reliability. They are resilient against environmental (dust, dirt) and mechanical effects (shock, vibration) and they can work for years without any form of maintenance. They provide excellent performance in harsh environments and they are low in cost. On the other hand, these devices exhibit lower resolution compared to optical instruments and they are susceptible to electromagnetic interference (EMI).

Magnetic encoders exploit magnetic field properties to determine the position of the mechanical input. Three fundamental magnetic encoding methods described in the literature are variable reluctance, magnetoresistance and hall effect.

The earliest method was to sense the perturbation in the magnetic field that is caused by a toothed wheel that is constructed from a ferromagnetic material. The sensor used to pick up the signal is constructed by placing a permanent magnet in a coil and often attaching it to a ferromagnetic object. This structure is called as variable reluctance sensor, or in short VR sensor. These sensors are passive sensors which means there is no need to supply power for their operation. The main drawback is the signal dropouts at low speeds due to the limited rate of change in the magnetic flux when the wheel spins slowly. Furthermore, the resolution of the system is relatively low, since the mechanical limits dictate the teeth size of the wheel and the number of teeth placed around the wheel determines the upper limit of the resolution.

Certain types of ferrous materials show resistivity changes under applied magnetic field. These materials can be used to sense the magnetic field changes around a wheel with magnetized poles. In this method, a voltage divider is formed by a magnetically sensitive material that has lower resistance when a magnetic field is exerted on the material. The

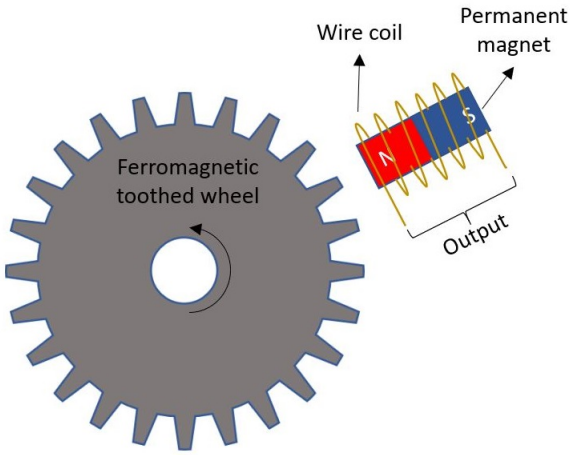


Figure 2.6. Variable reluctance encoder

resistance rapidly turns back to normal when the field is retracted. Permalloys are commonly employed as magnetoresistive materials due to high sensitivity to magnetic field and being robust against mechanical stress.

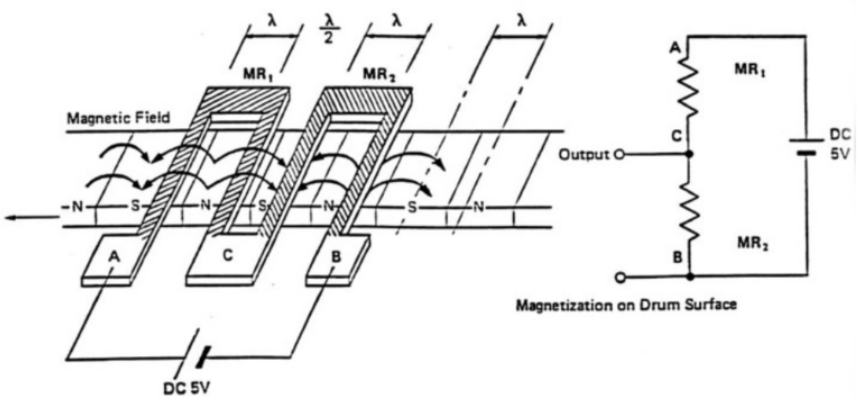


Figure 2.7. Magnetoresistive sensor strip (Source: Ireland (2010))

Hall effect sensors rely on Lorentz force, which is defined as the effect of the electric and magnetic fields on a single charge. In this method, the shaft is coupled axially with a permanent magnet. The stationary sensor is simply a semiconductor material (p doped silicon) with a constant voltage applied to it. As the current passes over the semiconductor material, the magnetic field exerts a force over the electrons deflecting them from their ideal route. The electrons and holes are attracted and repelled depending on the direction of the magnetic field, and a voltage measurement from the semiconductor is

taken in the direction perpendicular to the current flow. This voltage changes depending on the magnetic field direction as the holes and electrons accumulate in opposite directions during the shaft rotation.

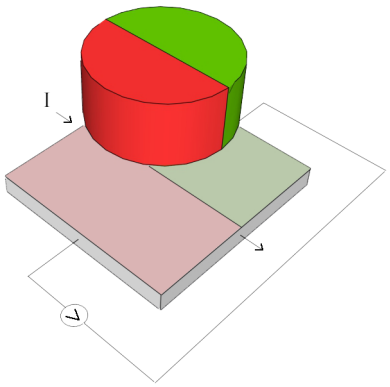


Figure 2.8. Hall effect sensor: measured voltage depends on the applied magnetic field

2.1.2.3. Optical Encoders

Optical encoders are state of the art devices that produce finest resolution with highest accuracy along with fast dynamic response. The main drawback of these devices is the susceptibility to dust and dirt. Good sealing techniques compensate for this drawback.

Optical encoders provide excellent performance where high resolution, accuracy and fast dynamic response are required. The simplest method to build an optical encoder is to couple the rotating shaft with a disk that is equipped with a grating pattern on it. Placing a light source and a photodetector on the opposite sides of the disk will provide an incremental optical encoder. The light that is emitted by the light source is transmitted through the periodic slits of the grating. In this case, light that is incident on the sensor is a function of how much the opaque sections cover the light source. The photodetector will produce a voltage at high logic level when the transparent section passes between the light source and the photodetector and it will produce a voltage at low logic level when the opaque section passes. A periodic wave is generated as the disk rotates in between. A threshold value is determined and a comparator is used to create pulses that are equal distances apart. At the receiver end, a counter is used to count the incoming pulses to determine the change in the rotary position of the shaft. This solution lacks the sense of

direction though. To provide a sense of direction, another photodetector that will produce an output 90 degree out of phase with the primary photodetector needs to be placed. Then the direction can be extracted by determining which one of the photodetector signals is lagging.

An improvement to the system to achieve fine resolutions is covering the sensors with the same grating pattern as the rotating wheel. The light received at the sensor is maximized when two lines of these gratings coincide. This adjustment achieves greater dynamic range in the output signal by limiting the receiving aperture of the photoactive region of the photodetector. Otherwise, the photodetector is always illuminated to a certain degree which is effectively reducing the dynamic range of the sensor outputs.

2.2. Operation Modes of Rotary Encoders

Encoders are classified further based on whether the output is produced according to a fixed reference point or not. The two types introduced based on this classification are incremental and absolute.

2.2.1. Incremental Encoders

Incremental encoders produce relative position information with respect to a reference point that changes every time the device is powered up. The shaft of the encoder is equipped with a uniform periodic pattern (grating pattern on optical encoders or magnetized poles around a drum in magnetic encoders) that is intended to work in conjunction with the sensors of the encoder. The sensors generate a pulse every time they interact with the periodic elements of the pattern. The number of pulses per revolution of the shaft is determined by the number of periodic patterns on the shaft. To determine the direction of motion, incremental encoder employs at least two sensor channels that would provide two similar signals that are 90 degree out of phase. The direction of the motion can be extracted by observing which channel is lagging behind.

2.2.2. Absolute Encoders

Absolute encoders produce absolute position information according to a fixed reference point. To be qualified as an absolute device, an encoder must show the correct

angular position information right from the start up. The sensors of absolute encoder can generate either digital or analog waveforms to express the angular position information. The shaft of the digital output encoder is equipped with concentric lines of patterns that follows a coding scheme. Linearly displaced multiple sensors produce a binary word by tracking these lines. Incorporated coding scheme usually follows Gray Code instead of binary numbers to avoid misinterpretations caused by erroneous readings. Only one bit toggles between consecutive readings of Gray Code whereas multiple bits toggle in binary number sequence. By this way Gray Code inherently restricts the error, caused by poor reaction times, in the range of $\pm$1LSB region. Analog encoders produce two waveforms, ideally a sine and a cosine. These two signals take a unique value for every angular position that can be mapped back with appropriate methods.

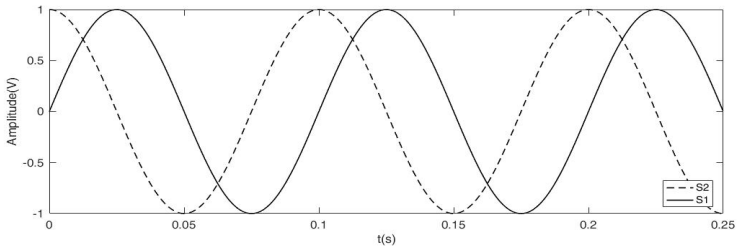
2.3. Analog Sensor Waveform

Incremental and absolute encoders of any sensor type can produce analog signals that can be traced to determine angular position to a degree. In incremental encoders, such as magnetoresistive encoders and grating based optical encoders, analog signals can be utilized to extract fine resolution while the countable steps provide rough position. Moreover, analog signals can be exploited to decrease the number of elements in the pattern and thus reduce the cost of the incremental encoders (Staebler, 1998). In analog absolute encoders, such as hall effect encoders, analog signals are the only resources that are being used to determine the angular position.

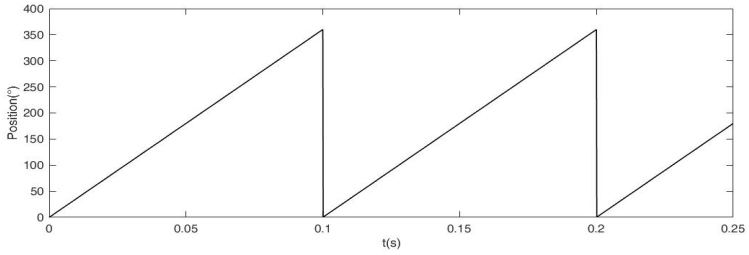
Mentioned analog signals are two sinusoidals that have a phase shift of 90 degrees. Since the signals have a phase shift of 90 degrees between them, they are commonly called quadrature signals. In incremental encoders, these signals are periodically produced for each step of the encoder while in analog absolute encoders signals are produced once for every revolution of the shaft. Figure 2.9 shows the sinusoidals received from two sensors for each revolution of the shaft.

There are two main benefits on using quadrature signals;

1. The rotation direction can directly be extracted by looking at which signal is lagging
2. The signals can be used interchangeably to determine location more accurately by making use of the sensor output with higher rate of change as a function of angular position.



(a) Quadrature signals obtained from sensors



(b) Angular position with respect to quadrature signals

Figure 2.9. Calculation of angular position based on quadrature signals

2.4. Extraction of Angular Position Information in Analog Absolute Encoders

Unlike an incremental encoder, where the angular position is incremented by one step every time a pulse is detected, an analog encoder has to resolve the analog signals to determine angular position. The quadrature signals of an analog encoder have a phase shift of 90 degrees which means one of them is sine and the other one is cosine function as a function of angular position. This means that dividing these signals will result in tangent of the angular position, thus taking the inverse tangent at any instant yields the angular position.

$$\theta = \arctan\left(\frac{S1}{S2}\right) \tag{2.2}$$

In implementation, the arctangent calculation can not be performed by using a single argument function because the sign of the signals are lost after division. As a consequence of the sign loss, the single argument function can only produce output in the range $(-\pi/2, \pi/2)$. On the other hand two argument function like atan2 in C can produce output in full range $(-\pi, \pi)$ by making use of signs of both arguments.

$$atan2(y,x) = \begin{cases} \arctan\left(\frac{y}{x}\right) & \text{if } x > 0 \\ \arctan\left(\frac{y}{x}\right) + \pi & \text{if } x < 0 \text{ and } y \geq 0 \\ \arctan\left(\frac{y}{x}\right) - \pi & \text{if } x < 0 \text{ and } y < 0 \\ +\frac{\pi}{2} & \text{if } x = 0 \text{ and } y > 0 \\ -\frac{\pi}{2} & \text{if } x = 0 \text{ and } y < 0 \\ \text{undefined} & \text{if } x = 0 \text{ and } y = 0 \end{cases} \tag{2.3}$$

The cost of arctangent computation poses another issue due to calculation complexity. Calculating arctangent by using a series expansion is demanding and even a fifth order Taylor series expansion would be inaccurate (Burke et al., 2000). Lookup tables offer a faster and computationally simpler method at the expense of memory. A lookup table (LUT) can be constructed by collecting calibration data. Piecewise polynomial function coefficients can be found using the calibration data and these coefficients can be stored for each quadrant. Another benefit of the LUT method is that the systematic deterioration in the analog signals due to system imperfections can be compensated because LUTs allow the calibration data to be stored as an arbitrary function. This is important because in reality, sensor outputs are commonly out of shape and they need preconditioning. Even with preconditioning, ideal sinusoidal outputs can rarely be obtained. To avoid these complications, it is best to make use of lookup tables or another method that utilizes calibration data. The drawback of using LUT is the memory requirement. CORDIC method (Volder, 1959) offers another option that uses less memory space.

2.5. Mechanical Interface

The installation and quality of the mechanical parts of the encoder is just as important as the internal mechanics of electrical measurement. Commonly, an encoder is seated inside a housing that is fixed to the body of the motor with screws or bolts. Selection of the components, correct mounting and precise alignment affect the accuracy and lifetime of the encoder (Mahn, 2010). Like most measurement systems, rotary encoders have to establish and maintain physical contact to be able to measure. Indeed, this contact is one of the most important aspects of the encoder since it provides the connection to follow mechanical motion. Encoders establish the connection to mechanical motion via a shaft that is coupled to the motor with a coupling component and secured to the encoder hous-

ing with a ball bearing. Hence, two important components of the mechanical interface is the coupling and ball bearing. The quality of these components as well as how well the encoder is mounted affects the accuracy of the angular position measurement.

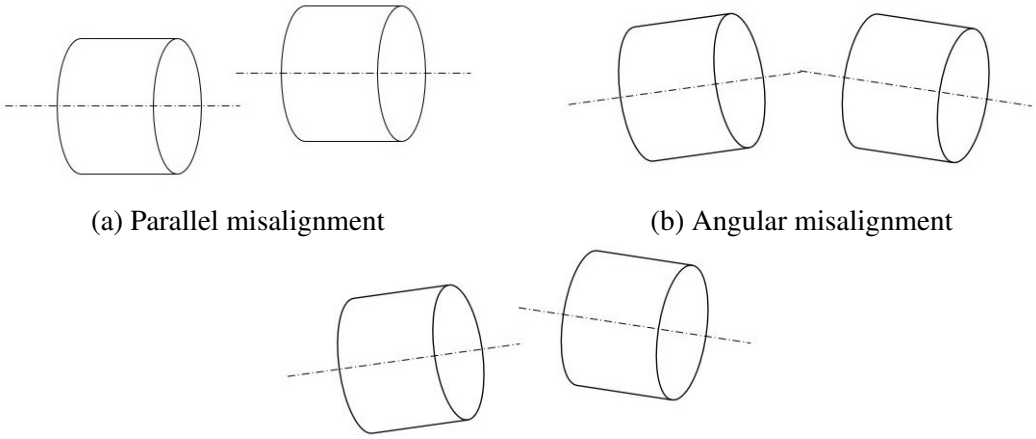
The deviation of the shaft rotation axis from the central axis of the encoder is examined under shaft misalignment. There are different types of shaft misalignment caused by several reasons such as installation issues, component failure or shaft deflection. Different coupling devices are developed to accommodate misalignment of different types and ball-bearings are designed to limit shaft end play in different directions.

