Spring 2022

A Low-Cost and Low-Tech Solution to Test for Variations Between Multiple Offline Programming Software Packages.

Steffen Wendell Bolz

A LOW-COST AND LOW-TECH SOLUTION TO TEST FOR VARIATIONS BETWEEN MULTIPLE OFFLINE PROGRAMMING SOFTWARE PACKAGES FOR INDUSTRIAL ROBOTS

A Thesis
Presented to
The Faculty of the School of Engineering and Applied Sciences
Western Kentucky University
Bowling Green, Kentucky

In Partial Fulfillment
Of the Requirements for the Degree
Master of Science

By
Steffen Wendell Bolz

May 2022

A Low-Cost and Low-Tech Solution to Test for Variations Between Multiple Offline Programming Software Packages

04/06/2022

Date Recommended

Dr. Gregory K. Arbuckle, Director of Thesis

Dr. J. Michael Galloway

Dr. A. Mark Doggett

Ranjit T. Koodali 04/28/2022

Associate Provost for Research and Graduate Education Date

DEDICATION

To my mother.

&

To the faculty of Western Kentucky University.

ACKNOWLEDGMENTS

This is a product of the hard work and dedication of my mother, who homeschooled me and provided me with the foundation that I needed to succeed in collegiate and vocational applications.

CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ANSI	American National Standards Institute
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CMM	Coordinate Measuring Machine
CNC	Computer Numeric Control
CPS	Cyber Physical Systems
DT	Digital Twin
HRI	Human Robot Interaction
ISF	Incremental Sheet Forming
ISO	International Organization for Standardization
JIT	Just In Time
OLP	Offline Programming
ROI	Return On Investment
SME	Small and Medium-sized Enterprise
SPIF	Single-Point Incremental Forming
TCP	Tool Center Point
TOC	Theory Of Constraints
TP	Teach Pendant
VR	Virtual Reality
.LS	Long Listing Format
.TP	Teach Pendant Format

A LOW-COST AND LOW-TECH SOLUTION TO TEST FOR VARIATIONS BETWEEN MULTIPLE OFFLINE PROGRAMMING SOFTWARE PACKAGES FOR INDUSTRIAL ROBOTS

Steffen Wendell Bolz May, 2022 68 Pages

Directed by: Dr. Gregory K. Arbuckle, Dr. J. Michael Galloway, Dr. A. Mark Doggett

School of Engineering and Applied Sciences Western Kentucky University

This research paper chronicles the attempt to bring forth a low-cost and low-tech testing methodology whereby multiple offline programming (OLP) software packages' generated programs may be compared when run on industrial robots. This research was initiated by the discovery that no real research exists to test between iterations of OLP software packages and that most research for positional accuracy and/or repeatability on industrial robots is expensive and technologically intensive. Despite this, many countries' leaders are pushing for intensive digitalization of manufacturing and Small and Medium-sized Enterprises (SMEs) are noted to be lagging in adoption of such technologies. The research consisted of creating a test utilizing commonplace and inexpensive measuring devices in dial indicators to test the X, Y and Z axes of movement on a Fanuc R-2000iC/165F industrial robot. Unfortunately, the robot in question was unable to produce consistent results so that the research could be properly examined. It is assumed that the inconsistency could be linked to wear on the physical robot due to it having been utilized in heavy industrial work prior to being donated to the University where this research was conducted. Recommendations for future research and methods whereby the research could be refined are presented in the final chapter.

Chapter 1

INTRODUCTION

1.1 General Information

Robotics are flexible automation methods that manufacturers can create/process products whose process requirements are dangerous, physically-demanding and/or repetitive (Ross, 2018, p. 12). Robots can operate in environments that have temperatures and fumes dangerous to humans while not requiring the cumbersome gear a human would require to perform in such an environment. Additionally, robots can repeat tasks continuously, absent the physical and mental exhaustion that human workers would suffer under similar circumstances. However, upfront costs combined with the requirements for tech-savvy personnel who can properly utilize and maintain such robots, can become a barrier to some businesses introducing robots into their processes (Ross, 2018, p. 356-357, 360). This is despite the fact that such robotic implementations can result in lowered overall costs and higher flexibilities (Slavkovic, 2014, p. 2083). Additionally, more governments are pushing for higher automation and digitalization implementations within their nations' industries in the form of "Industry 4.0" (Germany: Industrie 4.0, 2017, p. 3).

In order to implement Industry 4.0 into businesses, one must consider the associated factors of digitalization and flexible automation. A common method of programming robots is to jog through the points that the robot must operate by a human programmer who pairs those points with typed programs. This method of programming, also known as Lead-

Through Programming, requires the robot to be physically used during programming, that prevents it from producing (Burghardt, 2020, p. 3). It also can require the programmer to program without being able to easily reference the robot path and is very time-consuming (Burghardt, 2020, p. 10). Alternatively, robots can be programmed "online" or "offline" where respectively the robot is programmed through a connected computer that is actively influencing the robot or where the computer is not directly influencing the robot (Ross, 2018, p. 74). Programming the robot offline through the use of a virtual "digital twin" can allow the programmer to reference paths without running the virtual robot program, check functionality of the program within the workspace and to generate programs relatively quickly (Burghardt, 2020, p. 10). (See Figure 1.1)

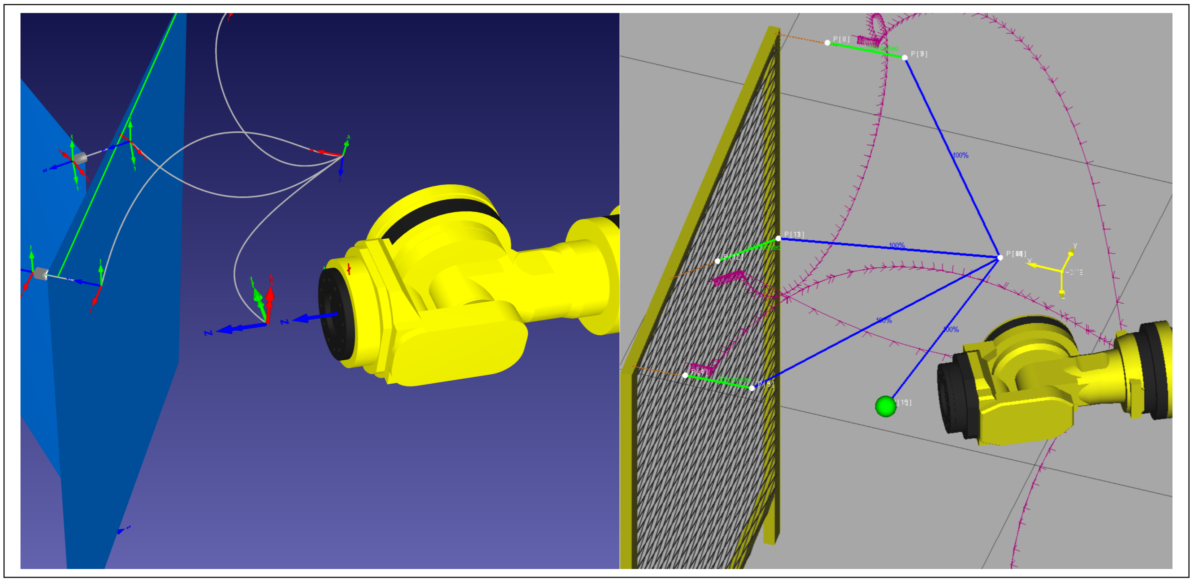


Figure 1.1: RoboDK and Roboguide virtual pathways.

The primary time investment is in the initial generation of the digital twin to replicate the physical workspace properly. This allows for a relatively easy generation of programs for the robot in a small amount of time. This OLP is facilitated through the use of robot OLP software that is made by the robot's manufacturer and/or third-parties. This method of OLP eliminates the need for a robot to be out of production while the programming occurs, thereby allowing Return On Investment (ROI) to continue. Additionally, OLP of this type can allow for higher satisfaction of the demands for digitalization by customers and governing bodies all while providing a higher level of productive utilization for such expensive equipment.

In order to match the increasing demands of greater flexibility and digitalization by governments, competitors and customers, robotics and virtual robotic programming are being viewed as an option for more flexible machining. Whether the robot is used machining, spraying, welding or for more advanced Incremental Sheet Forming (ISF), codes for such programs can be incredibly intricate and complex to generate by hand (Ross, 2018, p. 74). Instead, the use of Computer Aided Design/Computer Aided Manufacturing (CAD/CAM) programs such as FeatureCAM, a product within the Autodesk library, can aid in the generation of such programs (Autodesk, 2021). Unfortunately, FeatureCAM will not generate programs directly for a robot in a programming language that the robot understands. With this in mind, RoboDK, a product made for the offline programming of different robots made by different manufacturers, can make use of FeatureCAM and other CAM software, through a plug-in that translates machining paths into a robot path in the robot's language (RoboDK, 2021). This new code is generated into a text (.LS) file with the paths and points that can be uploaded into a robot controller for operation. However, many Fanuc robot controllers require that the code be uploaded in binary (.TP) format (FANUC, 2021b). In order to do so, one must either have a pricey plug-in for their controller (an RTL-R796-ASCII Program Loader) or one must load the program through a copy of Roboguide where the program may be converted to the proper .TP format (FANUC, 2021a). This option serves to translate the code while also allowing programmers to test and edit for possible errors and/or dangerous movements. This provides an excellent option to screen the robot program prior to uploading to the physical robot as Roboguide can diagnose many errors that may have slipped through during the RoboDK and FeatureCAM programming.