2.6. Shaft Misalignment

The mechanical motion is transmitted to encoder through a shaft. The alignment of the shaft is a concern due to following reasons;

- Misalignment results in early failure of the systems by damaging internal mechanical components.
- Extra stress on interface mechanism produces heat and vibration and wears the components out prematurely.
- Maintenance period of mechanical components is shorter when excessive wear is expected due to misalignment.
- Mechanical misalignment introduces measurement errors before any other error source in electronic components or signal processing.

Correct alignment is essential for the devices to work properly. Correct alignment implies that the rotation axes of two shafts being colinear. There are mainly two types of misalignments, parallel and angular, but also the combination of these two types is considered as a different case. Parallel misalignment occurs when two rotation axes are parallel to each other but have an offset in vertical or horizontal axis of the plane that is orthogonal to the rotation axis. Angular misalignment occurs when two rotation axes are making an angle with each other but the connection point is at the same level. When both of these issues are present, it is called combined parallel and angular misalignment. Three types of misalignments are illustrated in Figure 2.10.



(a) Parallel misalignment

(b) Angular misalignment

(c) Combined misalignment

Figure 2.10. Three types of shaft misalignments

2.7. Shaft Run-out

Shaft run-out is one of the causes of shaft misalignment. Run-out of the shaft indicates that there is a mechanical imperfection in the system that is causing shaft to move along the rotation axis or rotate eccentrically. Run-out errors are inevitable because there will always be alignment issues up to a degree. This dynamic error is reduced by off-line mechanical measurements and follow up alignment procedures.

There are two types of run-out, axial and radial. Axial run-out occurs when the shaft moves along the rotation axis. This type of run-out may originate from low gripping performance or high axial loads.

Radial run-out occurs when the shaft moves in the direction perpendicular to the rotation axis. Radial run-out is generally thought as more serious condition as the shaft starts to rotate about a secondary axis and causes vibration thus heat and damage. Radial run-out is commonly associated with ball-bearing precision.

2.8. Mechanical Components for Alignment Issues

Two mechanical components are mainly utilized for misalignment issues, flexible couplings and ball bearings. Flexible couplings accommodate misalignment while transmitting the motion to the encoder while ball bearings limit end play of the shaft.

2.8.1. Coupling Mechanisms

Couplings are mechanical devices that join shaft ends together in order to transfer mechanical motion. There are three main duties of coupling hardware;

1. Transmission of rotary motion
2. Accommodation of shaft misalignment
3. Compensation of shaft movement

There are mainly two types of couplings, rigid and flexible. Rigid couplings are relatively old technology and they do not find use unless the shaft movement is slow and misalignment is not expected. Flexible couplings provide a more sophisticated solution due to their capability of accommodating misalignments.

Rigid couplings do not exhibit any flexibility and they are not able to compensate for misalignment errors. If two devices are coupled with a rigid coupling, misalignment force will directly be reflected on the other mechanical components, mostly on other alignment components, such as bearings. The force acting on the shaft will deflect the shaft from its original rotation axis, it will wear the ball-bearing out causing lubricant spill out, inducing heat and thus driving the bearing to failure. Flexible couplings accommodate for slight misalignments, limited by a few degrees of angular misalignment and a small fraction of a millimeter of parallel misalignment.

The flexible couplings that are worth mentioning are beam couplings, jaw couplings and disc couplings. Disc couplings and jaw couplings are made of multiple parts whereas beam couplings are made of a single body. Disc couplings are made of a series of thin metallic discs that are connected with bolts. This structure allows for high torsional stiffness while accommodating for parallel and angular misalignments. Jaw couplings are made of three components, two jaw like parts and one elastic part that fits in between them. Jaw couplings are good at damping vibrations. Beam couplings are single body components which cut in a helical shape that could bend to accommodate misalignments.

While couplings transmit the motion from the motor to the encoder, they also transmit secondary motions caused by imbalances of the motor. Furthermore, any eccentricity or alignment problem on the shaft induces unwanted forces on the encoder. If these forces are left untreated they cause vibrations and one of the components, shaft, ball bearing, coupling or the internal electronics, may end up failing. Also, the forces acting on the encoder wear out the hardware prematurely. Flexible couplings are mechanically

(a) Disc coupling

(b) Jaw coupling

(c) Beam coupling

Figure 2.11. Different types of couplings

good at solving misalignment and play issues, on the other hand they may introduce lag in the rotation of lateral shaft due to low torsional stiffness or slight errors due to backlash (Ellin and Dolsak, 2008). These two issues compromise the accuracy of the angular measurements. Moreover, couplings do not eradicate the problem of misalignment completely, since they have limits specified by the manufacturers. Unless properly aligned, the only improvement the coupling will introduce to the system would be prevention of coupling failure instead of other component failure.

2.8.2. Ball Bearings

Although there are various forms of ball bearings that evolved to deal with different situations, a simple ball bearing is composed of two braces and balls inside them sometimes employing a metal cage around the balls for uniform separation. Two main types that can be mentioned are radial bearings and thrust bearings. Radial bearings are designed to deal with the radial forces perpendicular to the shaft axis, whereas thrust bearings are designed to deal with the axial forces parallel to the shaft axis. The main duty of the ball bearing is to minimize friction while securing the shaft in place in the presence of

secondary forces. Generally, the shaft is seated inside the inner brace while keeping outer brace steady so the that balls between braces move consistently.

Ball bearings are excellent at fastening the shaft in place, however they are susceptible to wear issues. They need proper maintenance, such as lubrication or shaft calibration, to preserve their precision. Furthermore, premature failure of ball-bearings are common in industry due to excessive force, overheating, excessive vibration, fatigue, contamination, lubricant failure, corrosion or misalignment.

Ball bearings are manufactured according to standards that are defined by organizations, such as American National Standards Institute (ANSI), International Standards Organization (ISO) or German Institute for Standardization (DIN). Equivalents of different precision grades defined by these institutions as well as the total run-out allowed for each grade is given in Table 2.1. These values are defined for 10mm inner ring bore diameter and 18mm outer ring bore diameter.

Table 2.1. Run-out specifications with according to various manufacturing standards

ANSI	**ISO**	**DIN**	**Inner ring run-out** (μm)	**Outer ring run-out** (μm)
ABEC 1	Class Normal	P0	10	15
ABEC 3	Class 6	P6	7	9
ABEC 5	Class 5	P5	4	6
ABEC 7	Class 4	P4	2.5	4
ABEC 9	Class 2	P2	1.5	2,5

These specifications associate run-out with only bore diameters of inner and outer rings. In reality, other sources also contribute to run-out such as internal clearance and ball accuracy. According to specifications, a normal class bearing can have a radial run-out up to $25\mu m$ in total where a class 2 bearing have only $4\mu m$. The price increases with the precision of the ball bearing. Other important aspects that affect the bearing precision over time are not controlled by the standards such as surface finish, cage type and lubrication.

2.9. Overview of Error Correction Methods for Rotary Encoders

Angular position errors in encoders result from a combination of electrical and mechanical imperfections. The effects of encoder errors on the accuracy of machine is studied to understand the implications of encoder errors on machine performance (Ale-

jandre and Artes, 2004a). The errors resulting from mechanical deficiencies are classified under three major topics;

1. Static errors due to displacement and deformation
2. Errors due to thermal expansion
3. Dynamic errors due to mechanical vibration and play

These errors are caused by a myriad of reasons such as installation issues, quality of the materials, and shaft misalignment. Most of the work on identifying these issues include a linear encoder studied under an isolated source of error. Thermal coefficients (Alejandre and Artes, 2004b) and non-linear thermal effects (Alejandre and Artes, 2006) have been investigated and correction methods are suggested. The effects of vibration have been analyzed to quantify errors in the presence of dynamic motion and deteriorations in signals are shown in Lissajous figures (Alejandre and Artés, 2007). Furthermore, vibration under different mounting conditions (Lopez et al., 2011) and for different rotary encoders with different installations (Lopez et al., 2012) are examined.

Errors can also be caused by electrical components utilized in the encoder. Sensor sensitivity differences, inequality in signal amplification, uneven illumination can be given as examples that deteriorate the sinusoidal signals (Matsuzoe et al., 2002). The errors from either mechanical or electrical sources, can be widely classified as static or dynamic depending on their behavior. The correction methods in the literature can be examined under these two major topics.

2.9.1. Correction of Static Errors

Static errors imply that the error is inherent in the system and does not change over time. These errors are resulting from imbalances and constant forces in the system and they are systematic in nature. Static errors are not expected to change over time, therefore most of the work in literature rely on off-line measurements to identify deteriorations in the quadrature signals and apply on-line corrections accordingly.

The early work in the field consists of correction of sinusoidal pairs of interferometers that measure path length. According to these studies, error in the system manifests itself in phase shifts, amplitude changes and offsets. The deviation of the Lissajous figure from the ideal circle to ellipse is calculated with least squares fitting method to identify the amount of errors (Heydemann, 1981). The experimental data is collected off-line and

corrections applied on-line. Later, Heydemann method further modified to obtain nanometric accuracy in path length determination (Birch, 1990). Since rotary encoders provide similar quadrature sinusoidals, this method also modified and applied for rotary encoders to solve inherent errors which is called metrological errors in the paper (Sanchez-Brea and Morlanes, 2008). In this paper, linear series expansions are utilized to obtain expressions for errors.

A promising work focuses on deriving higher order sinusoids from existing quadrature signals and employing them to generate binary pulses to be decoded by servo controllers. The method involves a LUT which is constructed off-line where the errors in the sinusoidals are compensated beforehand (Tan et al., 2002).

The bottom line is that these methods analyze systematic deviations with off-line measurements and apply corrections on-line. They are excellent at correcting stationary errors, but they are susceptible to changes that occur after the characterization of error sources.

2.9.2. Correction of Dynamic Errors

Dynamic errors are commonly result from mechanical motion. Loose installation of components, mechanical motion in the base of the system or shaft deflection affect the measurements and result in error. These error sources may be deterministic such as a mechanical vibration at a constant frequency that appears on the measurements as a sinusoid at the same frequency. The errors may also be random, such as the run-out error that results from dynamic movement of the shaft where there is no prior knowledge about direction and amount.

Using Kalman filter is an early idea that is adopted to correct both deterministic and random errors (Yang et al., 2002), but this work is not meant to work for quadrature signals. With a good model of error dynamics, adaptation for quadrature signals might be promising. Neural networks are also employed in recent studies that claims 90% improvement in 16-bit resolvers (Dhar et al., 2009). The former study on using artificial neural networks (ANN) for encoders is very interesting where a two stage radial basis function (RBF) ANN is employed to identify and correct errors (Tan and Tang, 2005). The first stage of RBF is responsible for detection of anomalies in the signal and the second stage apply interpolation accordingly. The drawback of the ANN method is that it requires training data and adaptation takes time. Other methods are also developed such as using quadrature all digital phase locked loops (QADPLL) (Le et al., 2008) or Lissajous figure

fitting under sine vibration with LUTs (Lopez and Artes, 2012). Lately, gradient descent algorithm is applied to the problem to iteratively find the optimum correction parameters (Lara and Chandra, 2014). Observers are also employed to create a feedback loop to estimate error and compensate it, but the algorithm works for repetitive errors and it introduces lag (Albrecht et al., 2017) in other situations.

Overall, the methods developed until now are good at compensating systematic and repetitive errors. They can only provide a delayed response, or they completely fail when the dynamic error is not deterministic. Some dynamic errors such as run-out of the shaft cause sudden changes in the signal. These methods require time to adapt such changes, thus there needs to be a technique that employs only instantaneous data to correct such errors.

CHAPTER 3

FIELD GENERATION AND EFFECTS OF RUN-OUT

The objective of this study is to correct angular position errors in rotary encoders caused by radial run-out of the shaft. To achieve this objective, a heuristic algorithm was developed. The algorithm relies on additional sensor data to estimate the shaft run-out direction and applies angular correction proportional to the shaft offset from the center.

In order to test the proposed algorithm, a flexible simulation environment was created considering run-out scenarios of different amounts in different directions. A graphical user interface (GUI) was developed in the front end using LabVIEW software for various reasons such as adjusting simulation conditions, applying correction algorithm variations and visualizing the results. In the back-end, a collection of functions was written in C programming language, and compiled as a dynamic-link library (DLL) to work in conjunction with the GUI to perform the computational work. Mathematical expressions governing the field generation and algorithms to determine angular position were implemented in this DLL to verify the performance of the methods. This chapter is dedicated to explore the former objective of the DLL, specifically the generation of the sensor field that incorporates two dimensional intensity variations with the sensor errors resulting from radial run-out conditions. The following objectives are going to be fulfilled throughout this chapter;

1. Introduction of mathematical expressions governing the field generation
2. Explanation of how sensor outputs are affected by the misalignment of the shaft
3. Demonstration of the collected data for centered and misaligned shafts

3.1. Field Properties and Field Generation

The shaft of the rotary encoder is responsible for forming a distinguishable field pattern. In this sense, two or more sensors pick up the information from this field and a processor interprets this sensory data to obtain angular position information. In this thesis, it was assumed that the shaft is rotating about z axis and it reflects a field to xy plane as illustrated in Figure 3.1. The field was located on the plane underneath and around the

shaft. The field is stationary when the shaft is not rotating and it has a circular distribution with a distinguishable angular pattern. Main features of the field can be listed as;

1. The field follows an angular variation function that varies as a function of the angular position.
2. The field intensity is distributed over a circular region and it is concentrated (it has the highest intensity) in the middle of this region.
3. The angular variation and radial variation functions are independent of each other.

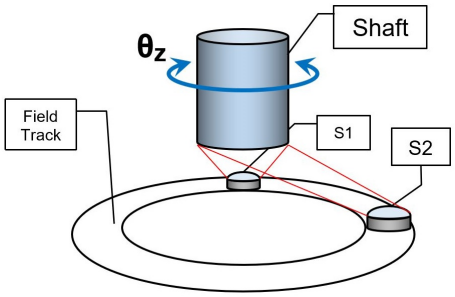


Figure 3.1. Field location

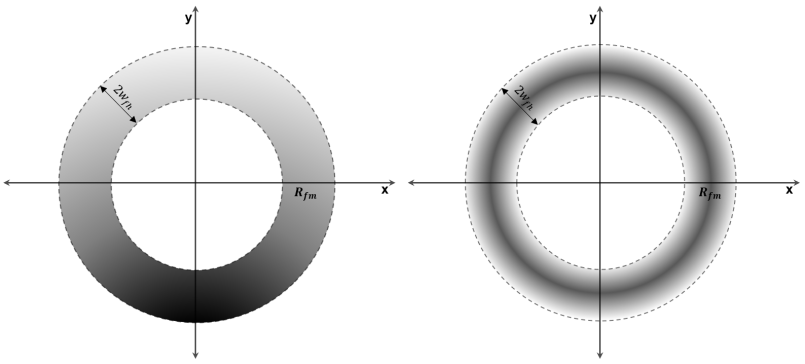
Field intensity function is a product of the radial and angular field variation functions. It is convenient to use polar coordinate system due to the shape of the field intensity being a function of the angle and distance from the shaft. The field intensity is a function of r and θ as follows;

$$I_f(r,\theta) = I_{fR}(r)I_{fA}(\theta) \tag{3.1}$$

where $I_{fA}(\theta)$ is the angular field variation in which the angular position information is encoded, and $I_{fR}(r)$ is the radial field variation over the field ring. The shape of the $I_{fA}(\theta)$ and $I_{fR}(r)$ in cartesian coordinate system are roughly illustrated in Figure 3.2

3.1.1. Angular Field Variation

Angular field variation function was modeled as a sinusoidal where a single revolution of the shaft will complete a single period of the sinusoidal. Bipolar sinusoidal



(a) Angular field variation (b) Radial field variation

Figure 3.2. Arbitrary field intensity graph (black means higher intensity)

function was considered which requires a constant offset as follows;

$$I_{fA}(\theta) = I_{fos} + I_{fAvar} sin(\theta) \tag{3.2}$$

where I_{fAvar} is the amplitude and I_{fos} is the offset of the angular field variation.