1.2 Research Goal

This research aimed to produce a low-cost, low-tech option for assessing whether or not program degradation had occurred between multiple iterations of offline programming (OLP) software. The experiment sought to examine whether or not the differences could be ameliorated through the programming software and/or through measured tolerances in the physical workspace. The research's goal was to enable companies to assess what margin of error exists and whether or not the programmers can adjust for those margins in their software or on their physical workspace. The usage of these programs and tools may allow manufacturers to have a larger range of flexible automation, at lower cost, while having a higher level of confidence in the accuracy between the various programs they may require. The tools utilized for this experiment included a Fanuc R-2000iC/165F Robot with its R-30ib controller, three dial indicator gauges (accurate to 0.001") and the software packages of Roboguide, RoboDK and FeatureCAM.

1.3 Statement of Purpose

The author found that current literature does not study the interactions of multiple OLP software packages and how they interact with one another. This research intended to determine whether or not such OLP software interactions would leave variations in the resulting robot program and how large those variations would potentially be. The research also endeavored to ascertain whether those variations were a result of human error in constructing the digital workspaces or through source code variations between the software packages. These goals were pursued by comparing three OLP software package's robot programs along the three axes of movement in order to ascertain how much variation occurred and whether the variation could be rectified.

1.4 Question of the Study

How can Small and Medium-sized Enterprises (SMEs) and other organizations measure their OLP software interactions in the deployed robot program? This research attempted to provide a first-step in the research of testing OLP software interaction variances and to provide a low-cost and low-tech solution for SMEs to initiate robotics with digital twins. The research was also an attempt to serve as a first-step for SMEs to commence Incremental Sheet Forming/Single-Point Incremental Forming (ISF/SPIF) operations within their organizations.

1.5 Statement of Need

Many organizations are hesitant to adopt new digitalization/automation methods due to cost and technological inhibitions (Ghobakloo, 2019, p. 2). This prevents these organizations from fully competing against companies that do not have such inhibitions. Moreover, these organizations are not in compliance with new initiatives provided by governing bodies and other organizations who are pushing for higher digitalization/automation and the benefits of this technology is thereby unavailable to affected SMEs (Ghobakloo, 2019, p. 1-2). This research attempted to remove these inhibitions by providing an affordable and easily-deployed solution whereby SMEs can remove the need to have costly measurement tools and costly technological experts to measure. Finally, there is still no research devised that examines whether there are variations in robot programs that have been developed across multiple OLP software packages. This is despite the fact that ISF/SPIF often requires such software collaboration at this early stage of the technology's development.

1.6 Limitations

This research was conducted by a student attending a publicly funded university and as such, did not have access to some of the higher cost/tech equipment available to other researchers. With this in mind, the results of the data collected cannot be compared to some other measuring research options, as such equipment far exceeded the budgetary limitations of the project. Additionally, this research was conducted utilizing a Fanuc robot and other robots may not have similar OLP software packages offered for them and/or such OLP software packages may not have similar coding flexibility whereby the experiment can be replicated.

1.7 Delimitations

Attempts to replicate this research should be equipped with a computer with FeatureCAM, RoboDK, Roboguide, a Fanuc R-2000iC/165F robot with its R-30ib controller and three dial indicators (accurate to 0.001"). Moreover, the replication of the research must be conducted by an individual who can utilize these tools and software packages in a safe and capable manner.

1.8 Definition of Terms

The terms below were used within this research paper and are defined here for the reader's ease.

1. ANSI - The American National Standards Institute is a private and non-profit

standards organization that administrates voluntary standards to promote American industry (ANSI, 2022). ANSI cooperates as the sole US representative and full member of the ISO (ANSI, 2022).

2. CAD/CAM - Computer Aided Design/Computer Aided Manufacturing is the utilization of computer systems to create potential products. The output of such software packages are electronic files that can print designs and/or produce machining programs. CAD is utilized primarily in producing the design of the product while CAM focuses on the method that it will be manufactured (Deans, 2021).

3. CMM - A Coordinate Measuring Machine is a machine that generates high accuracy measurements. Because of this, they are often utilized in research as the measurements are also highly traceable. The referenced CMM in this research utilizes optical measurements of a miniature robot placed within the CMM (Vuola, 2012, 126-128).

4. CNC - Computer Numeric Control machines are automated manufacturing machines that are guided by a computer-generated numeric code. This code guides the machine through specific movements, at specific speeds and with specific input data controlling coolant flow and other machine operations (Deans, 2021).

5. Corner-cases - Corner-cases is a term used to describe "... execution paths that are not part of the core functionality of a system." (Banabic, 2015, p. 2). It is often used to describe situations that compromise a system in a manner not foreseen by system designers. An example would include debris flying into the sensor of an autonomous industrial robot (Afzal, 2020, p. 4).

6. CPS - Cyber Physical Systems is a term used to describe systems that have virtual and physical components. Robotics is a classification within CPS but the term encapsulates

a broader scope including items like networking systems and power grids (Afzal, 2020, p. 1).

7. DT - A Digital Twin is a virtual replica of a physical entity. A DT must correspond to the layout and behavior of the physical entity and present them in an intuitive and comprehensive manner (Zhuang, 2021, p. 1-3).

8. Fanuc - Fanuc is a major manufacturer of robots based in Japan. Fanuc manufactures a wide range of robots and also publishes the OLP software package of Roboguide a programmer can utilize to program Fanuc robots (FANUC, 2021a).

9. FeatureCAM - FeatureCAM is a product from the Autodesk software suite. FeatureCAM specifically is used to produce CNC programs to manufacture parts through CNC machines. FeatureCAM can also be employed to produce robot machining programs through the use of a plugin option within the RoboDK software package (Autodesk, 2021).

10. Gauge RR - A Gauge Repeatability and Reproducibility study is used to remove potential errors from the usage of precision gauges in measurements. It is used to ensure that the gauges are functioning properly and that the inspectors of the gauges are properly performing the inspections (Pan, 2006, p. 501). It is not required to analyze reproducibility when there is only one operator performing the analysis (Austin, 2014, p. 30).

11. HRI - Human Robot Interaction is an intuitive term to define the collaboration and closeness between humans and robots that are operating in semiautonomous or autonomous operation (Choi, 2020, p. 3). Such interactions can range from workers in a factory working alongside industrial robots to individuals using or around self-driving cars (Choi, 2020, p. 3).

12. Industry 4.0 - Industry 4.0 is a term that defines the push to increase digi-

talization, automation and the use of Cyber Physical Systems (CPS) in modern industry (Ghobakhloo, 2019, p. 2). Industry 4.0 was first introduced as a concept at the 2011 Hannover Fair and has brought about other initiatives in China, North America and South Korea (Ghobakhloo, 2019, p. 2). This is through their "Made in China 2025", "Industrial Internet" and "Manufacturing Industry Innovation 3.0" initiatives respectively (Ghobakhloo, 2019, p. 2).

13. ISF - Incremental Sheet Forming is an innovative process whereby shapes are produced through localized plastic deformation through the use of a hemispherical tool (Bahloul, 2014, p. 163). ISF utilizes different methodologies from using CNC machines or robots, partial or full dies and/or singular or multiple points of contact (Bahloul, 2014, p. 164). ISF is an umbrella that various sheet forming methodologies reside under and included within ISF are Single Point Incremental Forming (SPIF) and Roboforming.

14. ISO - The International Organization for Standardization is a global federation of standards organizations (ANSI, 2022). Full members of the ISO can participate in the formulation of standards and the sale and adoption of those standards (ANSI, 2022).

15. ISO 9283:1998 - The International Organization for Standardization standard for the methodology of specifying and testing certain performance characteristics of manipulating industrial robots (ANSI, 2021). This standard was published in 1998 and has not been updated since (Bi, 2020, p. 1).

16. Jidoka - Jidoka is defined by the Japanese auto manufacturer Toyota to mean "automation with a human mind" (Dennis, 2017, p. 123). The objective of Jidoka is to prevent defects from flowing through a process as a result of human errors that will occur

(Dennis, 2017, p. 126-129). This is often accomplished through the use of "poka-yokes" (Dennis, 2017, p. 127-128).

17. JIT - Just In Time production methodology defines producing the correct item(s), in the correct quantities at the correct time (Dennis, 2017, p. 89). JIT is comprised of of the key elements of flow, pull, standard work and takt time (Womack, 2003, p. 349).

18. Kaikaku - Kaikaku is a Japanese term that translates loosely to "radical improvement" (Womack, 2003, p. 23). Overall, the term defines the "... radical improvement of an activity to eliminate muda (waste) ..." (Womack, 2003, p. 349).

19. Kanbans - Kanbans are visual tools that are utilized to enable JIT in production (Dennis, 2017, p. 96). Kanbans are signals that stand as an authorization to produce or withdraw within a system (Dennis, 2017, p. 96).

20. Karel - Karel is a robot programming language used by Fanuc robots and was developed in 1981 (Ross, 2018, p. 75).

21. Monuments - Monuments are tools that require designs, order and products to accumulate waiting wastes in queue prior to moving through the monument (Womack, 2003, p. 350). These are contrasted with "right-sized" tools that are more flexible than monuments in dealing with production and are typically more simple, less automated and slower (Womack, 2003, p. 60).

22. OLP - Offline Programming is a method of robot programming that utilizes software off of the physical robot. This process allows for robot programs to be generated without taking the robot out of production for the programming process. Such software can

additionally allow the programmer to troubleshoot programs, reference robot paths/movements and increase turnover of robot programs (Burghardt, 2020, p. 10).

23. Poka-Yokes - Poka-Yokes are devices or procedures that are designed to prevent defects in a system (Womack, 2003, p. 350). Poka-Yokes should be simple, reliable, low cost and designed for the environments where they are used (Dennis, 2017, p. 128).

24. Right-sized - Right-sized is a term within lean methodology that descrbibes "A design, scheduling or production device that can be fitted directly into the flow of products within a product family so that production no longer requires unnecessary transport and waiting." (Womack, 2003, p. 351).

25. RoboDK - RoboDK is a software package offered by the company of the same name. It can be used to program many different robots from many different manufacturers through the usage of post-processors that translate the RoboDK program code into the language used by the robot. It also has plugin options with other OLP software packages such as FeatureCAM (RoboDK, 2021).

26. Roboforming - Roboforming is a process whereby one can perform ISF with a robot as opposed to a CNC machine (Bârsan, 2021, p. 3). Roboforming is advantageous through its relative felxibility and distribution, however, it does lose the stiffness of CNC machines due to a robot's articulated joints (Bârsan, 2021, p. 3).

27. Roboguide - Roboguide is an OLP software package offered by Fanuc that can be used to program the various robots they produce. The software can upload programs directly to the robot or the programs can be transferred with a flash drive (FANUC, 2021b).

28. ROI - Return On Investment is a financial term that defines the formula of (Benefits-Costs)/Costs (Erdogmus, 2004, p. 19). In layman's terms, it defines what one

should expect in return from what they have put in. If one garners little from what they have put in, the ROI is low and if they garner much, the ROI is high.

29. SME - Small and Medium-sized Enterprises is an intuitive term for enterprises that are typically privately-owned and of smaller size relative to more developed and/or affluent competitors (Łobos, 2020, p. 115). SMEs are often characterized by their more incremental development that is done out of need for increased operations, greater efficiency in those operations and/or less costly of operations (Łobos, 2020, p. 122).

30. SPIF - Single-Point Incremental Forming is an ISF method that utilizes one point of force to introduce the incremental forming of the sheet (Bahloul, 2014, p. 164). In this process, the sheet is clamped peripherally and a hemispherical tool introduces plastic deformation to the sheet without the use of a die (Bahloul, 2014, p. 164). This is contrasted with Two-Point Incremental Forming (TPIF) that uses a partial or full die on the other side of the sheet from the tool (Bahloul, 2014, p. 164).

31. TCP - Tool Center Point is where a robot's end effector will first be able to contact a theoretical target. In a roboforming environment, this would be the tip of the hemispherical tool that will press against the material. On a typical robot hand, this would be the central point between the grippers. TCP is often used when referring to a tool's coordinates in relation to the robot's base frame (Garbev, 2020, p. 4), (Vocetka, 2020, p. 1).

32. Teach Pendant - Teach pendants are used to program robots through teach pendant programming (Ross, 2018, p. 23). These are handheld devices that can input instructions into the robot and record points of movement (Ross, 2018, p. 23-24). They are also referred to as teach boxes or handheld programmers (Ross, 2018, p. 23).

33. Theory Of Constraints - The Theory Of Constraints (TOC) is a production methodology that was developed by Eliyahu M. Goldratt. The methodology utilizes "five focusing steps" to target a system's problems by identifying the system's constraint, deciding how to exploit the constraint, subordinating everything else to the prior decisions, elevating the system's constraint and if the constraint breaks in the prior steps, return to the first step while avoiding inertia to cause a new constraint (Goldratt, 2016, p. 363).

34. VR - Virtual Reality is a tool whereby one can visually experience an environment as though it were real (Ustundag, 2018, p. 20).

1.9 Assumptions

This research assumed that the robot and measuring tools have such limited variation within them that the experiment could be conducted successfully. This research also assumed that the Fanuc R-2000iC/165F robot had repeatability accuracy to +- 0.00787402". This assumption was researched by conducting a Gauge Repeatability and Reproducibility (Gauge RR) study to verify that the robot could maintain the level of accuracy required by the study. Also, the dial indicators are rated to measure to 0.001" by their manufacturers. This was verified through a Gauge RR study that ensured that the dial indicators were functioning properly. This research also assumes that the replicators of this experiment will be properly trained and equipped to replicate this research. Finally, this research assumes that replicators will have all equipment installed and ready to perform the experiment in conjunction with a facilitating workspace where the three axes of movement may be measured.

Chapter 2

BACKGROUND AND LITERATURE

Robotics as tools have high initial costs without a guaranteed payoff, and the failures of other companies to successfully implement robots in their systems can serve as deterrents to potential robot users (Ross, 2018, p. 356). High costs are also associated with additional tooling and equipment that a company would require their robots to have in order to work with the company's current and future state production lines (Ross, 2018, p. 360). These issues don't even begin to touch upon the elements of organized and/or unorganized labor opposition to robots (Ross, 2018, p. 360). Finally, the use of robots fluctuates wildly dependent upon region and/or industry (Ross, 2018, p. 357). This can leave some companies feeling lost as to when to begin using robots or even how to use them.

However, the push for robotics is a reality. Some governments of large countries, such as Germany, have been applying pressure on companies to expand digitalization and automation (Germany: Industrie 4.0, 2017). Smaller countries are also applying pressure, such as the Competitiveness of Enterprises and Small and Medium-sized Enterprises (COSME) associated nations of Turkey, Iceland and Serbia (Industry 4.0 initiatives in the COSME countries, 2017). These pressures combined with international competition can be major drivers for companies to adopt robotics. Robotics can also offer some advantages depending upon the manner of implementation. As an example, the utilization of robots in industrial applications can increase product quality due to accuracy and repeatability under certain conditions (Ross, 2018, p. 359). Heavy tooling that requires strength combined with precision can be performed repeatedly by robots while humans operating such tooling may struggle to replicate the same results over long shifts. Robots also do not require the rest and/or mental stimulation that human workers do, these factors make robots a possible solution for performing those tasks (Ross, 2018, p. 359). These facts allow companies to assign monotonous and repetitive tasks to robots and also have those robots perform those tasks for long periods of time without stopping. Finally, working conditions that humans cannot or will not work under can still be suitable for robots to operate to full capacity (Ross, 2018, p. 357-359). Robotic solutions deployed in these areas can act to promote worker safety while still performing the required work.

2.1 Background

2.1.1 The Industry 4.0 Concept

Industry 4.0 is a term that was first coined by the German government in 2011 at the Hannover Fair as a definition of what some regard to be the fourth industrial revolution (Ustundag, 2018, p. 4). One of the lead goals and benefits of Industry 4.0 is the promotion of responsiveness and flexibility of given systems that adds to their ability to manage short-run product cycles and/or prototypes (Pagliosa, 2019, p. 545). Industry 4.0 is a concept that encapsulates the combination of traditional manufacturing with modern, digitized practices in industry to allow greater flexibility and control over the value stream for a product/service lifecycle (Ghobakhloo, 2019, p. 3). Industry 4.0 can also be defined as an amalgamation of technologies and practices that interconnect people, equipment and their products throughout a value stream (Pagliosa, 2019, p. 544). Moreover, Industry 4.0 can be defined as the digitalization of physical assets and the mass integration of value stream stakeholders (Pagliosa, 2019, p. 544). As one might have guessed by now, the precise definition of Industry 4.0 still remains elusive despite the large amount of scholarly research into the subject (Ghobakhloo, 2019, p. 3). With the boundaries of Industry 4.0 being rather undefined, individual elements of the concept also lack definition. However, certain elements such as adaptive robots, autonomous robots, cyber physical systems (inclusive of digital twins), simulation, 3d visualization, Virtual Reality (VR) integration and others are all considered elements within the Industry 4.0 concept (Ustundag, 2018, p. 6).