3.1.2. Radial Field Variation

Radial field variation function follows a normalized distribution where the maximum value is located in the middle of the ring. Ramp, cosine and Gaussian functions were tested in this thesis. The mathematical expressions of these distributions are given below respectively;

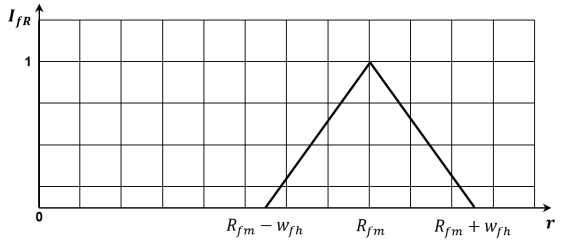
$$I_{fR}(r) = \begin{cases} 1 - \left| \dfrac{r - R_{fm}}{w_{fh}} \right| & \text{if } |r - R_{fm}| \leqslant w_{fh} \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

$$I_{fR}(r) = \begin{cases} cos\left(\dfrac{\pi(r - R_{fm})}{2w_{fh}} \right) & \text{if } |r - R_{fm}| \leqslant w_{fh} \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

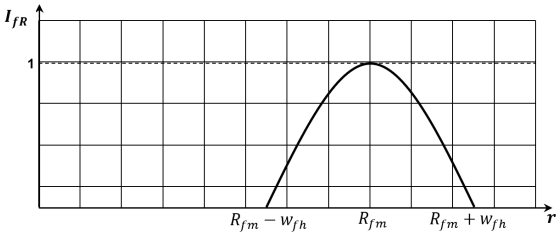
$$I_{fR}(r) = e^{\frac{(-r - R_{fm})^2}{w_{fh}^2}} \tag{3.5}$$

where R_{fm} is the middle circle radius of the field ring, and w_{fh} is the half width of the field ring. Visual representations of different distributions are given in Figure 3.3a, 3.3b

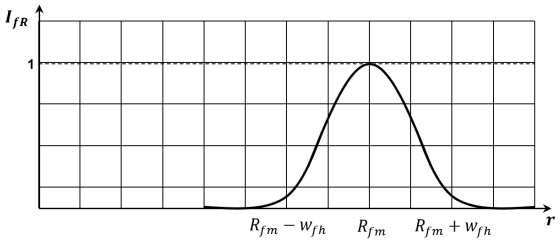
and 3.3c as a function of r. An exemplary field, which is a combination of triangular $I_{fR}(r)$ and sinusoidal $I_{fA}(\theta)$ functions, is depicted in Figure 3.3d in 3D.



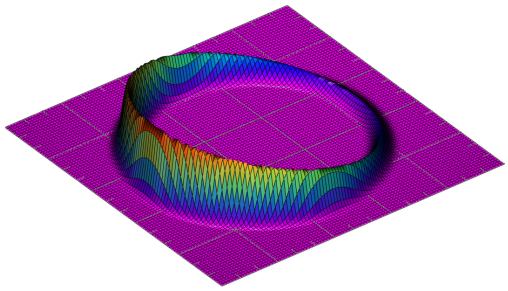
(a) Triangular variation over r



(b) Cosine variation over r



(c) Gaussian variation over r



(d) 3D visualization of an exemplary field

Figure 3.3. Field distributions in 2D and 3D

3.2. Sensor Placement in the Field

Four software sensors, assuming each of them exhibit perfect linear response to field strength, were placed in the field to produce four waveforms with 90 degree phase shift consecutively. The sensors measure the field intensity while the field rotates about z axis. Location of the sensors on the field is tabulated in Table 3.1.

Table 3.1. Default sensor locations

Sensor ID	Loc. in cartesian coordinate system	Loc. in polar coordinate system
S1	$(R_{sns1}, 0)$	$(R_{sns1}, 0)$
S2	$(0, R_{sns2})$	$(R_{sns2}, 90)$
S3	$(-R_{sns3}, 0)$	$(R_{sns3}, 180)$
S4	$(0, -R_{sns4})$	$(R_{sns4}, 270)$

Sensor outputs were calculated by evaluating the intensity function at the corresponding sensor locations considering rotation angle of the shaft as shown below;

$$S(\theta_z, k) = I_f(r, \theta)\Big|_{r=Rsns, \theta=k.90-\theta_z} \tag{3.6}$$

The intensity function yields the sensor data given in Figure 3.4 for one revolution of the correctly centered shaft when $R_{sns1,2,3,4} = 9.5mm$ $R_{fm} = 10mm$, $w_{fh} = 9mm$, $I_{fAvar} = 2V$ and $I_{fos} = 2.5V$.

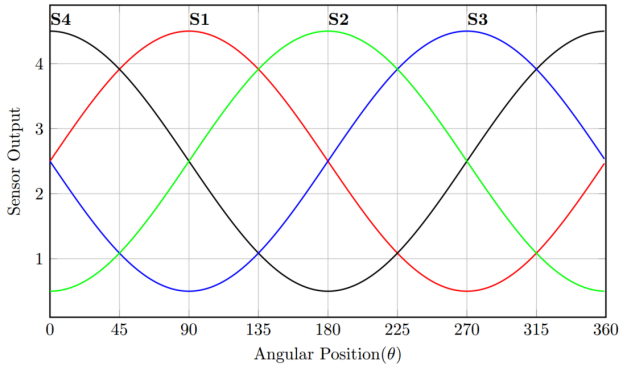


Figure 3.4. Sensor outputs for perfectly centered shaft

3.3. Effect of Run-Out on The Sensor Outputs

Run-out of the shaft causes the field to shift in the direction of the run-out in accordance with the amount of the run-out. Although the shape of the field is preserved, this translational motion leaves the sensors on undesired positions in the field and the sensors pick up erroneous information. It is necessary to correctly model the run-out effect on the field position in order to create an accurate simulation environment for the evaluation of the proposed correction algorithm.

The shift in the shaft position is illustrated in Figure 3.5. The translation variables that are going to be used in geometric transformations are; shaft rotation angle θ_z, shaft run-out distance R_{ro}, and shaft run-out direction ϕ_{ro}.

For vectors in polar coordinate system can not be added, they were transformed into cartesian coordinate system before translation;

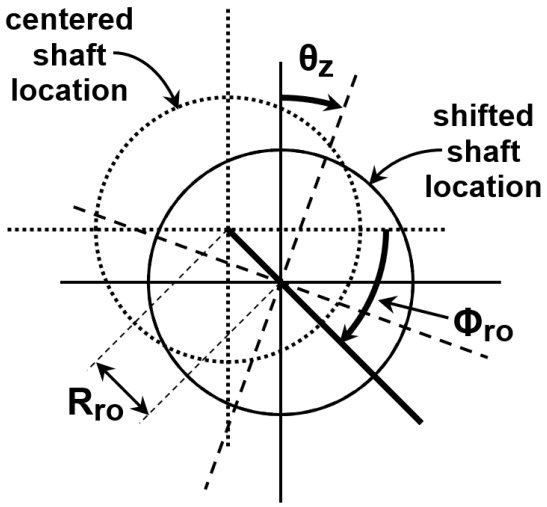


Figure 3.5. Illustration of shaft run-out

$$x_{ro} = R_{ro}cos(\phi_{ro}) \tag{3.7}$$

$$y_{ro} = R_{ro}sin(\phi_{ro}) \tag{3.8}$$

where x_{ro} is the run-out in x direction and y_{ro} is the run-out in y direction.

The translation is a simple geometric transformation in the cartesian coordinate system where the field components shift according to run-out conditions. The transformation is given by;

$$\underbrace{\begin{bmatrix} 1 & 0 & x_{ro} \\ 0 & 1 & y_{ro} \\ 0 & 0 & 1 \end{bmatrix}}_{\text{A}} \cdot \underbrace{\begin{bmatrix} x_f \\ y_f \\ 1 \end{bmatrix}}_{\text{B}} = \begin{bmatrix} x_{ro} + x_f \\ y_{ro} + y_f \\ 1 \end{bmatrix} \tag{3.9}$$

where A is the transformation matrix and B is the position vector in 2D plane with x_f and y_f being the original field coordinates. The new field ring was calculated from the centered ring using this linear transformation. The location of sensors remain unchanged because they are stationary. Shaft rotation is independent from the run-out distance and direction, therefore we assume that run-out affects only the position of the field over the sensor plane.

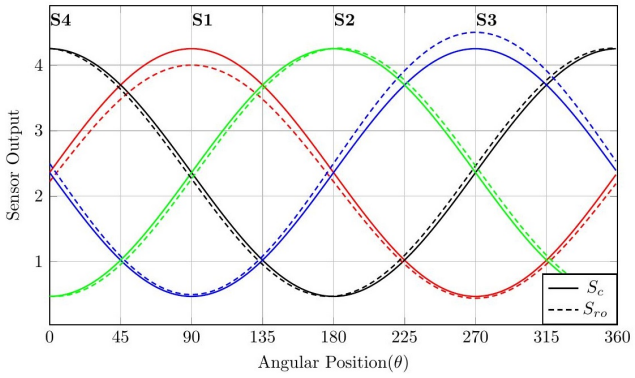


Figure 3.6. Sensor outputs for maximum run-out distance at 0° run-out direction

CHAPTER 4

RUN-OUT CORRECTION METHODS

Determining angular position information accurately in the presence of shaft run-out is a difficult task. Traditional calculations yield erroneous results when the shaft run-out is ignored. In this work, we have tackled this challenge by introducing additional sensors to the system that would help determine and correct the amount and direction of the shaft run-out, and thus increase the overall accuracy. At any given time, the system provides four data points simultaneously obtained from four sensors of the encoder. Utilizing the calibration data that was prepared beforehand, it is possible to determine the angular position information by backtracking the sensor data provided by the encoder.

In a simple case where we assume that the shaft is properly centered, the center calibration data can be searched thoroughly to determine angular position. However, when the shaft is shifted from the center, the calibration data we are searching for is no longer valid and the resulting angular position is erroneous. The best solution in this case would be to have a closed-form solution for the sensor behavior under the influence of run-out. Then the run-out distance and direction could be measured with a simple LED-PD mechanism and the calibration data could be regenerated according to this run-out information. Finally, searching over the regenerated calibration data would yield the correct angular position.

The problem is, closed-form solution is not available, and even if such a solution was available, it is hard to build a system that accurately reflects and maintains the behavior modeled as a closed-form solution. Also, there are limited amount of resources in terms of memory, processing power, and time. The algorithm should be able to run in an embedded system, and it should be fast enough to capture the rotation speed of the shaft.

The solution suggested here is to collect calibration data not only for correctly centered shaft but also for shifted shaft in distinct run-out configurations that are evenly distributed over 360 degrees. In this case, the problem boils down to finding the closest calibration dataset for the obtained sensor data. To find the closest dataset in the run-out direction, minimum distance between sensor errors is employed as a criterion. To determine the corrected angular position, centered calibration data and run-out calibration data were combined to generate a set in between and search algorithm was run through this generated dataset.

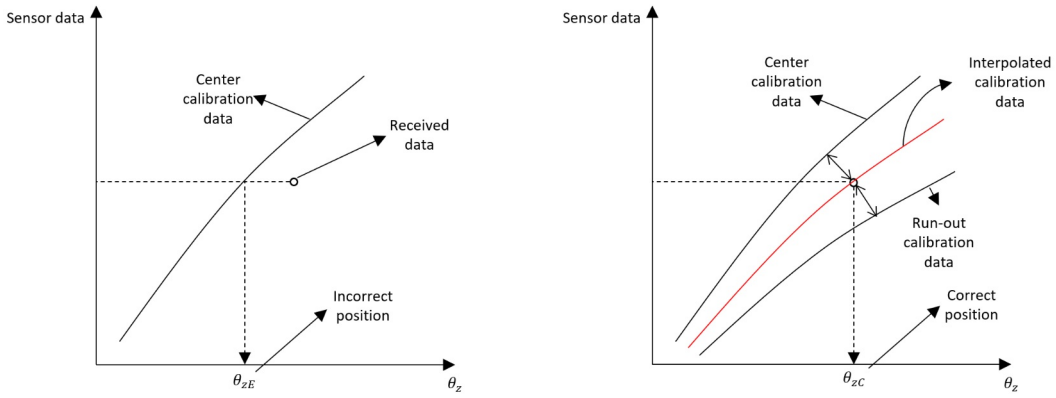


Figure 4.1. Generating calibration data by interpolation

4.1. Memory Allocation for Calibration Data

In practice, the algorithm is designed to collect calibration data from a stepper motor in $N_{\theta zCal}$ steps. In the data collection procedure, shaft of the encoder is planned to be shifted to shaft run-out range in $N\phi_{Cal}$ equally spaced run-out directions. In other words, the run-out calibration data will be collected in $\phi_{CalStep} = 360°/N\phi_{Cal}$ angular steps. In reality, an ordinary stepper motor has 200 steps per revolution and in this case, ADCs record data at every $\theta_{zCalStep} = 360°/N_{\theta zCal}$ reference angular positions which corresponds to 1.8° step size of the stepper motor. Similarly in the simulation program, the data was generated for every $\theta_{zCalStep}$ and stored in distinct memory locations for all $N\phi_{Cal}$ run-out directions. Each calibration dataset is further divided into four sections, so to say to its quadrants. Outputs of two of the sensors in each quadrant have high rate of change as a function of angular position which is preferable for yielding useful results. One of the two useful sensor outputs exhibits monotonically increasing behavior whereas the other one exhibits monotonically decreasing behavior in each quadrant.

A quadrant normally covers 90° section out of 360° full revolution of the shaft. This coverage was extended to accommodate errors caused by run-out of the shaft. Finally, calibration datasets were organized in 70 data points for each quadrant, sensor and run-out angle. This organization allows faster search operations in the region of interest after figuring out in which quadrant the received sensor data resides.