These concepts may seem inconsistent with other non-traditional methodologies such as lean, that focuses upon the elimination of waste, improving productivity and quality, while focusing upon the customer's desires to define value (Pagliosa, 2019, p. 544). Lean often focuses on "right-sized" and flexible equipment (Womack, 2003, p. 60). The result of this thinking means that "monuments", lean's definition for pieces of equipment that are low-flexibility, bulky, automated and "sophisticated", end up being dumped for more flexible and lower-tech solutions (Womack, 2003, p. 176, 178). Similar process improvement methodologies such as the Theory Of Constraints (TOC), emphasize that elements of automation (specifically robots) can create additional expense and additional inventory without creating additional throughput (Goldratt, 2016, p. 27-28). The improper deployment of these automation mechanisms fails to further a business towards "The Goal" of TOC whereby the important measures of throughput, Return On Investment (ROI) and inventory reduction, are often left in a worse state than prior (Goldratt, 2016, p. 59-60).

As a result, one may question whether or not these people and practice-based innovations are compatible with the high-tech equipment-based innovations of Industry 4.0 (Pagliosa, 2019, p. 544). In truth, both concept types share the goals of waste elimination, seeking improvements in production and quality while focusing upon customer desires for the products they purchase (Pagliosa, 2019, p. 544). Another example is the lean concept of “Kaikaku” (Japanese expression for “radical improvement”) that promotes the adoption of technological innovation to dramatically reorient the value stream (Womack, 2003, p. 23, 27). Also, the lean concept of “Jidoka” (automation with a human mind/autonomation) is a pillar of the “House of Lean Production” (Dennis, 2017, p. 26). Jidoka’s aims include “poka-yokes” (fool-proofs) and other concepts centered around human interaction with equipment to prevent defects from being produced and/or flowing downstream while ensuring errors are reported to discover solutions (Dennis, 2017, p. 123, 127). These goals coincide with Industry 4.0 having interconnectivity between automation cells, operators and the products being produced so that responsiveness is raised within the system (Pagliosa, 2019, p. 545). Industry 4.0 interconnectivity can also aid in the promotion of Just In Time (JIT) concepts whereby information flows of the physical flows can be more easily and more broadly disseminated leading to more information-based planning being made possible (Ghobakhloo, 2019, p. 7-8). These elements streamline formerly complex and time-consuming analysis work and allows for information-based decision making through the swift information sharing processes (Ghobakhloo, 2019, p. 8). JIT is the second pillar in the house of lean and the interconnectivity of automation in Industry 4.0 provides important “Kanbans” (signals to commence or cease production) within the system (Dennis, 2017, p. 26). In truth, Industry 4.0 concepts are compatible with process improvement strategies and can potentially serve as components of those improvement strategies based on the deployment of those concepts.

However, cost is still a factor to be considered and can be significantly more obstructive for Small and Medium-sized Enterprises (SMEs) (Sanders, 2016, p. 813). While larger, more affluent and more experienced manufacturers have greater flexibility to adapt, SMEs may find the cost and technological requirements daunting (Ghobakloo, 2019, p. 2). Low SME adoption is reflected in the fact that a mere 5.5 percent of German SMEs have the infrastructure needed to implement Industry 4.0 concepts (Ghobakloo, 2019, p. 2). Despite awareness of the added benefits of successful implementation, SMEs were found to be more hesitant to do so due to the youth of the concepts, the costs, the technological requirements and whether the implementation could aid them specifically in creating value (Ghobakloo, 2019, p. 5). Steps should be taken to attempt to incrementally reduce the technical and fiscal demands of various elements within Industry 4.0 in order to further break down the barriers between such SMEs and the benefits of Industry 4.0. A small part of cost and technology concerns was attempted to be addressed by this research in hopes of empowering more SMEs in their implementation of robotics and other Industry 4.0 concepts.

2.1.2 Incremental Sheet Forming (ISF) and Single-Point Incremental Forming (SPIF) through Roboforming

Incremental Sheet Forming (ISF) refers to the process of introducing localized plastic deformation to a sheet of material through the use of a hemispherical tool (Bahloul, 2014, p. 163). The tool follows incremental contours whereupon completion, the next contour is traced by the tool at an incremental depth (Bahloul, 2014, p. 164). Upon completion of these steps, the final geometry of the product will be rendered in the deformed material. ISF methods are differentiated from one another based upon whether a die is used (partial or full) and how many points of contact are employed (Bahloul, 2014, p. 164). Single Point

Incremental Forming (SPIF) specifically refers to when a singular contact point (the tool) is used to cause the deformations of the sheet (Bahloul, 2014, p. 164). This innovation's main factors are the size of the vertical incremental steps and the diameter of the tool being used (Bârsan, 2021, p. 3). "Roboforming" refers to the process of performing ISF/SPIF with a robot as opposed to a Computer Numeric Control (CNC) milling machine (Bârsan, 2021, p. 1). This alternative can be cheaper and more flexible than a CNC while sacrificing the stiffness of a CNC due to a robot's serial structure (Bârsan, 2021, p. 1, 6). Roboforming can be a solution whereby a company may reduce costs of prototyped products and low-volume productions (Bârsan, 2021, p. 3). ISF/SPIF can aid immensely in product development efforts that are demanded by lean to be performed in significantly lower amounts of time than traditional methods (Womack, 2003, p. 140-141).

2.1.3 Computer Aided Design/Computer Aided Manufacturing (CAD/CAM) in Roboforming

Robots can be used in a wide variety of manufacturing applications; some applications are more demanding than others with more intricate robot paths. Machining operations, like milling and SPIF, can have thousands of points that are not reasonably generated by hand. The need for incrementally altered points given each iteration of a product's contour and the incremental vertical shifting of those contours, require significant amounts of points and linking code before even considering factors like tool speed (Bahloul, 2014, p. 164). As a result, the utilization of CAD/CAM software to generate and simulate such toolpaths is especially helpful in order to increase accuracy in Roboforming (Bârsan, 2021, p. 6). CAD/CAM products, such as FeatureCAM, provide users a level of control over the mentioned factors of incremental depth and tool speed. FeatureCAM also allows users to indicate what size tool is being used that in turn allows the user to have greater control over the contour increments of the machining program. Computing these various factors in hand-written programs leaves many opportunities for mistakes and would be immensely time-consuming to generate.

2.1.4 Digital Twins

A “Digital Twin” (DT) “can be considered the most intuitive and comprehensive digital description of the internal and external attributes for a corresponding physical entity” (Zhuang, 2021, p. 3). In other words, a DT will exemplify the attributes and behaviors of a physical entity and present them through a digital output in a manner that is easily understood and interacted with. The goal of utilizing a DT is to assist in decision-making, optimization and the tracking of all processes in the physical counterpart so as to create a “... multi-dimensional dynamic, fusion and global model” (Zhuang, 2021, p. 3). A DT is required to characterize five things: realness, integration, dynamicity, visibility, and computability (Zhuang, 2021, p. 3).

Realness ensures that the DT exemplifies the physical counterpart’s overall shapes, dimensions, textures and behaviors (Zhuang, 2021, p. 3). Realness is exemplified in FeatureCAM, RoboDK and Roboguide through the 3-dimensional representations of the robots, materials and tools being used. The limitations of movement of the robot/tooling models and the methods of controlling that movement through the software are integrated, reflecting realism. (See Figure 2.1)

Integration corresponds to the DT’s ability to holistically reflect the physical counterpart’s overall characteristics and behavior in conjunction with all of its various inputs and outputs (Zhuang, 2021, p. 3). In FeatureCAM, RoboDK and Roboguide, integration

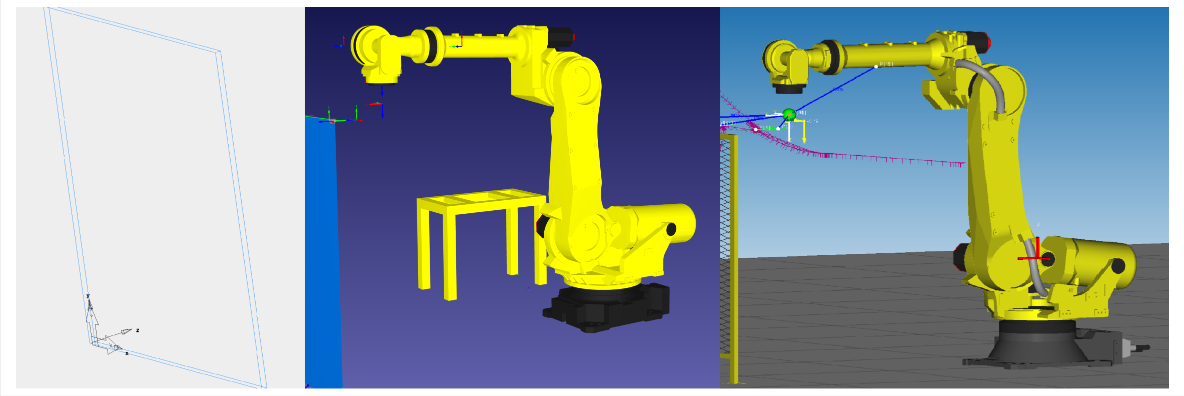
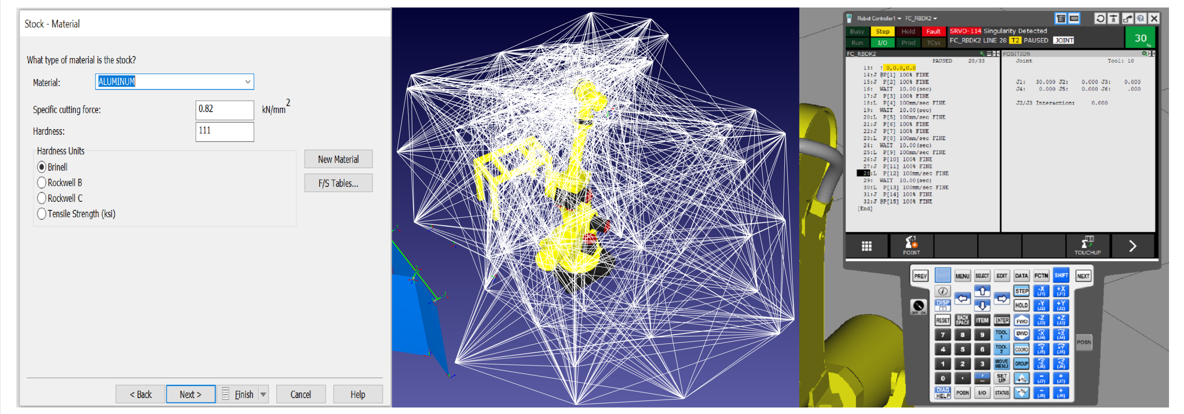


Figure 2.1: Realness is shown in the physical models, scaling and the reflection of real-world values like the Cartesian Coordinate system.

is seen in the need to dictate what materials are to be utilized such as in FeatureCAM's software. Also, how a robot can encounter errors in the robot program and display them back to the user as is performed by Roboguide's in-software Teach Pendant (TP). Finally, the collision-free motion planner option in RoboDK aids in integrating the workspace fully into the robot program. Such software features serve as poka-yokes to aid in the disposal of program bugs, code failures and erroneous movements prior to actually uploading to the physical robot on the line (Dennis, 2017, p. 127-128). (See Figure 2.2)



Stock - Material

What type of material is the stock?

Material: ALUMINUM

Specific cutting force: 0.82 kN/mm^2

Hardness: 111

Hardness Units

● Brinell
○ Rockwell B
○ Rockwell C
○ Tensile Strength (ksi)

SRVO-114 Singularity Detected
FC_RBDK2 LINE 28 T2 PAUSED JOINT

```
FC_RBDK2
                         PAUSED     28/33
   13:  ! 0,0.0,0.0
   14:J @P[1] 100% FINE
   15:J  P[2] 100% FINE
   16:  WAIT  10.00(sec)
   17:J  P[3] 100% FINE
   18:L  P[4] 100mm/sec FINE
   19:  WAIT  10.00(sec)
   20:L  P[5] 100mm/sec FINE
   21:J  P[6] 100% FINE
   22:J  P[7] 100% FINE
   23:L  P[8] 100mm/sec FINE
   24:  WAIT  10.00(sec)
   25:L  P[9] 100mm/sec FINE
   26:J  P[10] 100% FINE
   27:J  P[11] 100% FINE
   28:L  P[12] 100mm/sec FINE
   29:  WAIT  10.00(sec)
   30:L  P[13] 100mm/sec FINE
   31:J  P[14] 100% FINE
   32:J @P[15] 100% FINE
  [End]
```

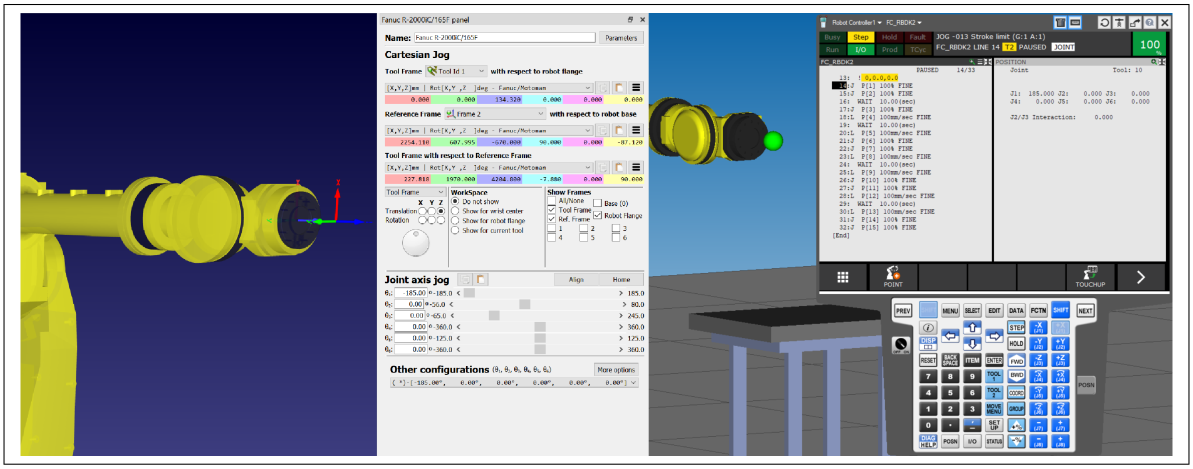
```
POSITION
   Joint                       Tool: 10

   J1:   30.000 J2:    0.000 J3:    0.000
   J4:    0.000 J5:    0.000 J6:     .000

   J2/J3 Interaction:     0.000
```

Figure 2.2: Material inputs with FeatureCAM help the software determine speeds and feeds. RoboDK helps reveal possible collisions within the DT workspace and Roboguide's TP is the same as the physical TP.

Dynamicity refers to the fact that DTs are not static design models and instead correspond with the evolution of the physical counterpart, simulate the physical counterpart's behavior and can be used to show the physical counterpart's behavior in advance (Zhuang, 2021, p. 3). Dynamicity is reflected in the previously mentioned software packages in their ability to display the programs developed for them in action, the errors/collisions that could occur and while also reflecting the outcomes of the actions that have occurred. RoboDk and Roboguide also offer dynamicity through ensuring the joint rotation limits of the actual robots are reflected in the virtual robots, thereby preventing the DT from allowing the programmer to code movements the robot cannot perform. (See Figure 2.3)



Name: Fanuc R-2000iC/165F

Tool Frame: Tool Id 1 with respect to robot flange

[X,Y,Z]mm | Rot[X,Y ,Z]deg - Fanuc/Motoman					
0.000	0.000	134.320	0.000	0.000	0.000

Reference Frame: Frame 2 with respect to robot base

[X,Y,Z]mm | Rot[X,Y ,Z]deg - Fanuc/Motoman					
2254.110	607.995	-670.000	90.000	0.000	-87.120

Tool Frame with respect to Reference Frame

[X,Y,Z]mm | Rot[X,Y ,Z]deg - Fanuc/Motoman					
227.818	1970.000	4204.800	-7.880	0.000	90.000

Tool Frame

	X	Y	Z
Translation	○	○	●
Rotation	○	○	○

WorkSpace
- ● Do not show
- ○ Show for wrist center
- ○ Show for robot flange
- ○ Show for current tool

Show Frames
- ☐ All/None
- ☑ Tool Frame
- ☑ Ref. Frame
- ☐ Base (0)
- ☑ Robot Flange
- ☐ 1 ☐ 2 ☐ 3
- ☐ 4 ☐ 5 ☐ 6

	Value	Min	Max
θ_1:	-185.00	-185.0	185.0
θ_2:	0.00	-56.0	80.0
θ_3:	0.00	-65.0	245.0
θ_4:	0.00	-360.0	360.0
θ_5:	0.00	-125.0	125.0
θ_6:	0.00	-360.0	360.0

Other configurations (θ_1, θ_2, θ_3, θ_4, θ_5, θ_6)

(*)-[-185.00°, 0.00°, 0.00°, 0.00°, 0.00°, 0.00°]

```
JOG -013 Stroke limit (G:1 A:1)
FC_RBDK2 LINE 14 T2 PAUSED JOINT
```

```
FC_RBDK2
                         PAUSED     14/33
   13:  ! 0,0.0,0.0
   14:J  P[1] 100% FINE
   15:J  P[2] 100% FINE
   16:  WAIT  10.00(sec)
   17:J  P[3] 100% FINE
   18:L  P[4] 100mm/sec FINE
   19:  WAIT  10.00(sec)
   20:L  P[5] 100mm/sec FINE
   21:J  P[6] 100% FINE
   22:J  P[7] 100% FINE
   23:L  P[8] 100mm/sec FINE
   24:  WAIT  10.00(sec)
   25:L  P[9] 100mm/sec FINE
   26:J  P[10] 100% FINE
   27:J  P[11] 100% FINE
   28:L  P[12] 100mm/sec FINE
   29:  WAIT  10.00(sec)
   30:L  P[13] 100mm/sec FINE
   31:J  P[14] 100% FINE
   32:J  P[15] 100% FINE
 [End]
```

```
POSITION
   Joint                          Tool: 10

   J1:  185.000 J2:    0.000 J3:    0.000
   J4:    0.000 J5:    0.000 J6:    0.000

   J2/J3 Interaction:     0.000
```

Figure 2.3: RoboDK's robot panel and Roboguide's virtual Teach Pendant show that Joint 1 is at full rotation to the respective negative and positive rotational values.