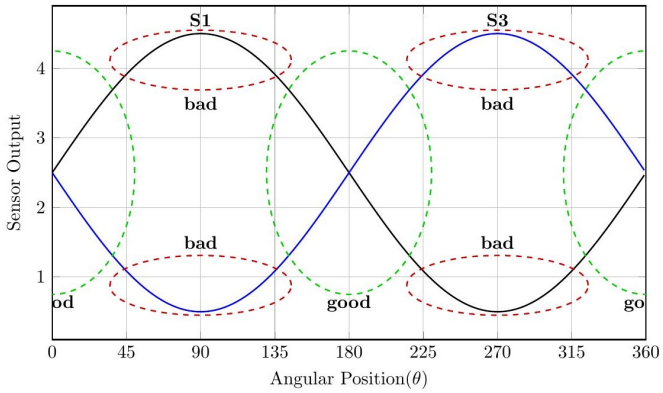


Figure 4.2. Useful and useless portions of sensor data

4.2. Correction Algorithm

The correction algorithm makes a rough position estimate from the received sensor data using the centered calibration data and then corrects the position error by using the run-out calibration data. The details of the algorithm can be discussed in three steps;

1. Finding the initial angular position estimate
2. Estimating the direction of run-out
3. Combining calibration data and finding the corrected angular position

4.2.1. Finding the Initial Angular Position Estimate

The initial angular position estimate is calculated based on center calibration data. Two of the sensors provide useful rate of change in a given quadrant, therefore the first step of the position calculation is to select the quadrant that will be used in angular position calculation. A series of comparisons between the received sensor data determines the quadrant. The conditions given in Table 4.1 show the corresponding quadrants according to sensor values.

After figuring out the quadrant, two of the sensors were selected according to Table 4.1 that lists the sensors with monotonic behavior as a function of angular position. The sensor with monotonically increasing behavior is listed as the rising sensor and the other one as the falling sensor. The sensor index k was changed to k_r to indicate the

Table 4.1. Quadrant sensor relationship

Quadrant	Position Range	Condition	Rising Sensor	Falling Sensor
1	45-135	S1>S2 & S1>S4	S2	S4
2	135-225	S2>S1 & S2>S3	S3	S1
3	225-315	S3>S2 & S3>S4	S4	S2
4	315-45	S4>S1 & S4>S3	S1	S3

rising sensor and k_f to indicate the falling sensor. The algorithm given below describes the selection process of quadrant, rising sensor and falling sensor.

Q, k_r, and k_f are the variables for quadrant, rising sensor and falling sensor indices, respectively.

procedure FINDQUAD(S(4))

if $(S(1) > S(2))\&(S(1) > S(4))$ **then** $Q \leftarrow 1$

else if $(S(2) > S(1))\&(S(2) > S(3))$ **then** $Q \leftarrow 2$

else if $(S(3) > S(2))\&(S(3) > S(4))$ **then** $Q \leftarrow 3$

else then $Q \leftarrow 4$

$k_r \leftarrow (Q \mod 4) + 1$

$k_f \leftarrow ((Q + 2) \mod 4) + 1$

return Q

end procedure

The last step of position estimation is to search over the center calibration data to find the initial position estimate. For sake of simplicity, an exhaustive search method was used where every single value of the dataset is progressively tested against the received sensor data. As the calibration data is sorted in increasing or decreasing order, the search is terminated where the error between received data and the calibration data changes sign. The pseudocode given below summarizes the search algorithm for the rising sensor data.

---

procedure FINDPOS(S(4),Q)

$\theta_{zEr} \leftarrow \theta_z$ at the beginning of the quadrant

while $S(k_r) > S_{ctr}(Q, \theta_{zEr}, k_r)$ **do**

$\theta_{zEr} \leftarrow \theta_{zEr} + \theta_{zStep}$

end while

$$\theta_{zEr} \leftarrow \frac{S_{ctr}(Q, \theta_{zEr}, k_r) - S(k_r)}{S_{ctr}(Q, \theta_{zEr}, k_r) - S_{ctr}(Q, \theta_{zEr} - 1, k_r)}(\theta_{zEr} - 1)$$

$$+\frac{S(k_r) - S_{ctr}(Q, \theta_{zEr} - 1, k_r)}{S_{ctr}(Q, \theta_{zEr}, k_r) - S_{ctr}(Q, \theta_{zEr} - 1, k_r)}(\theta_{zEr})$$

return θ_{zEr}

end procedure

In the algorithm, the position where the while loop terminates is marked as the rough position estimate. The rough position estimate is used with the consecutive data point and interpolated to obtain the fine position estimate.

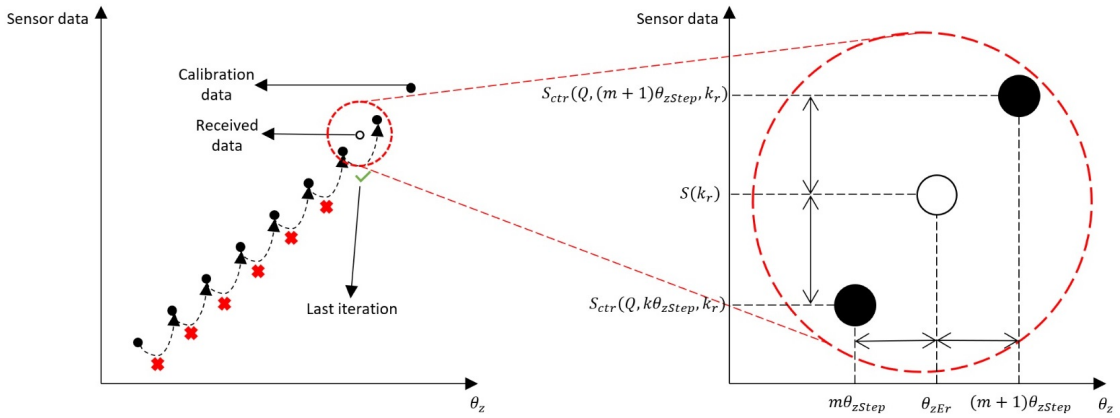


Figure 4.3. Finding position by using calibration data

The position that is calculated using rising sensor data is stored as θ_{zEr} and the other one is stored as θ_{zEf}. Averaging θ_{zEr} and θ_{zEf} yields the initial angular position estimate.

$$\theta_{zE} = \frac{\theta_{zEr} + \theta_{zEf}}{2} \mod 360 \tag{4.1}$$

It is important to perform the averaging on unwrapped θ_{zEr} and θ_{zEf} to avoid wrapping errors. To illustrate the problem, averaging 359.9° and 360.1° gives 360° which can be wrapped back to 0° correctly. On the other hand, if 360.1° was wrapped back to 0.1° before averaging then the result would have been incorrectly identified as 180°.

4.2.2. Estimation of Run-out Direction

The initial position estimate needs to be corrected using one of the available run-out calibration datasets. The aim here is to determine the most suitable calibration dataset that yields the best match with the error pattern in the present sensor outputs. The approach was based on the distance between the errors in received sensor data and the errors in calibration datasets resulting at the calculated initial position estimate. The dataset that yields the lowest absolute distance is picked as the run-out direction dataset.

Two variables are defined for the purpose of comparison to be used in the identification process. The relevance between these variables is expected to yield the direction of the run-out. The first variable is the sensor errors that are calculated by taking as reference the center calibration data evaluated at the initial position estimate θ_{zE}.

$$\Delta S_c(k) = S(k) - S_c(\theta_{zE}, k) \tag{4.2}$$

where $\Delta S_c(k)$ is the error between received sensor data and center calibration data for k'th sensor, $S(k)$ is the received data recorded by k'th sensor at reference angular position θ_z of the shaft, $S_c(\theta_{zE}, k)$ is the center calibration data recorded by k'th sensor at initial position estimate θ_{zE} of the shaft. The second variable is the sensor error between the run-out calibration data and the center calibration data evaluated at the initial position estimate θ_{zE}.

$$\Delta S_{cro}(\phi_{ro}, k) = S_{ro}(\phi_{ro}, \theta_{zE}, k) - S_c(\theta_{zE}, k) \tag{4.3}$$

where $\Delta S_{cro}(\phi_{ro}, k)$ is the error between center calibration data and calibration data at run-out direction ϕ_{ro} for k'th sensor, and $S_{ro}(\phi_{ro}, \theta_{zE}, k)$ is the run-out calibration data recorded by k'th sensor at initial position estimate θ_{zE} of the shaft in the run-out direction of ϕ_{ro}.

Total absolute distance was utilized in order to reveal the relationship between $\Delta S_c(k)$ and $\Delta S_{cro}(\phi_{ro}, k)$ as a function of ϕ_{ro}. In order to correctly identify the run-out direction in distinct shaft offset amounts, absolute distances were scaled by the maximum error found among four sensors. The maximum error among four sensors have been found as follows;

$$\Delta S_{cM} = max(|\Delta S_c(1)|, \ldots, |\Delta S_c(4)|) \tag{4.4}$$

where ΔS_{cM} is the maximum absolute error attained by absolute value of $\Delta S_c(k)$ for sensors 1,. . . ,4.

The maximum error for run-out calibration data was calculated in the same manner but individually for all run-out directions;

$$\Delta S_{croM}(\phi_{ro}) = max(|\Delta S_{cro}(\phi_{ro}, 1)|, \ldots, |\Delta S_{cro}(\phi_{ro}, 4)|) \tag{4.5}$$

where $\Delta S_{croM}(\phi_{ro})$ is the maximum absolute error at ϕ_{ro} attained by absolute value of $\Delta S_{cro}(\phi_{ro}, k)$ for sensors 1,. . . ,4.

After finding maximum absolute error, the total absolute distance for each run-out direction was calculated over the absolute sum of errors scaled by the corresponding maximum values for each sensor.

$$D(\phi_{ro}) = \sum_{k=1}^{4} \left| \frac{\Delta S_c(k)}{\Delta S_{cM}} - \frac{\Delta S_{cro}(\phi_{ro}, k)}{S_{croM}(\phi_{ro})} \right| \tag{4.6}$$

where $D(\phi_{ro})$ is the total distance of absolute values of errors between normalized errors $\Delta S_c(k)/\Delta S_{cM}$ and $\Delta S_{cro}(\phi_{ro}, k)/\Delta S_{croM}(\phi_{ro})$, at run-out direction ϕ_{ro}. The run-out direction estimate was then chosen among the total absolute distances where the minimum from the set of $D(\phi_{ro})$ occurs.

$$\phi_{roE} = \phi_{ro} \text{ where } D(\phi_{ro}) \text{ is minimum} \tag{4.7}$$

4.2.3. Correction of Angular Position Error

The corrected angular position is calculated utilizing the run-out direction estimate ϕ_{roE}. Intermediate data points are generated on the go and tested against the sensor data with the same brute force search method used in calculation of the first position estimate. The generated data is simply a combination of run-out calibration data in the direction of ϕ_{roE} and center calibration data.

The data combination is accomplished by taking a weighted average of the $S_c(\theta_z, k)$ and $S_{ro}(\phi_{roE}, \theta_z, k)$ for rising and falling sensors. Firstly, the total absolute distances are calculated to quantify the shaft offset from the center.

$$D_c = \sum_{k=1}^{4} |\Delta S_c(k)| \tag{4.8}$$

$$D_{cro} = \sum_{k=1}^{4} |\Delta S_{cro}(\phi_{roE}, k)| \tag{4.9}$$

where D_c is the total absolute distance of received sensor data to center calibration data and D_{cro} is the total absolute distance of run-out calibration data to center calibration data.

The run-out calibration data is recorded at $R_{ro} = R_{roM}$, so it defines the maximum absolute distance from the center in ϕ_{roE} direction. Therefore, weighting factors are calculated using D_c/D_{cro} ratio as a scaling factor.

$$W_{ro} = \frac{D_c}{D_{cro}} \tag{4.10}$$

$$W_c = 1 - W_{ro} = \frac{(D_{cro} - D_c)}{D_{cro}} \tag{4.11}$$

where W_c is the weight of the center calibration data and W_{ro} is the weight of the run-out calibration data.

The intermediate data points are generated by applying related weights to center and run-out calibration datasets for rising and falling sensors.

$$S_{genr}(\theta_z) = W_c S_c(\theta_z, k_r) + W_{ro} S_{ro}(\phi_{roE}, \theta_z, k_r) \tag{4.12}$$

$$S_{genf}(\theta_z) = W_c S_c(\theta_z, k_f) + W_{ro} S_{ro}(\phi_{roE}, \theta_z, k_f) \tag{4.13}$$

where $S_{genr}(\theta_z)$ is the generated intermediate dataset for the rising sensor and $S_{genf}(\theta_z)$ is the generated intermediate dataset for the falling sensor.

$S_{genr}(\theta_z)$ and $S_{genf}(\theta_z)$ datasets replace the centered calibration data in the search for individual corrected angular positions, θ_{zCr} and θ_{zCf} by using the sensor outputs $S(k_r)$ and $S(k_f)$, respectively. Finally, the corrected angular position, θ_{zC}, is calculated by averaging the two corrected positions from rising and falling sensors.

$$\theta_{zC} = \frac{\theta_{zCr} + \theta_{zCf}}{2} \tag{4.14}$$

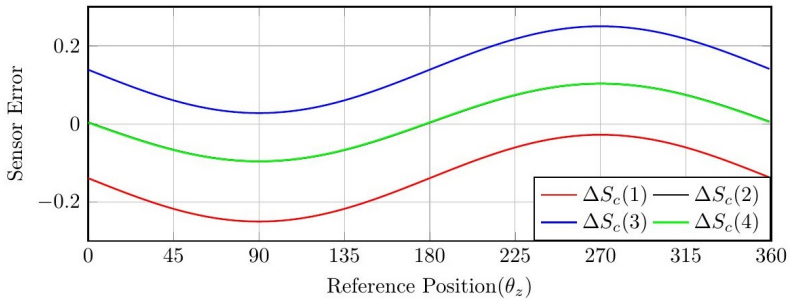
4.2.4. Improving Run-out Direction Estimation Accuracy

The algorithm described in section 4.2.2 fails to determine the run-out direction with sufficient accuracy due to error mismatch. If the error definitions given in eq.4.2 and eq.4.3 are examined closely, it can be noted that the first terms in the error expressions need to resemble each other for the algorithm to correctly match and identify the run-out direction. In the simplest case where the shaft offset from the center R_{ro} exactly matches the maximum shaft run-out range R_{roM}, it can be seen that the first terms of equations originate from different angular positions. The problem gets more complicated when R_{ro} variation is reintroduced. Although this method is still viable, better results can be obtained by characterizing the error that would result from the first position estimate, θ_{zE}.

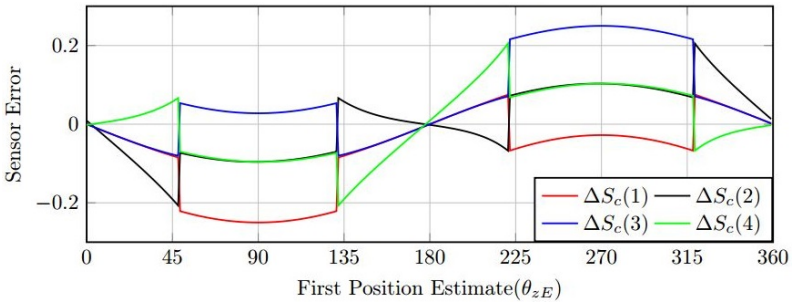
To make the problem more clear, the raw calibration data, $S_{ro}(\phi_{ro}, \theta_{zE}, k)$ used in eq.4.3, is recorded at exact reference positions and the received sensor data, $S(k)$ used in eq. 4.2, is from an unknown angular position with unknown run-out amount and direction. The initial angular position, θ_{zE}, is inevitably off-target due to presence of run-out error in $S(k)$. The algorithm uses θ_{zE}, which is incorrectly identified, to calculate the error between run-out calibration data and center calibration data in eq. 4.3. The algorithm then tries to match this off-target error calculation to the error that was calculated from received data and fails to determine the run-out direction estimate correctly.

Figure 4.4 shows the error patterns obtained under two different circumstances. The first figure shows the error pattern when reference angular position θ_z is used to calculate the error and the second one shows the error pattern obtained as a function of the initial position estimate θ_{zE}. It can be seen from the figures that the error pattern changes significantly in two of the quadrants as a consequence of the shift in initial position estimate, $\theta_{zE} - \theta_z$. This also results in discontinuities at the quadrant boundaries, since different sensor outputs are used for the angular position calculations in each quadrant.

In order to make a better estimation of run-out direction, a map of the error patterns resulting from the initial position estimate, $E_{ro}(Q_c, \phi_{ro}, \theta_{zE}, k)$, is calculated as described in the following section. This error map is utilized instead of the error obtained from the raw calibration data $\Delta S_{cro}(\phi_{ro}, k)$ in the correction algorithm. In this case, $\Delta S_{cro}(\phi_{ro}, k)$ defined in 4.2.2 was replaced with the error map $E_{ro}(Q_c, \phi_{ro}, \theta_{zE}, k)$ without changing



(a) Error Pattern based on reference positions



(b) Error Pattern based on initial position estimates

Figure 4.4. Comparison of error patterns as a function of reference and estimated angular positions. The run-out parameters are $R_{ro} = R_{roM}$, $\phi_{ro} = 0°$.

other parts of the algorithm.

$$E_{ro}(Q_c, \phi_{ro}, \theta_{zE}, k) \rightarrow \Delta S_{cro}(\phi_{ro}, k) \tag{4.15}$$

4.2.4.1. Preparing the Error Map

Having the run-out calibration data at hand, the error map is prepared by utilizing the initial position estimation algorithm. Firstly, the sensor data collected from four sensors at reference position θ_z is retrieved from run-out calibration dataset. This data is processed through the position estimation algorithm described in section 4.2.1 to find the position estimate, θ_{zE} corresponding to the reference angle θ_z. After that, the error at reference position θ_z, $\Delta S_{cro}(\phi_{ro}, k)$, is stored in the error map at the angular position reserved for the estimated angular position θ_{zE}. The difference of the first approach and this one is that this approach assumes that the first estimate is not correct and achieves the correct error pattern by making calculations beforehand. This approach allows the program to accurately match the correct run-out direction.

For every quadrant and every run-out direction the following instructions are followed to build the error map;

1. Initial estimate, θ_{zE}, is calculated using run-out calibration data $S_{ro}(\phi_{ro}, \theta_z, k)$ at the reference position θ_z. This calculation is based on the sensor output variation in the centered calibration data as a function of the angular position for each quadrant.

2. The error $\Delta S_{cro}(\phi_{ro}, k)$ at the reference position θ_z is recorded at $E_{ro}(Q_c, \phi_{ro}, \theta_{zE}, k)$ where θ_{zE} is the initial estimate calculated in the first step and Q_c is the index of the quadrant used in the calculation.

where the algorithm in implementation can be described as follows;

---

procedure BUILD ERROR MAP

for all $Q_c \leftarrow 1, 2, 3, 4$ **do**

for all $\phi_{ro} \leftarrow 0, 45, .., 315$ **do**

$\theta_{z1} \leftarrow$ the first θ_z in the extended region

$\theta_{zE1} \leftarrow$ FINDPOS$(S_{ro}(\phi_{ro}, R_{ro}, \theta_{z1}, k), Q_c)$

for all the remaining θ_z in the extended region **do**

$\theta_{z2} \leftarrow \theta_z$

$\theta_{zE2} \leftarrow$ FINDPOS$(S_{ro}(\phi_{ro}, R_{ro}, \theta_{z2}, k), Q_c)$

for all $\theta_{zM} = N * \theta_{zStep}$ where $\theta_{zE1} < \theta_{zM} < \theta_{zE2}$ **do**

$$W_{M1} \leftarrow \frac{\theta_{zE2} - \theta_{zM}}{\theta_{zE2} - \theta_{zE1}}$$

$$W_{M2} \leftarrow \frac{\theta_{zM} - \theta_{zE1}}{\theta_{zE2} - \theta_{zE1}}$$

$$E_{ro}(Q_c, \phi_{ro}, \theta_{zM}, k) \leftarrow W_{M1}(S_{ro}(\phi_{ro}, \theta_{z1}, k) - S_c(\phi_{ro}, \theta_{zM}, k)) + W_{M2}(S_{ro}(\phi_{ro}, \theta_{z2}, k) - S_c(\phi_{ro}, \theta_{zM}, k))$$

end for

$\theta_{z1} \leftarrow \theta_{z2}$

$\theta_{zE1} \leftarrow \theta_{zE2}$

end for

end for

end for

end procedure

Although these two steps are fairly straightforward, recording the results as instructed in step 2 is complicated. θ_{zE} calculated in the first step is not an integer multiple of the 1.8° reference angle step size, thus the sensor errors calculated for θ_{zE} cannot be placed directly in the error map. Instead, the next position estimate is calculated in the extended quadrant range and any error map position at the integer multiples of 1.8 is filled in by a weighted average of the errors of consecutive reference positions. The process uses the current position estimate along with the previous one, so θ_{z1} and θ_{zE1} are used for the first reference position and first estimate that the algorithm yields, θ_{z2} and θ_{zE2} are used for second reference position and the corresponding position estimate. The following calculations are performed for every integer multiple of the angular step θ_{zM} between θ_{zE1} and θ_{zE2} to prepare the error map;

$$W_{M1} = \frac{\theta_{zE2} - \theta_{zM}}{\theta_{zE2} - \theta_{zE1}} \tag{4.16}$$

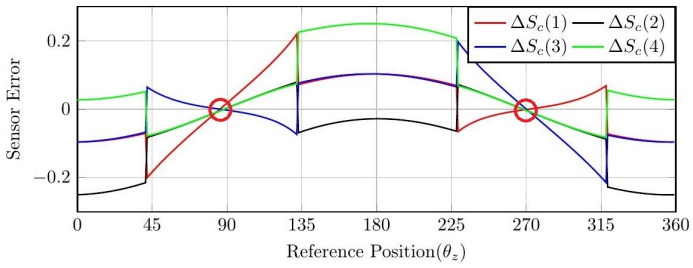
$$W_{M2} = \frac{\theta_{zM} - \theta_{zE1}}{\theta_{zE2} - \theta_{zE1}} \tag{4.17}$$

$$\begin{aligned} E_{ro}(Q_c, \phi_{ro}, \theta_{zM}, k) = W_{M1}(S_{ro}(\phi_{ro}, \theta_{z1}, k) - S_c(\phi_{ro}, \theta_{zM}, k)) \\ +W_{M2}(S_{ro}(\phi_{ro}, \theta_{z2}, k) - S_c(\phi_{ro}, \theta_{zM}, k)) \end{aligned} \tag{4.18}$$

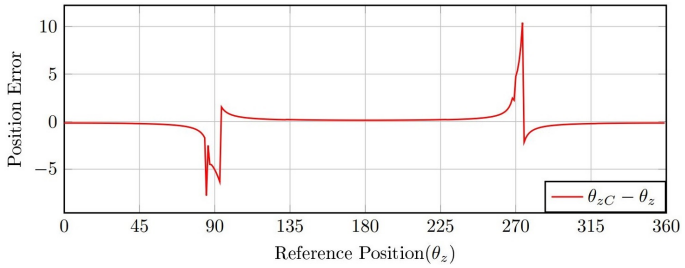
where W_{M1} and W_{M2} are the weights of the past and current position estimates, respectively, and $E_{ro}(Q_c, \phi_{ro}, \theta_{zM}, k)$ is the error map entry at θ_{zM} for the run-out direction ϕ_{ro} and in the quadrant of calculation Q_c.

4.2.5. Dealing with Singularities

For certain angular positions, the errors $\Delta S_c(k)$ between received sensor data and center calibration data get so small that neither the correct run-out direction can be identified nor the weights can be deduced to apply the correction. These special occurrences, where the errors cannot be observed but actually exist, are called as singular configurations. Since the algorithm cannot correctly find parameters for these configurations, the correction attempts result in large errors. Therefore, a way to avoid singularities need to be developed in order to handle such conditions.



(a) Singular points on error pattern



(b) Corresponding errors in position estimates

Figure 4.5. Marginal position errors caused by singularities

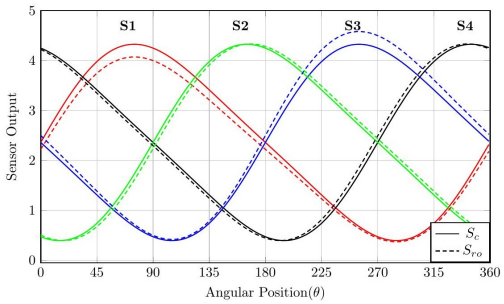
4.2.5.1. Adding Second Harmonic to Angular Field Variation

The singularities in error map result from combined symmetry properties of the angular field variation and the placement of sensors. One way to solve the singularity issue was to add second harmonic to angular field variation function. The second harmonic breaks the symmetry of the system to provide sufficient error in all angular positions so that singular configurations can be avoided. For the addition of the second harmonic, the angular field variation function previously given in equation 3.2 is modified as follows;

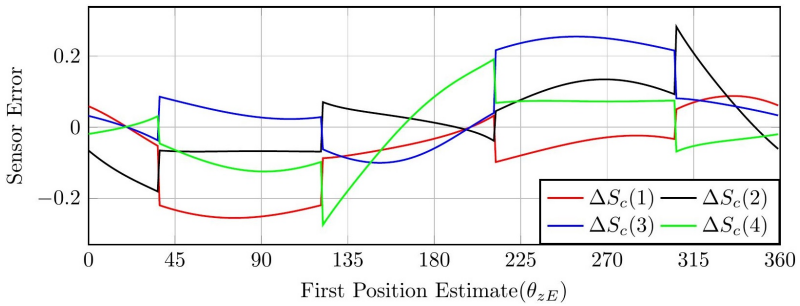
$$I_{fA}(\theta) = I_{fos} + I_{fAvar1} sin(\theta) + I_{fAvar2} sin(2\theta) \tag{4.19}$$

where I_{fAvar2} is the amplitude of the second harmonic inserted.

Figure 4.6a shows the impact of second harmonic on obtained signals. Error values obtained for sensors shown in Figure 4.6b demonstrates how errors are always kept at a significant level.



(a) Sensor outputs with the addition of second harmonic

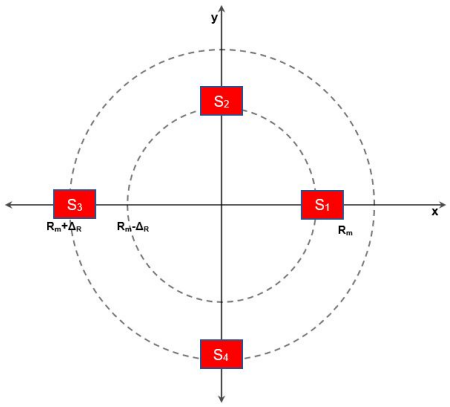


(b) Changes in error pattern based on initial position estimates

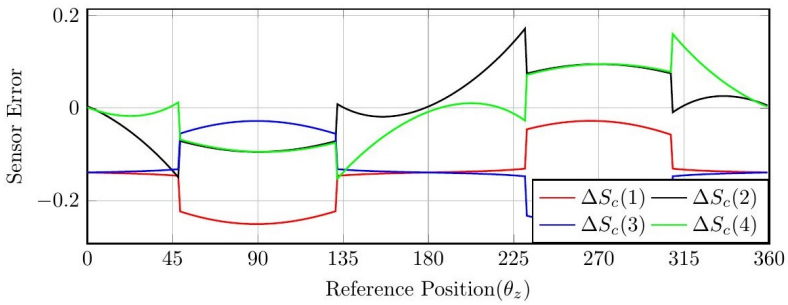
Figure 4.6. Changes in sensor errors after second harmonic addition

4.2.5.2. Changing Sensor Geometry

Another attempt to eliminate error map singularities was to change the locations of the sensors. The symmetry in sensor locations is another reason that there are singular configurations. To break this symmetry, sensors S1 and S2 kept at the same location ($R_{sns1} = R_{sns2} < R_{fm}$) while the other sensors are slightly moved away from the shaft, and they are specifically located on the other side of the middle ring ($R_{sns3} = R_{sns4} > R_{fm}$). Figure 4.7a shows the changes in the sensor locations and Figure 4.7b shows the change in the error patterns.



(a) Changes in sensor positions



(b) Changes in error pattern based on initial position estimates

Figure 4.7. Changes in sensor errors after changing sensor geometry

CHAPTER 5

RESULTS AND DISCUSSION

The performance of the correction method is demonstrated by both quantifying overall performance in tables and providing graphical results for specific cases. The overall performance is quantified by carrying out a large number of trials. Since it is not possible to show all the results in graphs, a number of important configurations are selected to show the behavior and the limitations of the algorithm in the worst case scenarios. The graphical results are given for a single revolution of the shaft.

The complete evaluation of the correction algorithm requires scanning of three variables in their full range:

1. true angular position, θ_z, of the shaft
2. radial direction, ϕ_{ro}, of the shaft run-out
3. run-out distance, R_{ro}, of the shaft movement

In the simulation, θ_z and ϕ_{ro} vary between 0° and 360° in 1° steps and R_{ro} varies between 0.1mm and 0.5mm in 0.1mm steps.

Different case scenarios are shown to discuss the performance of the algorithms under specific run-out configurations. These scenarios are carefully chosen to show the development process of the correction methods and weaknesses and strengths of the suggested algorithms.

5.1. Simulation Conditions and Initial Results

The algorithm was tested arbitrarily by adjusting the run-out distance and direction utilizing the LabVIEW GUI, and the results for that specific run-out configuration were calculated for all reference angular positions and displayed graphically. Also, estimation and correction parameters of the program has been monitored at each step of the algorithm to evaluate its behavior.

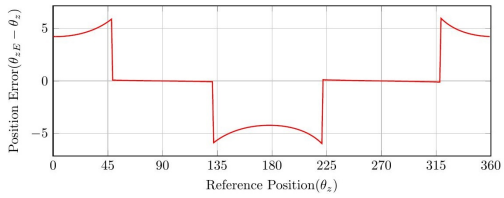
The sensor datasets are generated for four sensors in the field where the field variation is defined by the parameters given in Table 5.1. The signals or the field shapes that were given in Chapter 3 are not repeated in the following sections to keep focus on

the correction methods. Unless it is specified otherwise, triangular variation is employed in test trials.