Visibility refers to a DT's ability to act as a dynamic monitor of the physical counterpart's condition and behavior in a 3d representation (Zhuang, 2021, p. 3). FeatureCAM accomplishes visibility by showing the dimensioned materials and how they appear post-operation. RoboDK and Roboguide display the total movement of the robot programs in real-time while also showing line paths of the Tool Center Point (TCP) in the workspace. (See Figure 2.4)

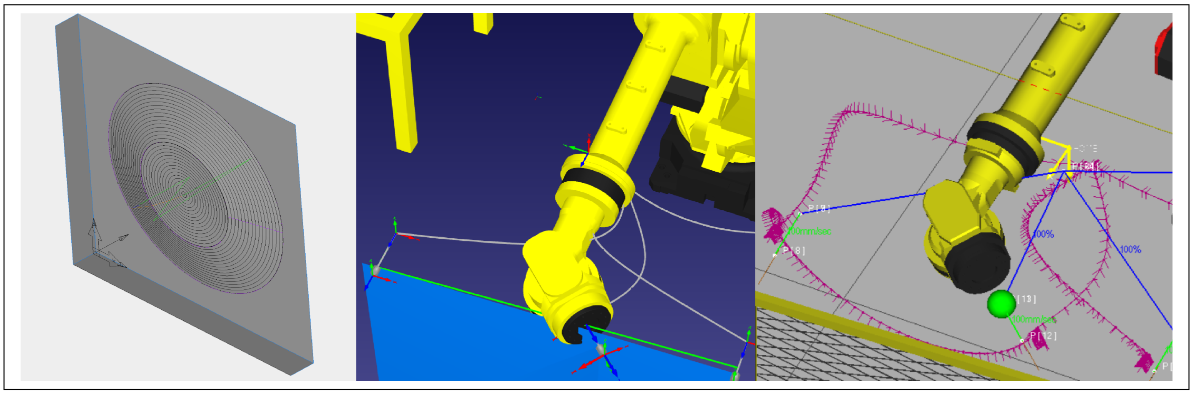


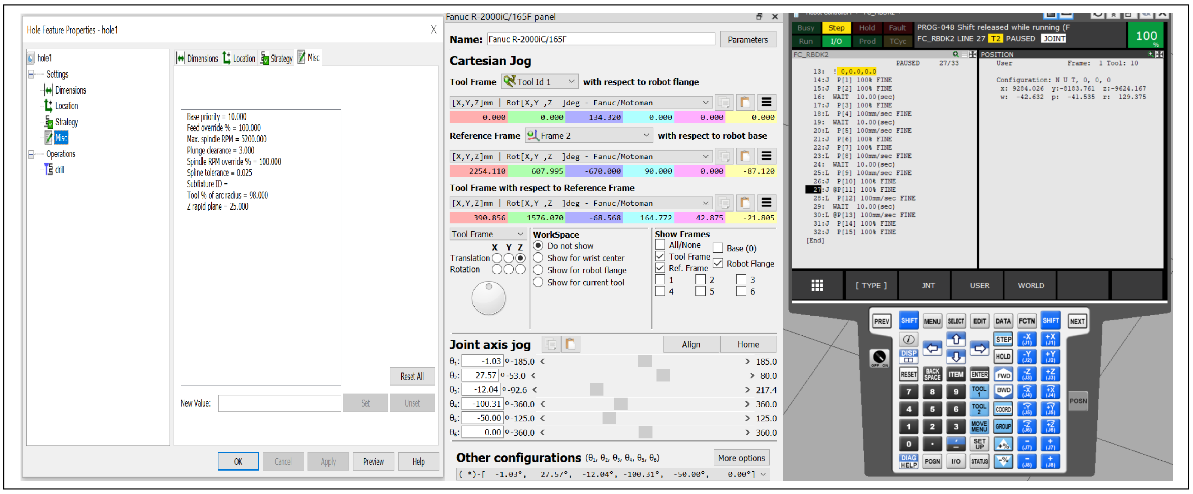
Figure 2.4: Pathway lines, formed parts and visual representations of the programs' actions are all reflections of the concept of visibility.

Computability refers to the ability to use a DT to predict future behavior based upon the data put out by a DT (Zhuang, 2021, p. 3). All of the already mentioned programs have outputs of data collected from running the simulations in the programs. This data can be locational data in Cartesian coordinates or joint rotational degree values. FeatureCAM puts out data throughout the design phase as it recommends speed and feed data, cutter types and cutting methods. All of these data outputs reflect the computability of how the physical workspace and DT should behave. (See Figure 2.5)

2.2 Related Work

2.2.1 Gauge Repeatability and Reproducibility

In order to properly ascertain the success or failure of a process/system, one must be able to measure it in order to ensure that modifications have had effect for good or ill. Absent an adequate measurement system, there is little chance of ensuring that the process/system is valid, repeatable or will render usable results (Pan, 2006, p. 500). One



Base priority = 10.000
Feed override % = 100.000
Max. spindle RPM = 5200.000
Plunge clearance = 3.000
Spindle RPM override % = 100.000
Spline tolerance = 0.025
Subfixture ID =
Tool % of arc radius = 98.000
Z rapid plane = 25.000

Name: Fanuc R-2000iC/165F

Tool Frame: Tool Id 1 with respect to robot flange

[X,Y,Z]mm | Rot[X,Y ,Z]deg - Fanuc/Motoman					
0.000	0.000	134.320	0.000	0.000	0.000

Reference Frame: Frame 2 with respect to robot base

[X,Y,Z]mm | Rot[X,Y ,Z]deg - Fanuc/Motoman					
2254.110	607.995	-670.000	90.000	0.000	-87.120

Tool Frame with respect to Reference Frame

[X,Y,Z]mm | Rot[X,Y ,Z]deg - Fanuc/Motoman					
390.856	1576.070	-68.568	164.772	42.875	-21.805

Tool Frame: Translation X ○ Y ○ Z ●; Rotation X ○ Y ○ Z ○

WorkSpace
● Do not show
○ Show for wrist center
○ Show for robot flange
○ Show for current tool

Show Frames
☐ All/None
☑ Tool Frame
☑ Ref. Frame
☐ Base (0)
☑ Robot Flange
☐ 1 ☐ 2 ☐ 3
☐ 4 ☐ 5 ☐ 6

Joint axis jog

θ_1:	-1.03	-185.0	185.0
θ_2:	27.57	-53.0	80.0
θ_3:	-12.04	-92.6	217.4
θ_4:	-100.31	-360.0	360.0
θ_5:	-50.00	-125.0	125.0
θ_6:	0.00	-360.0	360.0

Other configurations (θ_1, θ_2, θ_3, θ_4, θ_5, θ_6)

```
( *)-[  -1.03°,   27.57°,  -12.04°, -100.31°,  -50.00°,    0.00°]
```

```
PROG-048 Shift released while running (F
FC_RBDK2 LINE 27 T2 PAUSED JOINT
```

```
FC_RBDK2
                        PAUSED     27/33
   13:   ! 0,0.0,0.0
   14:J  P[1] 100% FINE
   15:J  P[2] 100% FINE
   16:  WAIT  10.00(sec)
   17:J  P[3] 100% FINE
   18:L  P[4] 100mm/sec FINE
   19:  WAIT  10.00(sec)
   20:L  P[5] 100mm/sec FINE
   21:J  P[6] 100% FINE
   22:J  P[7] 100% FINE
   23:L  P[8] 100mm/sec FINE
   24:  WAIT  10.00(sec)
   25:L  P[9] 100mm/sec FINE
   26:J  P[10] 100% FINE
   27:J @P[11] 100% FINE
   28:L  P[12] 100mm/sec FINE
   29:  WAIT  10.00(sec)
   30:L @P[13] 100mm/sec FINE
   31:J  P[14] 100% FINE
   32:J  P[15] 100% FINE
  [End]
```

```
POSITION
  User               Frame:  1 Tool: 10

  Configuration: N U T, 0, 0, 0
   x: 9284.026  y:-8183.761  z:-9624.167
   w:  -42.632  p:  -41.535  r:  129.375
```

Figure 2.5: FeatureCAM, RoboDK and Roboguide display extensive information regarding simulated operations.

must not only verify that the device being measured is accurate to a determined degree, one must also ensure that the measuring tools are properly accurate. Absent that verification step, one could potentially bring erroneous data from faulty measurements into a test that were not present prior (Austin, 2014, p. 31). In Austin's research, an electronic Mitutoyo measurement caliper was connected to MeasurLink Real Time Plus software to remove human error from the recordings of data (Austin, 2014, p. 30). This step eliminated much human error that can be produced from monotonous and repetitive recording of data as well as misspellings and misinterpreted transcription errors that can occur. Austin's study did not require a reproducibility study as there was but a single operator utilizing the caliper for measurements (Austin, 2014, p. 30). These steps were simply to ensure the proper verification of the measurement caliper and not the measurements of the research subject components themselves (Austin, 2014, p. 30-31). Outside of these actions, the researcher and the readers of his research, could not verify that the following measurements were accurate as there would be no assurance that the researcher was capable of producing sound data with the tool. This is regardless of whether the tool or the researcher would be to blame and as such, repeatability testing is necessary to produce sound research.