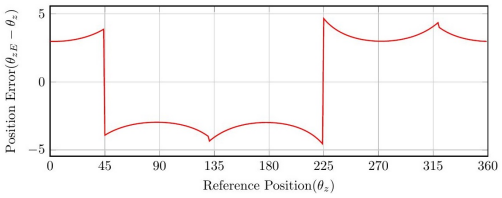
Table 5.1. Sensor field parameters

Description	Parameter	Value
Distance of sensors	R_{sns}	9.5mm
Middle circle radius of the field ring	R_{fm}	10mm
Half width of the field ring	w_{fh}	9mm
Maximum allowed run-out distance	R_{roM}	0.5mm
Offset of angular field variation	I_{fos}	2.5V
Amplitude of angular field variation	I_{fAvar}	2V

Figure 5.1 shows the position error pattern for two different configurations where the field is generated with triangular distribution. Our aim was to correct angular position errors in all angular positions that result from any run-out configuration by using only the available data from four sensors at that particular instant.



(a) Position errors for $R_{ro} = R_{roM}$ and $\phi_{ro} = 0°$



(b) Position errors for $R_{ro} = R_{roM}$ and $\phi_{ro} = 45°$

Figure 5.1. Exemplary position errors of initial position estimates

5.2. Performance Evaluation Tools

The overall performance of the proposed method was evaluated under two main criteria;

1. Run-out direction estimation accuracy
2. Position correction performance

The estimation accuracy is quantified by scoring the direction estimation based on the difference between the estimated run-out direction and the correct run-out direction. The difference $\Delta\phi_{roE}$ was calculated as shown in eq. 5.1;

$$\Delta\phi_{roE} = \phi_{roE} - \phi_{ro} \tag{5.1}$$

where ϕ_{roE} is the estimated run-out direction and ϕ_{ro} is the correct run-out direction. The difference is wrapped into +/-180 range following the rules given in eq. 5.2

$$f_w(x) = \begin{cases} x + 360 & \text{if } \Delta_{\phi_{roE}} \leqslant -180 \\ x & \text{if } -180 < \Delta_{\phi_{roE}} \leqslant 180 \\ x - 360 & \text{otherwise} \end{cases} \tag{5.2}$$

The wrapped difference is mapped into individual score where maximum score is $N_{\phi Cal} - 1$ and minimum score is 0;

$$d = \frac{|f_w(\Delta\phi_{roE})|}{\frac{\phi_{CalStep}}{2}} \tag{5.3}$$

$$s_i = (N_{\phi Cal} - 1) - \lfloor d \rfloor \tag{5.4}$$

where d is the scaled difference and s_i is the individual score.

The overall score was obtained by accumulating the individual scores and then scaling the total score to produce a value between 0 and 100.

$$S_{acc} = \frac{100}{N} \sum_{i=1}^{N} \frac{s_i}{N_{\phi Cal} - 1} \tag{5.5}$$

Figure 5.2 shows the corresponding scores for each run-out direction when $N_{\phi Cal}$ is 8 and $\phi_{CalStep}$ is 45°. When the run-out direction angle is 32° and the algorithm correctly selects 45° as ϕ_{roE}, the corresponding score s_i is 7 which is the maximum attainable score. On the contrary, if the algorithm selects 225° as ϕ_{roE}, the corresponding score s_i is 0 which is the minimum attainable score.

Position correction performance is quantified over two metrics in order to observe the degree of improvement. The two metrics are maximum error $\Delta\theta_{zMax}$ and mean error $\Delta\theta_{zMean}$ that were found as follows

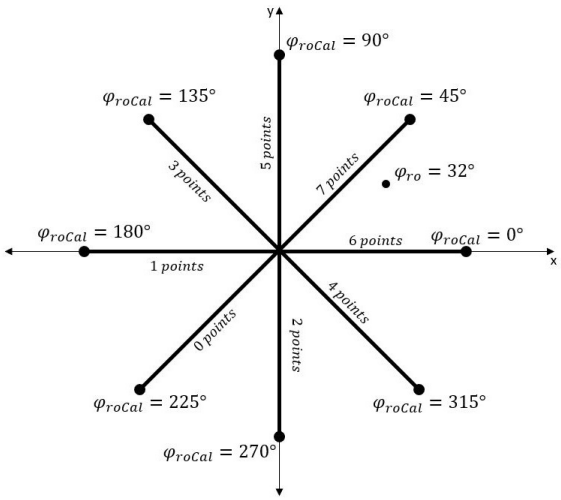


Figure 5.2. Direction estimation scores when $\phi_{ro} = 32°$

$$\Delta\theta_{zMax} = \text{maximum } |\Delta\theta_z| \text{ among } N \text{ trials} \tag{5.6}$$

$$\Delta\theta_{zMean} = \frac{1}{N}\sum_{n=1}^{N}|\Delta\theta_z| \tag{5.7}$$

where $\Delta\theta_z$ is the error in angular position, which is equal to either the first estimate error $\Delta\theta_{zE}$ or the error $\Delta\theta_{zC}$ obtained after the correction algorithm.

The mean and maximum errors in initial position estimates are 1.81 and 6.08 degrees, respectively. The overall performance of the method is determined by how much improvement is made over these parameters.

5.3. Progressive Evaluation of Method Performance

Performance of the method is evaluated at each step of the development which are listed in versions in Table 5.2. At each step, the changes made in the correction method are briefly explained and then the performance evaluation parameters are shown for the particular version as well as the prior versions. The performance of the last method version is inspected further in specific cases over the graphical results for a single revolution of the shaft.

Table 5.2. Method versions

	Error Source	$N_{\phi Cal}$	Singularity Avoidance
Version 1	Raw Data	8	No
Version 2	Error Map	8	No
Version 3	Error Map	16	No
Version 4	Error Map	16	$I_{fA}(\theta)$ modification
Version 5	Error Map	16	Sensor geometry

5.3.1. Version 1: Using Raw Calibration Data

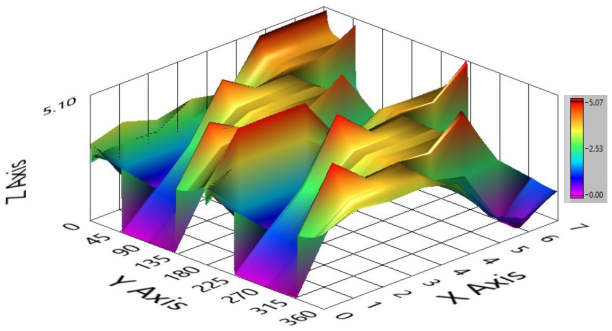
The first version of the method made use of calibration datasets as error sources to estimate the run-out direction. This version resulted in low direction estimation accuracy and little or no improvement in mean and maximum errors as listed in Table 5.3.

Table 5.3. Performance evaluation parameters for method version 1

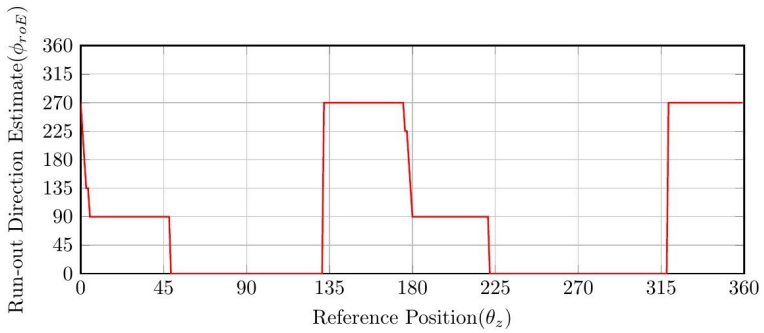
	Error Source	$N_{\phi Cal}$	Singularity Avoidance	S_{acc}	$\Delta\theta_{zMean}$	$\Delta\theta_{zMax}$
Version 1	Raw Data	8	No	76.24	1.80	5.95

Figure 5.3a shows the distance values (z axis) for all θ_z angular positions (y axis) for each calibration dataset (x axis). The calibration dataset in the direction that yields the shortest distance (one of the numbers from x axis) is chosen as the run-out direction estimate. In short, we are looking for unique minima in x axis. In this particular case the run-out configuration was $\phi_{ro} = 0°$ and $R_{ro} = R_{roM}$, therefore the graph needs to have minima at $x = 0$ for all θ_z angular positions. However, this is only true for first and third quadrants and the algorithm failed to identify the correct run-out directions in other quadrants. The correct run-out direction estimates were only determined in quadrants one and three as seen in the Figure 5.3b.

The failure of the position correction is a consequence of the shift in the first position estimate as highlighted in Figure 5.4. The shift causes error mismatch which was described in depth in section 4.2.4.



(a) Isometric view of 3D graph of distances



(b) Resulting run-out direction estimates

Figure 5.3. Direction estimation results for method version 1 when $\phi_{ro} = 0°$ and $R_{ro} = R_{roM}$

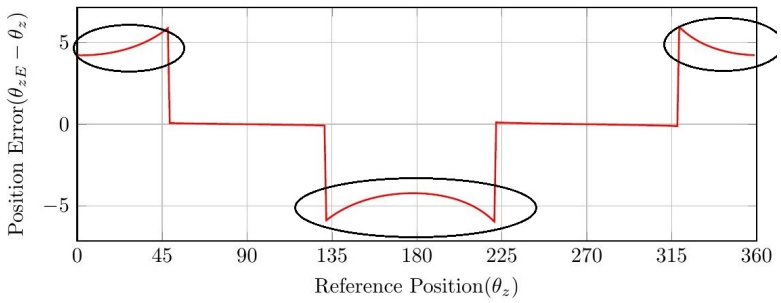


Figure 5.4. Shift highlights on position errors when $\phi_{ro} = 0°$ and $R_{ro} = R_{roM}$

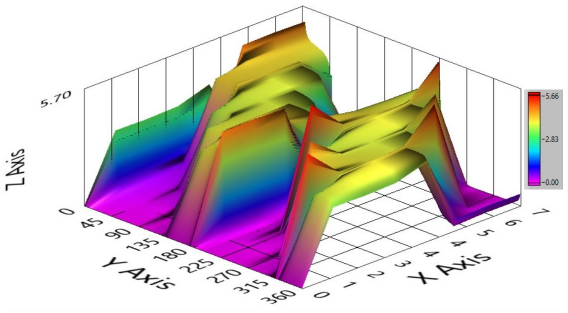
5.3.2. Version 2: Using Error Map

The failure due to error mismatch was corrected in the second version by preparing an error map. Incorporating the error map as the error source dramatically improved direction estimation accuracy score S_{acc} from 76.24 to 93.12 and reduced mean error $\Delta\theta_{zMean}$ from 1.80 to 1.04.

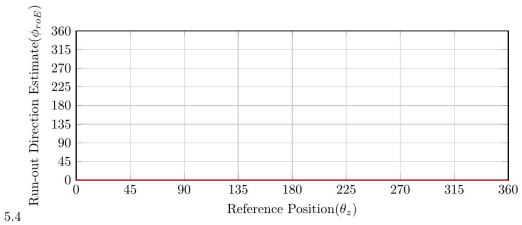
Table 5.4. Updated performance evaluation parameters for method version 2

	Error Source	$N_{\phi Cal}$	Singularity Avoidance	S_{acc}	$\Delta\theta_{zMean}$	$\Delta\theta_{zMax}$
Version 1	Raw Data	8	No	76.24	1.80	5.95
Version 2	Error Map	8	No	93.12	1.04	16.81

Figure 5.5a shows the resulting distance values when $\phi_{ro} = 0°$ and $R_{ro} = R_{roM}$. The distance function has a minimum for all θ_z angular positions and, as anticipated, the run-out direction is estimated correctly at all angular positions as seen in Figure 5.5b.



(a) 3D visualization of the distances

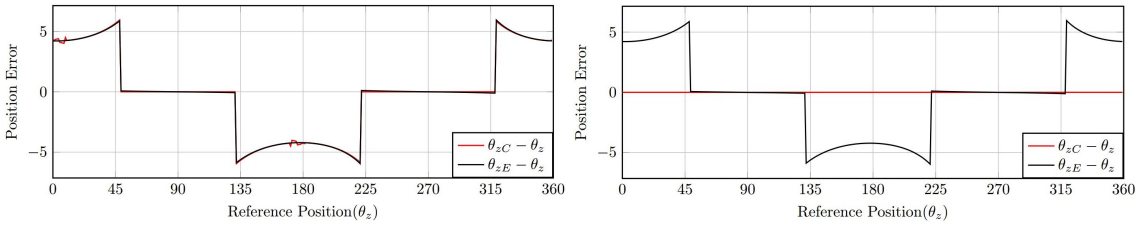


(b) Resulting run-out direction estimates

Figure 5.5. Direction estimation results for method version 2 when $\phi_{ro} = 0°$ and $R_{ro} = R_{roM}$

Figure 5.6 shows the position errors after correction for both versions of the method. The first method fails to correct the errors in position even though the tested sensor data exactly match the calibration data with the same run-out parameters. On the other hand, the second version corrects the entire error at all angular positions.

The main problem in using an error map is the large number of elevated errors. As seen in Table 5.4 $\Delta\theta_{zMax}$ suddenly increased to 16.81. This problem was overcome



(a) Angular position errors for version 1

(b) Angular position errors for version 2

Figure 5.6. Error correction comparison of version 1 and version 2 when $\phi_{ro} = 0°$ and $R_{ro} = R_{roM}$

in the later versions of the algorithm.

5.3.3. Version 3: Increasing the Number of Calibration Datasets

The algorithm was further tested by using calibration data obtained in 16 equally separated run-out directions instead of 8. The effect of doubling the number of calibration datasets is listed in Table 5.5.

Table 5.5. Updated performance evaluation parameters for method version 3

	Error Source	$N_{\phi Cal}$	Singularity Avoidance	S_{acc}	$\Delta\theta_{zMean}$	$\Delta\theta_{zMax}$
Version 1	Raw Data	8	No	76.24	1.80	5.95
Version 2	Error Map	8	No	93.12	1.04	16.81
Version 3	Error Map	16	No	94.73	0.75	14.67

Despite the slight improvement in accuracy score S_{acc}, approximately 25% reduction was observed in the mean error $\Delta\theta_{zMean}$. At this stage, the $\Delta\theta_{zMax}$ was still very high, and the reason was found to be the singularities introduced in the error map and the multiple minima in certain configurations. These problems were inspected in different case scenarios.

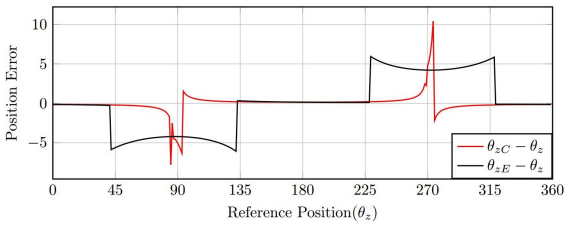
Case 1: Failures Due to Singularity in The Error Map

Run-out direction: $\phi_{ro} = 92°$

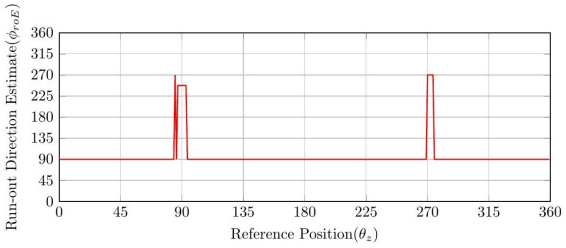
Run-out distance: $R_{ro} = R_{roM}$

Figure 5.7 shows the position errors, direction estimates and sensor errors at all angular positions. At around 90° and 270° there are two large position errors that can be

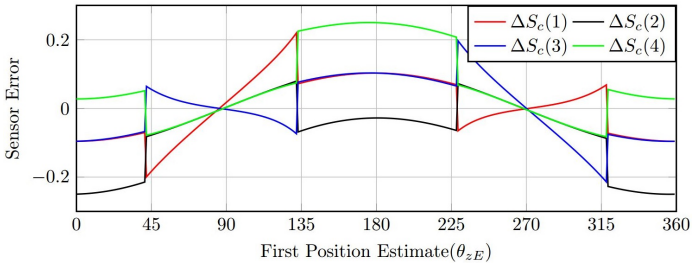
seen in Figure 5.7a. The direction estimates at these positions are incorrectly identified as seen in Figure 5.7b due to singularities in the sensor errors. Figure 5.7c shows the errors as a function of first position estimate. It can be noted that all the errors approach 0 at around 90° and 270° positions. Thus, the algorithm can not detect the correct run-out direction at these regions where the sensor error is undetectable but the position error exist.



(a) Position errors before and after the correction



(b) Run-out direction estimates



(c) Errors in sensor outputs as a function of first position estimate

Figure 5.7. Results for method version 3 when $\phi_{ro} = 92°$ and $R_{ro} = R_{roM}$

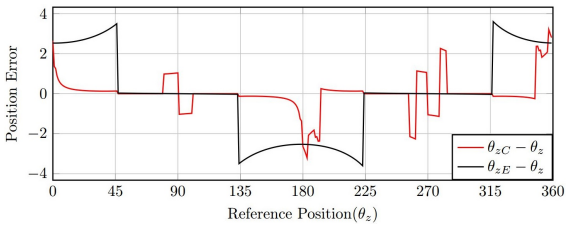
Case 2: Failures Resulting from Multiple Minima in Error Distance Function

Run-out direction: $\phi_{ro} = 0°$

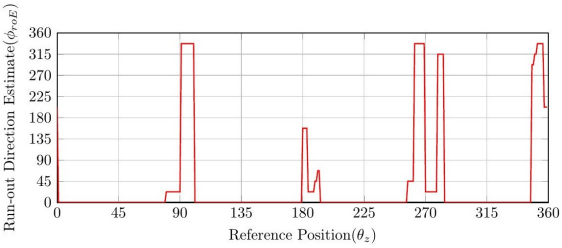
Run-out distance: $R_{ro} = 0.3mm$

When the run-out distance is decreased to $R_{ro} = 0.3mm$ and run-out distance kept at $\phi_{ro} = 0°$, singularities occur at around angular positions 0° and 180°. In addition to that, position errors start to appear at two more regions at around 90° and 270°. In these regions, the correct run-out direction could not be identified as seen in Figure 5.8b due to multiple minima in the error distance function. Figure 5.8d shows the top view of the error distance function. Violet color indicates that the distance is small and yellow color indicates that the distance is large. It can be clearly seen that the regions around the actual minima are also very low and the run-out directions in these regions are competing with each other to determine the run-out direction. This also means that the method is

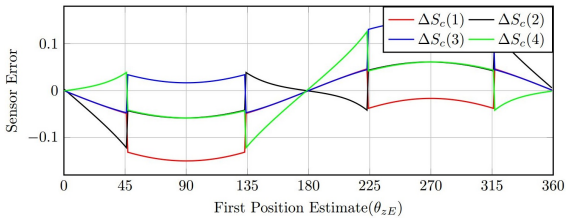
susceptible to noise in its current state since small changes in sensor outputs can lead to incorrect estimation of run-out direction easily.



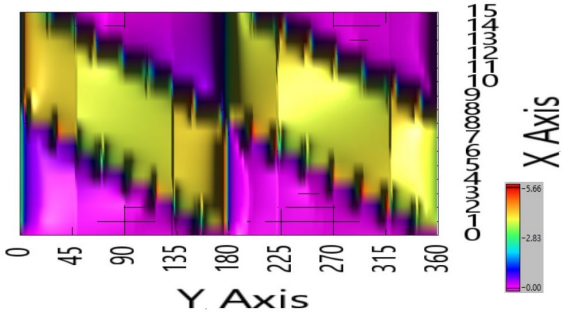
(a) Angular position errors before and after the correction



(b) Run-out direction estimates



(c) Errors in sensor outputs as a function of first position estimate



(d) Top view of 3D distance function

Figure 5.8. Results for method version 3 when $\phi_{ro} = 0°$ and $R_{ro} = 0.3mm$

All in all, at this point, the method has two main flaws that jeopardize the viability as a correction method;

1. Singularities in the error map
2. Multiple minima in error distance function

These flaws result in incorrect identification of run-out direction and thus large position errors. They were solved in later versions of the program by introducing asymmetries to the system.

5.3.4. Version 4: Changing Angular Field Variation

The improvement introduced in the fourth version of the method was singularity avoidance by adding a second harmonic to the angular field variation. The updated field parameters after addition of the second harmonic with amplitude of $I_{fAvar2} = 0.3V$ are listed in Table 5.6.

Table 5.6. Updated field parameters for the additions of the second harmonic

Description	Parameter	Value
Distance of sensors	R_{sns}	9.5mm
Middle circle radius of the field ring	R_{fm}	10mm
Half width of the field ring	w_{fh}	9mm
Maximum allowed run-out distance	R_{om}	0.5mm
Offset of angular field variation	I_{fos}	2.5V
Amplitude of angular field variation	I_{fAvar}	2V
Amplitude of second harmonic of angular field variation	I_{fAvar2}	0.3V

Figure 5.9 shows how the addition of the second harmonic in the angular field variation affects the sensor errors when $\phi_{ro} = 0°$ and $R_{ro} = R_{roM}$. It can be noted that at least one of the sensors exhibit significant error at all angular positions.

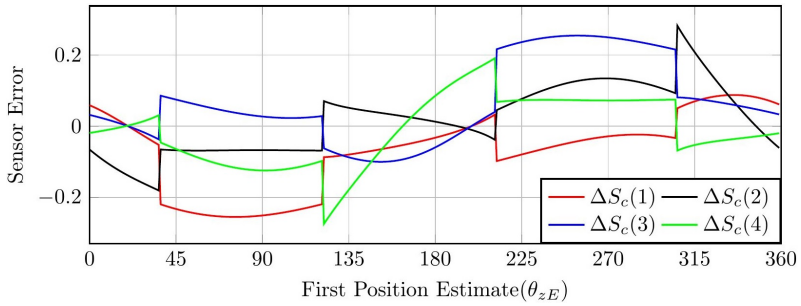


Figure 5.9. Change in error pattern with the addition of the second harmonic when $\phi_{ro} = 0°$ and $R_{ro} = R_{roM}$

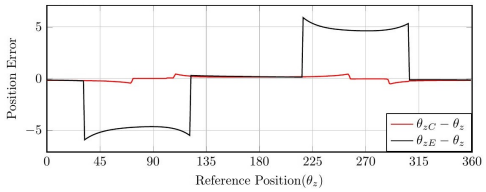
The improvement also manifests itself in overall results that are listed in Table 5.7. Compared to third version of the method, accuracy score S_{acc} increased from 94.73 to 99.09 and mean error $\Delta\theta_{zMean}$ decreased from 0.75 to 0.39. Besides, the most critical problem about using error maps, which was the large errors due to singularities, is solved. The maximum error $\Delta\theta_{zMax}$ is now significantly lower than the error obtained in second and third versions of the method.

Figures 5.10a and 5.10b show the position error graphs for the case scenarios examined for the third version of the method. It can be noted that the marginal errors associated with singularities and multiple minima are eliminated.

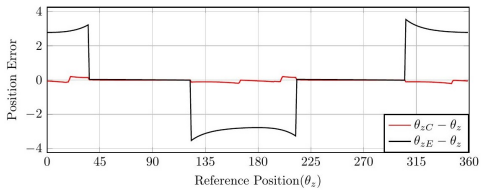
Adding second harmonic makes the minima more distinguishable by lifting the distance values up in the neighborhood of correct minima points as seen in Figure 5.11. In this manner, the algorithm also avoids the problems related to multiple minima. However,

Table 5.7. Updated performance evaluation parameters for method version 4

	Error Source	$N_{\phi Cal}$	Singularity Avoidance	S_{acc}	$\Delta\theta_{zMean}$	$\Delta\theta_{zMax}$
Version 1	Raw Data	8	No	76.24	1.80	5.95
Version 2	Error Map	8	No	93.12	1.04	16.81
Version 3	Error Map	16	No	94.73	0.75	14.67
Version 4	Error Map	16	$I_{fA}(\theta)$ modification	99.09	0.39	3.46



(a) Position error graph for case 1, where $R_{ro} = R_{roM}$ and $\phi_{ro} = 92°$



(b) Position error graph for case 2, where $R_{ro} = 0.3mm$ and $\phi_{ro} = 0°$

Figure 5.10. Position error graphs after addition of second harmonic

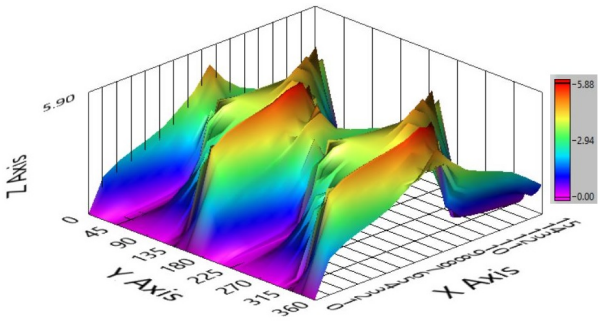
it can be seen that the values in minima regions are still very close to each other which indicates that the algorithm is still susceptible to external noise.

5.3.5. Version 5: Changing Sensor Geometry

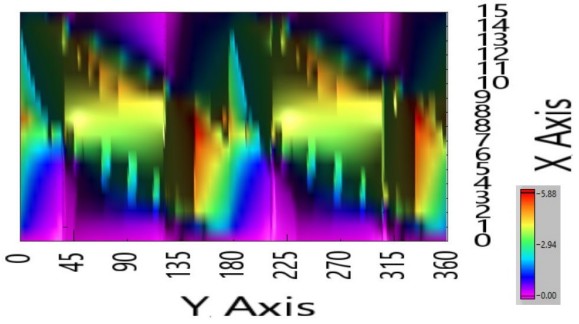
Although the fourth version of the method was successful in avoiding singularities, it did not provide a significant change between minima points. Changing sensor geometry on the other hand, not only solved the singularity issue but also solved the multiple minima problem.

The first outcome of changing sensor positions was a different error pattern in the first position estimates as shown in Figure 5.12. To achieve these error patterns, positions of S1 and S2 are kept the same whereas the S3 and S4 are shifted 1 mm outwards. In other words the new settings are $R_{sns3} = 10.5mm$ and $R_{sns4} = 10.5mm$ where $R_{sns1} = 9.5mm$ and $R_{sns2} = 9.5mm$ remain unchanged.

The benefit of changing the geometry can also be observed from the error parameters of the first position estimates. The initial error parameters without correction are

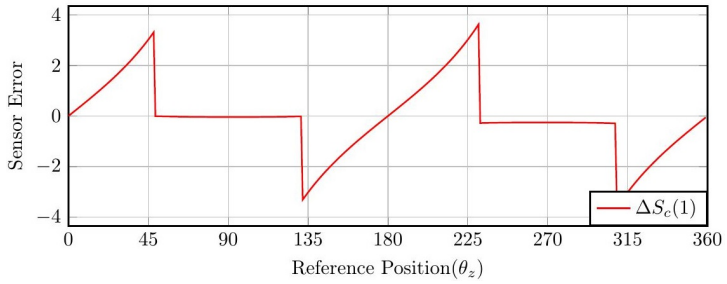


(a) Isometric view of error distance function

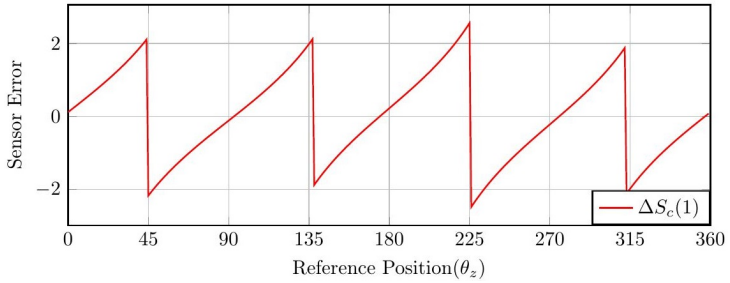


(b) Top view of error distance function

Figure 5.11. Graphical visualization of error distance function



(a) Position errors for altered sensor geometry, when $R_{ro} = 0.5mm$ and $\phi_{ro} = 0°$



(b) Position errors for altered sensor geometry, when $R_{ro} = 0.5mm$ and $\phi_{ro} = 45°$

Figure 5.12. Position errors examples for alternative sensor geometry

listed in Table 5.8.

Table 5.8. Change in initial error parameters when sensor geometry is altered

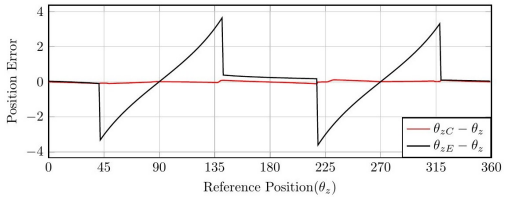
	$\Delta\theta_{zMean}$	$\Delta\theta_{zMax}$
Former geometry	1.81	6.08
Latter geometry	0.58	3.79

Table 5.9 lists the updated performance parameters after sensor geometry is altered. There is a slight increase in accuracy score S_{acc}, but a dramatic decrease in mean error $\Delta\theta_{zMean}$ and max error $\Delta\theta_{zMax}$.

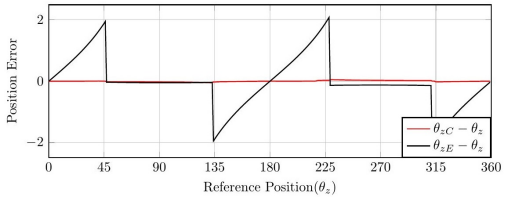
Table 5.9. Updated performance evaluation parameters for method version 5

	Error Source	$N_{\phi Cal}$	Singularity Avoidance	S_{acc}	$\Delta\theta_{zMean}$	$\Delta\theta_{zMax}$
Version 1	Raw Data	8	No	76.24	1.80	5.95
Version 2	Error Map	8	No	93.12	1.04	16.81
Version 3	Error Map	16	No	94.73	0.75	14.67
Version 4	Error Map	16	$I_{fA}(\theta)$ modification	99.09	0.39	3.46
Version 5	Error Map	16	Sensor geometry	99.61	0.07	1.12

Figure 5.13 shows the position errors for the case scenarios examined in third and fourth versions. It is clear that the errors associated with singularities and multiple minima no longer exist, and the error profile after the correction is lower compared to the prior versions.