Overall, there are two main types of errors involved in Gauge Repeatability and Reproducibility (Gauge RR) studies with them being gauge error and inspector error (Pan, 2006, p. 501). Gauge error constitutes the utilization of a gauge that produces erratic measurement values despite being used under the same operating conditions (Pan, 2006, p. 501). This error stems from the gauge itself and constitutes repeatability errors within the research (Pan, 2006, p. 501). Inspector error stems from multiple inspectors checking the gauge that has been utilized under the same operating conditions and get different data from one another (Pan, 2006, p. 501). Inspectors who are not properly measuring and/or not following proper procedures are producing a reproducibility problem within their research (Pan, 2006, p. 501). Good research will seek to minimize such errors by ensuring that their tools are repeatable and the human elements within the research are reproducible.

2.2.2 Standards of Testing OLP and/or Simulation Software for Robotics

Bi and his coauthors stated in their research that, "A standard is defined as a prescribed set of rules, conditions, or requirements for definitions of terms, classification of components, specifications of materials, delineation of procedures, or the measurement of quantity and quality for materials, products, systems, services or practices." (Bi, 2020, p. 2). Bi additionally noted that standardization is a critical step in technological evolution and that such standardization can lead to explosive technological and economic results (Bi, 2020, p. 1). In order to conduct a standardized test, one must examine the standard(s) established prior. Industrial robots fall under the active standard of ISO-9283, however this standard has not been updated since it was published in 1998 (Bi, 2020, p. 1). Moreover, the standard itself was limited to “Pose accuracy and pose repeatability; multi-directional pose accuracy variation; distance accuracy and distance repeatability; position stabilization time; position overshoot; drift of pose characteristics; exchangeability; path accuracy and path repeatability; path accuracy on reorientation; cornering deviations; path velocity characteristics; minimum posing time; static compliance; weaving deviations.” (ANSI, 2021). The standard did not take into account pose accuracy based upon offline, simulation software nor pose accuracy between iterations of offline, simulation software.

This directly influences the processes of ISF/SPIF that can require high precision to come through various layers of offline programming software. This lack of updates to the standard’s content despite the advancement of the technology that can include such variations as a result, leaves quite a blind-spot for businesses attempting to utilize these new technologies. The problem exists despite the fact that machining and assembling are some of the most difficult tasks for robots to perform (Bi, 2020, p. 6).

The lack of academic research into standardized testing practices and challenges became apparent once this research began and is highlighted within other works as well (Afzal, 2020). Software testing and Cyber Physical System (CPS) testing are real and academically researched categories that appear to have substantial academic focus into them (Afzal, 2020, p. 1). Regardless of robotics being a sub-category of CPS, academic research into the testing practices and challenges of that specific group are virtually non-existent from a CPS point of view (Afzal, 2020, p. 1). However, testing must occur for robotics CPS nonetheless. As a result, Afzal’s work focused upon conducting a survey to ascertain the general practices of robotics professionals (Afzal, 2020, p. 2-3). Within this survey, it became apparent that the general lack of standards had resulted in compliance testing being primarily a pursuit of internal compliance verification with the exception of certain sub-categories of robotics like self-driving automobiles (Afzal, 2020, p. 5).

Through a lack of standards, a general industry culture that views testing as something of little to no import was found to have emerged (Afzal, 2020, p. 5). When customers and/or sponsors didn't value testing, it resulted in developers not being interested in and/or being discouraged from devising and conducting tests that effectively assessed the capabilities of robotic systems (Afzal, 2020, p. 5). Such circumstances were discovered to be exacerbated by the time-consuming nature some tests require as well as the high costs that can often put off smaller enterprises from investing in the testing (Afzal, 2020, p. 6). Those elements were found to affect the essential collaboration that must occur to develop effective testing; separate teams can often negatively departmentalize an organization's development and testing teams (Afzal, 2020, p. 6). Collaboration can further break down due to a lack of documentation on third-party components within a robotic system and closed-source code (Afzal, 2020, p. 6).

Simulation testing is a promising method whereby testing can be conducted with lowered costs and with increased automation of testing (Afzal, 2020, p. 4). Unfortunately, between the lack of interest in testing and the collaboration breakdown in testing, a distrust of simulation testing within the robotics culture is apparent from reading the research (Afzal, 2020, p. 8). This distrust was found to be due to the difficulty in using simulation software, the lack of real-world elements within simulation, "corner-cases", and the belief that the simulation software lacks the fidelity to be considered reliable (Afzal, 2020, p. 5, 8). Regularly, the respondents to Afzal's study referred back to a lack of standards within their environment of testing and the reader can see that many of the issues could be stemming from that same lack of standards (Afzal, 2020, p. 3-6). As a result of the reading, it is not hard to agree with part of Afzal's conclusion that recommended "… general-purpose guidelines and standards …" to improve robotic systems development and testing (Afzal, 2020, p. 10).

2.2.3 Research into OLP and/or Simulation Software for Robotics

In spite of the lack of focus on testing standards within robotics, a piece of research conducted by H. S. Choi was primarily focused upon the usage of simulation software within the field of robotics and that simulation produces five opportunities for robotics (Choi, 2020).

First, simulation was found to produce low-cost training data in an expeditious manner (Choi, 2020, p. 2). Choi's team did note that the aspect of simulation did not provide a "silver bullet" due to the difference between the simulation and the real-world environment (Choi, 2020, p. 2). Regardless, simulation was discovered to allow for solutions to be presented that would not have been considered possible let alone considered as options at all and that simulation allowed new solutions to be given a trial run prior to real-world testing and deployment (Choi, 2020, p. 2). Secondly, Choi's researchers asserted that simulation could speed the design process for robotic systems while reducing the costs of the design cycle (Choi, 2020, p. 2). Within this category, Choi's team noted that design cycles consisted of the two sub-elements of mechanical design and the control policy design (Choi, 2020, p. 2). Those elements regarded the solution that allows the prescribed tasks to be accomplished and the programming solution to ensure that the robotic system(s) performed those tasks properly (Choi, 2020, p. 2). Both of those elements combined and were proven to have to produce robotic system prototypes, that when vetted, produced a candidate solution (Choi, 2020, p. 2). That stage required time, expense and testing processes and involved situations that could be unsafe for the prototype(s) and/or human operators (Choi, 2020, p. 2). Regardless, the researchers asserted that simulation could reduce the complexity of conducting the testing, as well as the expense and risks associated (Choi, 2020, p. 2). This is through the third opportunity that Choi's research team proposed in that simulation could allow for quicker, safer and fully controlled testing (Choi, 2020, p. 2). This, the research claimed, is due to simulation's ability to have high repeatability, control and a total lack of risk of human and equipment damage from the simulated tests (Choi, 2020, p. 2-3). The fourth opportunity found in the research was the ability to facilitate the design of more intelligent robotic systems by allowing foresight of broader outcome possibilities within a system and could lead to "… introspective control policies …" whereby negative outcomes could be minimized (Choi, 2020, p. 3). The fifth and final opportunity of simulation was the facilitation of understanding Human-Robot Interaction (HRI) (Choi, 2020, p. 3). HRI poses many potential risks in testing and deployment due to the inherent risk of putting a human worker in close proximity to a sturdily constructed and potentially very fast traveling robotic apparatus (Choi, 2020, p. 3). Simulation was found to allow for developers to identify manners whereby they could minimize dangers while still researching industrial, medical and other robotic solutions to problems across myriads of industries and their subprocesses (Choi, 2020, p. 3).

It should be noted that simulation was discovered to have certain drawbacks. Among them, it was uncovered that model composability had come primarily from gaming development that focused on visual fidelity foremost as opposed to accuracy that robotics simulation requires (Choi, 2020, p. 4). Constructing accurate models in simulation were time-consuming and tedious from accurately reproducing the robotic system's functions to reproducing the robotic system's capabilities (Choi, 2020, p. 5). Such circumstances could be exacerbated by not having an actual physical robotic system constructed yet that further complicates design efforts (Choi, 2020, p. 5). The complexity of a model could be varied and the imbuement of a model with excess detail was discovered to consume valuable time (Choi, 2020, p. 5). The other problem was that simplistic simulations of complex systems could lead to inaccurate results that defeated the purpose of the simulation (Choi, 2020, p. 5). In other words, if a simple pick-and-place robotic system is imbued with excessively high detail the turnover on a functional system will be long, expensive and require excess effort. On the other hand, if a complex machining system is given a quick and painless simulation of low detail, the results of the simulation could lead to faulty programs leaving damaged system components and/or a hazardous working environment.