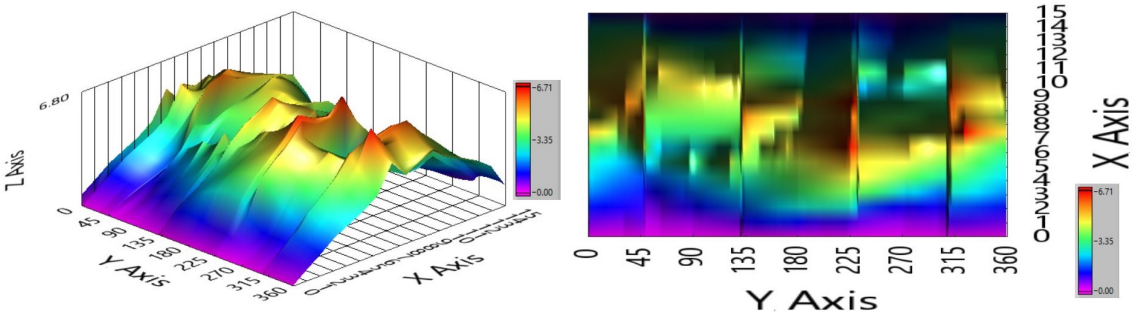
(a) Position error graph for case 2, where $Rro = 0.5mm$ and $\phi_{ro} = 92°$



(b) Position error graph for case 3, where $Rro = 0.3mm$ and $\phi_{ro} = 0°$

Figure 5.13. Position error graphs with the altered sensor geometry

Figure 5.14 shows the error distance function for the fifth version of the method when $R_{ro} = R_{roM}$ and $\phi_{ro} = 0$. The minima points for all angular positions are gathered at the line for calibration data obtained at $\phi_{ro} = 0$, and there is no more close competition between different run-out datasets.



(a) Isometric view of error distance function

(b) Top view of error distance function

Figure 5.14. Error distance function for altered sensor geometry when $\phi_{ro} = 0°$ and $R_{ro} = R_{roM}$

5.4. Performance of the Method on Different Radial Field Variations

The viability of the method was also examined for different radial field variations. The sensor locations were slightly changed to obtain similar mean errors for a fair comparison and also to utilize linear regions of the field variations.

Table 5.10. Sensor geometry configurations for different radial variation functions

	Triangular	**Cosine**	**Gaussian**
$R_{sns1,2}(mm)$	9.5	7.25	8
$R_{sns3,4}(mm)$	10.5	12.75	12

The results given in Table 5.11 shows that similar results can be achieved using cosine or Gaussian distributions in radial field variations.

Table 5.11. Error metrics for different radial variation functions

	Triangular		**Cosine**		**Gaussian**	
	Before	**After**	**Before**	**After**	**Before**	**After**
$\Delta\theta_{zMax}$	3.79	1.12	3.35	0.99	3.62	1.02
$\Delta\theta_{zMean}$	0.58	0.07	0.58	0.07	0.56	0.07

5.5. Success Rate as a Function of Final Error

Another success criterion for the correction method was defined to show the performance of the algorithm for all run-out conditions. In practice, a correction algorithm should work for all θ_z, ϕ_{ro}, and R_{ro}, with no exceptions. The purpose of the success rate evaluation is to find out how good the correction method is when no exceptions are allowed.

In this sense, the success condition is defined by;

$$\frac{\Delta\theta_{zC}}{\Delta\theta_{zMax}} < N\% \tag{5.8}$$

and the number of cases where $\Delta\theta_{zC}$ is kept below $N\%$ of $\Delta\theta_{zMax}$ are counted for all θ_z, ϕ_{ro}, and R_{ro}. If the initial error is too small then correction is not necessary. Taking this into consideration, success rate is evaluated only when there is significant error at the beginning;

$$\frac{\Delta\theta_{zE}}{\Delta\theta_{zMax}} \geq N\% \tag{5.9}$$

Percentage of successful cases as a function of percentage of initial error to maximum error is shown in Figure 5.15 for the fifth version of the method. The algorithm can keep the error below 28% of θ_{zMax} for 100% of the cases and below 12% of θ_{zMax} for 99% of the cases.

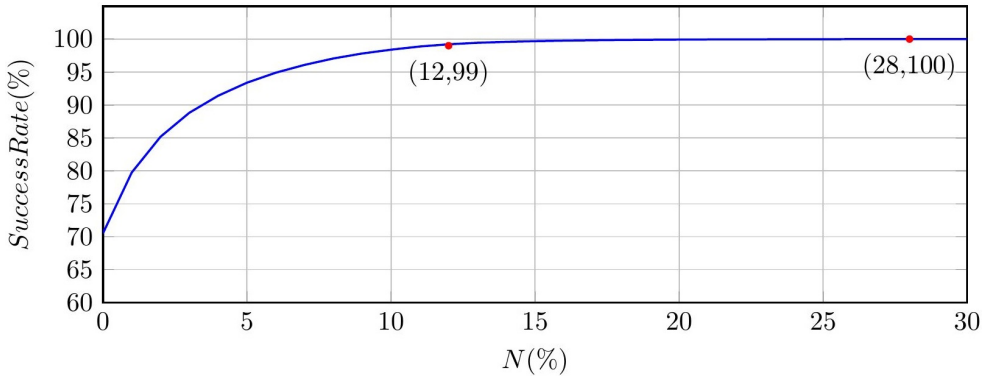


Figure 5.15. Success rate of method version 5 where sensor geometry is altered

CHAPTER 6

CONCLUSION

Rotary encoders are electromechanical devices and they are susceptible to mechanical imperfections such as radial run-out, where the shaft is deviated from its original rotation axis. Since a rotary encoder requires its shaft to be centered, radial run-out causes errors in position calculation. To minimize the problems incurred by run-out of the shaft, mechanical alignment hardware, such as high precision ball-bearings, are commonly utilized. However, installation of the encoder becomes more difficult as the mechanical precision increases, and additional hardware, such as flexible coupling are required to protect the system components. All of these precautions increase the cost, and furthermore, flexible couplings may introduce inaccuracy as a consequence of their functionality. This thesis offers a software method to eliminate position errors caused by the radial run-out of the shaft. The method utilizes additional sensors to determine the run-out parameters. Besides, the additional sensors are also employed in position calculations to increase the precision. These additional sensors are placed to the opposite locations of the two main sensors to obtain four quadrature signals instead of two. This sensor geometry allows the sensors to be placed in a single plane so that if the device is fabricated the assembly procedure will be easier and the cost will be lower. Lastly, the algorithm was designed keeping in mind that the angular position measurements should be provided in a timely manner. Therefore, instead of using an iterative approach, the proposed method was developed to produce the results within a deterministic response time.

All algorithms for simulation of encoder outputs under varying conditions and angular position correction methods were implemented in C language. The GNU C compiler for Windows operating system and the Code:Blocks open source development environment were used for programming. The algorithms were compiled as Dynamic Link Library (DLL) functions that are linked to a LabVIEW GUI for easy setting of function parameters and visualization of results.

By using the tools that are developed for simulation, a sensor field was generated over a 2D surface by using a sinusoidal angular variation function and a linear, sinusoidal or Gaussian radial variation function. The effect of the radial run-out was simulated by translating the field with the same amount and at the same direction of the shaft run-out. Calibration datasets for the sensor outputs were obtained from this field for correctly

centered shaft (center calibration data) and dislocated shaft (run-out calibration data) in distinct run-out configurations.

Position calculation and error correction algorithms were implemented by utilizing calibration datasets. First position estimate was calculated using the center calibration data whereas the other datasets were utilized to determine the run-out and apply correction accordingly. The correction method was evaluated in special cases as well as in large number of trials. To extract overall scores, 648,000 trials were carried out for all possible combinations of 360 angular positions, 360 run-out directions, and 5 run-out distances. The first version of correction algorithm used raw calibration data, and it was performing poorly due to error mismatch. The error mismatch occurred, because the error patterns obtained at the first position estimate do not resemble the error patterns as a function of true angular position. This issue was resolved by preparing an error map that correctly maps the sensor errors to the first position estimates. Using the error map yielded significantly better results compared to using raw calibration data in terms of run-out direction estimation accuracy and mean position error. However, it also introduced two new problems, that are singularities in error map and multiple minima in distance function.

At singular configurations, the error map showed no sensor error although there were position errors. As a result, the correct run-out direction could not be identified, and the correction attempts resulted in large errors. The singularities resulted from the symmetry in the system, and two solutions were implemented to overcome this problem. The first one was to add a second harmonic to the angular field variation. Although adding second harmonic provided sufficient sensor error at all angular positions and thus prevented singularities to occur, multiple minima in distance function was still jeopardizing the direction estimation accuracy. In the second solution, sensor geometry was slightly altered to break the symmetry instead of changing angular field variation. Consequently, it is observed that the errors associated with singularities and multiple minima were eliminated. Also, the robustness of the method was validated over the distance graph where minima points were correctly aligned for all angular positions. Changing the sensor geometry offers a more practical and easier solution than changing the radial field variation which requires a complex design for the rotating encoder wheel. Moreover, similar results were also obtained for cosine and Gaussian radial field distributions by changing the sensor geometry slightly to make use of relatively more linear regions of these distributions. These results demonstrate the ability of the algorithm to correct the errors for radial field variations other than linear functions. Lastly, overall success of the method was evaluated and it is found that the method can keep the error under 28% of the initial error at 100%

of the trials and under 12% at 99% of the trials.

We have presented a heuristic algorithm to correct the errors caused by radial run-out of the shaft of the encoders and successfully reduced the mean position error down to %12 of the initial value regardless of the radial field variation function. In addition, it can be deduced that the algorithm requires a set of preliminary conditions to work properly. The preliminary conditions given below are about the relationship between sensor error and position error, since correction algorithm rely on sensor errors to correct position errors.

1. Sensor errors should change as monotonic functions of run-out distance for any angular position and run-out direction so that they can be interpolated accurately in the acceptable run-out range.

2. Sensor errors should change smoothly as a function of run-out direction at any run-out distance for all angular positions, so that the run-out direction can be estimated correctly based on the calibration data. In other words, run-out direction steps in the calibration data should be small enough to allow accurate tracking of the sensor errors as a function of run-out direction.

3. At least one of the sensor outputs need to exhibit significant error calculated at the first angular position estimates for all run-out directions and all angular positions when there is significant sensor error. Otherwise, singularities will occur. Also, the error pattern of all sensors should not repeat more than once as the run-out direction varies between 0 and 360.

4. Error in angular position due to shaft movement should be small enough, so that all corrections can be done by utilizing the calibration data collected for the extended angular range (i.e. 110° instead of 90°) for each quadrant.

At this stage, the algorithm is considered as mature enough to be experimentally tested since reduction of position error is achieved consistently under all conditions. The future work involves testing the algorithm by developing special hardware to validate the outcomes of this research experimentally. The required angular and radial field variations can be obtained by using optic or magnetic sensors with simple code wheel designs. Further improvement of the correction methods is also possible by interpolating sensor data and errors over the run-out direction and by applying nonlinear interpolation techniques. The proposed method is very valuable in terms of eliminating mechanical components

since it can be adapted to more severe run-out scenarios, and therefore it has the potential to make encoders work without bearings and flexible couplings. Thus, correcting the errors due to radial run-out paves the way towards more reliable and more economical rotary encoders.

REFERENCES

Albrecht, C., J. Klöck, O. Martens, and W. Schumacher (2017). Online estimation and correction of systematic encoder line errors. *Machines 5*(1), 1.

Alejandre, I. and M. Artes (2004a). Machine tool errors caused by optical linear encoders. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 218*(1), 113–122.

Alejandre, I. and M. Artes (2004b). Real thermal coefficient in optical linear encoders. *Experimental Techniques 28*(4), 18–22.

Alejandre, I. and M. Artes (2006). Thermal non-linear behaviour in optical linear encoders. *International Journal of Machine Tools and Manufacture 46*(12-13), 1319–1325.

Alejandre, I. and M. Artés (2007). Method for the evaluation of optical encoders performance under vibration. *Precision engineering 31*(2), 114–121.

Baxter, L. K. (2000). Capacitive sensors. *Ann Arbor 1001*, 48109.

Birch, K. (1990). Optical fringe subdivision with nanometric accuracy. *Precision Engineering 12*(4), 195–198.

Burke, J., J. F. Moynihan, and K. Unterkofler (2000). Extraction of high resolution position information from sinusoidal encoders. In *Proceedings of the international intelligent motion conference*, pp. 217–222. Intertec International, Inc.

CUI Inc. (2017, 12). *AMT31 Series*. CUI Inc.

Dhar, V., A. Tickoo, S. Kaul, R. Koul, and B. Dubey (2009). Artificial neural network-based error compensation procedure for low-cost encoders. *Measurement Science and Technology 21*(1), 015112.

Ellin, A. and G. Dolsak (2008). The design and application of rotary encoders. *Sensor Review 28*(2), 150–158.

Heydemann, P. L. (1981). Determination and correction of quadrature fringe measurement errors in interferometers. *Applied optics 20*(19), 3382–3384.

Ireland, J. (2010, 01). Optical vs. magnetic: How to pick the right encoder engine. Technical report.

Karg, F. H. (1960, 11). Rotary potentiometer.

Lara, J. and A. Chandra (2014). Position error compensation in quadrature analog magnetic encoders through an iterative optimization algorithm. In *Industrial Electronics Society, IECON 2014-40th Annual Conference of the IEEE*, pp. 3043–3048. IEEE.

Le, H. T., H. Van Hoang, and J. W. Jeon (2008). Efficient method for correction and interpolation signal of magnetic encoders. In *Industrial Informatics, 2008. INDIN 2008. 6th IEEE International Conference on*, pp. 1383–1388. IEEE.

Lopez, J. and M. Artes (2012). A new methodology for vibration error compensation of optical encoders. *Sensors 12*(4), 4918–4933.

Lopez, J., M. Artes, and I. Alejandre (2011). Analysis of optical linear encoders' errors under vibration at different mounting conditions. *Measurement 44*(8), 1367–1380.

Lopez, J., M. Artes, and I. Alejandre (2012). Analysis under vibrations of optical linear encoders based on different scanning methods using an improved experimental approach. *Experimental techniques 36*(6), 35–47.

Mahn, C. (2010). Optimize the life and performance of rotary encoders through correct mounting. *Conference Record of 2010 Annual Pulp & Paper Industry Technical Conference*, 1–4.

Matsuzoe, Y., N. Tsuji, and T. Yoshizawa (2002). Error dispersion algorithms to improve angle precision for and encoder. *Optical Engineering 41*(9), 2282–2290.

Mayer, A. (1960, 5). Exponential potentiometer.

Sanchez-Brea, L. M. and T. Morlanes (2008). Metrological errors in optical encoders. *Measurement Science and Technology 19*(11), 115104.

Staebler, M. (1998). Tms320f240 dsp-solution for high-resolution position with sin/cos-encoders. *Application Report SPRA496, Texax Instruments Inc*.

Tan, K. K. and K.-Z. Tang (2005). Adaptive online correction and interpolation of quadrature encoder signals using radial basis functions. *IEEE Transactions on Control Systems Technology 13*(3), 370–377.

Tan, K. K., H. X. Zhou, and T. H. Lee (2002). New interpolation method for quadrature encoder signals. *IEEE Transactions on Instrumentation and Measurement 51*(5), 1073–1079.

Volder, J. E. (1959, Sept). The cordic trigonometric computing technique. *IRE Transactions on Electronic Computers EC-8*(3), 330–334.

Yang, Y., N. Rees, and T. Chuter (2002). Reduction of encoder measurement errors in ukirt telescope control system using a kalman filter. *IEEE transactions on control systems technology 10*(1), 149–157.

Zheng, D., S. Zhang, S. Wang, C. Hu, and X. Zhao (2015). A capacitive rotary encoder based on quadrature modulation and demodulation. *IEEE Trans. Instrumentation and Measurement 64*(1), 143–153.