2.2.4 State of the Art for Testing Positional Repeatability

Positional accuracy and the increase of positional accuracy, is a facet of robotics that is heavily researched (Vocetka, 2020, p. 3). Additional research has been committed to the facet of repeatability within robotics (Vocetka, 2020, p. 3). Positional repeatability is defined as a robot being able to return its Tool Center Point (TCP) to the same position, multiple times from the same direction (Vocetka, 2020, p. 2). One such research project studied the use of high-resolution, high-framerate cameras in measuring positional repeatability (Józwik, 2016). This study was conducted to attempt to resolve the issues with robotics diagnostics due to lack of "... proper testing and measuring equipment ...". (Józwik, 2016, p. 86). Their research examined large varieties of innovative testing methods from laser interferometry, telescopic ballbar testing, 3D scanners and multi-camera vision systems (Józwik, 2016, p. 86-87). Such testing methods were discovered to be limited within robotics due to laser interferometry being concerned with linear measurements in one specified direction (Józwik, 2016, p. 86-87). Additionally, ballbar tests simply took their measurements from a singularly, defined plane (Józwik, 2016, p. 87). The research did cite that multiple ballbars allowed for multiple defined planes, however, a limited measurement range was diagnosed to still be an issue for such measurements (Józwik, 2016, p. 87).

As a result, the researchers produced a method whereby repeatability measurements could be taken with the camera and photo analysis software would be able to examine the differences between the repeated movements (Józwik, 2016, p. 92). Despite the accuracy of the methodology, the methodology had certain limitations that the researchers referenced in their conclusions. The method could not measure errors in multiple directions at a time as a result of measurements having been taken from a singular plane (Józwik, 2016, p. 92). The outcome was that additional issues were caused in that the direction of approach could affect repeatability (Vocetka, 2020, p. 21). Despite Józwik's team's research having been able to examine such directional movement errors, the singular plane limited the test to detecting no more than one at a time (Józwik, 2016, p. 92). On a positive note, the methodology did allow for factors such as "… dynamic movement, oscillation, vibration, etc." to be measured and accounted for (Vostek, 2020, p. 2-3). Some doubt can be cast upon the ultimate reliability of optical measuring methods though, as they cannot be considered exact (Vuola, 2012, p. 129).

Unfortunately, the tests that Józwik's team researched are concerned with the robot's positional repeatability and not with the repeatability/accuracy of a robot program generated by multiple OLP software packages. Furthermore, with regards to the research project conducted for this paper, Józwik's research was also less helpful to the goal of a low-tech and low-cost testing solution. The camera utilized in Józwik's research was a Phantom v2511 high-framerate, high-resolution camera (Józwik, 2016, p. 86). Such cameras come at a starting price of around one-hundred and fifty-thousand dollars with comparable lower-priced models from the same manufacturer starting at around one-hundred and ten-thousand dollars (Moynihan, 2014). Certain companies might not blink at such a price tag given the beneficial accuracy of the testing method that they can gain. However, with the singular plane being the result of a singular camera, companies may balk at needing multiple cameras of such prices in order to measure errors in multiple directions. SMEs who the current research hopes to aid, are even more likely to view such investments as unreasonable. Finally, although SMEs may find the technology to be easily utilized and fast, as Józwik's team stated, their reproduction of the technology may still be prohibitive (Józwik, 2016, p. 92). Such a reproduction would require highly tech-savvy employees that SMEs may not have available nor be able to afford to consult on such technological development. As a result, this research methodology, though important to the research of the field and of this paper, does not fit within the goals of keeping the technological and cost angles low.

2.2.5 State of the Art for Testing Positional Accuracy

Positional accuracy is defined as the ability of a robot to reach positions defined in space with the robot's TCP achieving these positions in respect to the robot's base frame (Vocetka, 2020, p. 2). Yeon Oh (or OH) cited this definition in his research on orientation errors while also citing a need for standardized testing techniques examining robot capabilities (Oh, 2019, p. 1). Oh stated that his research was motivated also by the desire to develop a strategy to evaluate accuracy utilizing low-cost hardware that he supplied with ball-bars (Oh, 2019, p. 2). The apparatus, that consisted of a pair of Heidenhain transducers, several springs, disk bearings and ball bearings, were held in between the robot end effector and the base ball bearing (Oh, 2019, p. 3). This equipment array was held in place by the spring pressure between the array and the end effector (Oh, 2019, p. 3). The transducers transmitted information from the array to external hardware and software that examined the data and produced graphs indicating the errors and their range for the user to examine (Oh, 2019, p. 4-11). This method allowed for the testing of angle errors, orientation errors and radial accuracy over a circle radius of a maximum of 70 mm (Oh, 2019, p. 3-4, 6-7, 9-10). The methodology allowed for simultaneous examinations of accuracy and repeatability in radial tests and some of the error causes were also able to be diagnosed by the method (Oh, 2019, p. 9-10). This array measured the link between the ball bearings through magnetic force and succeeded at Oh's goals by being accurate (as accurate as the transducer), being lightweight and relatively low-cost (Oh, 2019, p. 11).

However, the method was predicated upon a mounted apparatus to the robot that can offset accuracy and repeatability measurements slightly despite its relative low weight (Vuola, 2012, p. 129). This effect was exacerbated by Oh's method being held in place by spring pressure that magnifies whatever effect the weight of the apparatus already brought into the equation (Oh, 2019, p. 3). The technical requirements of constructing the measuring hardware for the robot and the external hardware/software combination for processing the data, could also be prohibitive to SMEs who don't necessarily have such talent on staff. Although the costs of the transducers may be relatively inexpensive for large companies, but with Heidenhain transducers starting in the hundreds of dollars, the further hardware requirements to process the transduced energy into measurable data can be daunting (Radwell, 2021). This method may be of a significantly lower cost compared to laser and optical methods, however the cost is still present in a manner that could be prohibitive to SMEs who are in initial stages of implementing Industry 4.0 technologies (Oh, 2019, p. 11). As such, despite the relative low-cost, the technical requirements make it prohibitive for SMEs to necessarily implement absent costly consultations from specialists in related fields.

Another method for measuring accuracy was produced by the research of Vuola whereby a miniaturized Mahr OMS 1000 was placed within a Coordinate Measuring Machine (CMM) that could accurately measure the robot's positional accuracy (Vuola, 2012). The leading advantage is that the method allowed for traceable measurements while interfering the least with the tested robot (Vuola, 2012, p. 126). Again, Vuola cited the leading motivation for the research to be the lack of suitable testing methods available with most methods testing repeatability as opposed to accuracy (Vuola, 2012, p. 126). Issues with this method are found to be predicated on the limitation of the robot motions as well as the robot type. Robot motions are limited to planar motions and the small size of the robot was essential for the robot to fit within the CMM (Vuola, 2012, p. 126-128). Regardless of the exceptional accuracy measurements that the research generated, the research focus and the direction of such research were for mini and micro robots that limits the usage of the research for industrial usage (Vuola, 2012, p. 133). Finally, CMMs cannot be used with "... traditional size robots ..." as the size of the robots would prohibit installation within the CMM and the touching probe method would affect the accuracy of miniaturized robots (Vuola, 2012, p. 126-127). Overall, for SMEs hoping to initiate Industry 4.0 transformations within their processes with more general robots and low-tech/low-cost methods of verifying their process capabilities, this method was also unsuitable for the goals of the research conducted for this paper.

2.2.6 Literature Review Conclusion

Having considered the voluminous and well-devised research into related fields, the status of updating robotics standards is still limited and the research of testing robotics CPS is immeasurably low. Many great options exist for the testing of repeatability and accuracy of individual robots, however the lack of research into the CPS methods of OLP interactions with each other and the final robotic programs they generate is apparent. Also, much of the research into such methods seeks to discover more highly accurate testing methods without fully fleshing out research into other areas of robotics processes such as ISF/SPIF and digital twins. In other words, although testing pose accuracy may be highly fleshed out with hardware/software combinations where the singular program is already well-known and hardwired into the tests, no such research exists to compare similar tests from multiple sources to one another. This research gap is combined with the fact that the methods are almost all highly technically demanding and/or highly expensive. The goal of the research conducted for this paper was to produce a testing method of low-cost and low-technical requirements for the purpose of testing multiple OLP robot programs against one another. This, as has been established before, is necessary for SMEs to be able to implement Industry 4.0 technologies such as ISF/SPIF and digital twinning. As such, this research was based upon what is inexpensive and readily-available within any machine-shop, while still maintaining a high enough accuracy to be applicable to general machining operations.

Chapter 3

RESEARCH METHODOLOGY

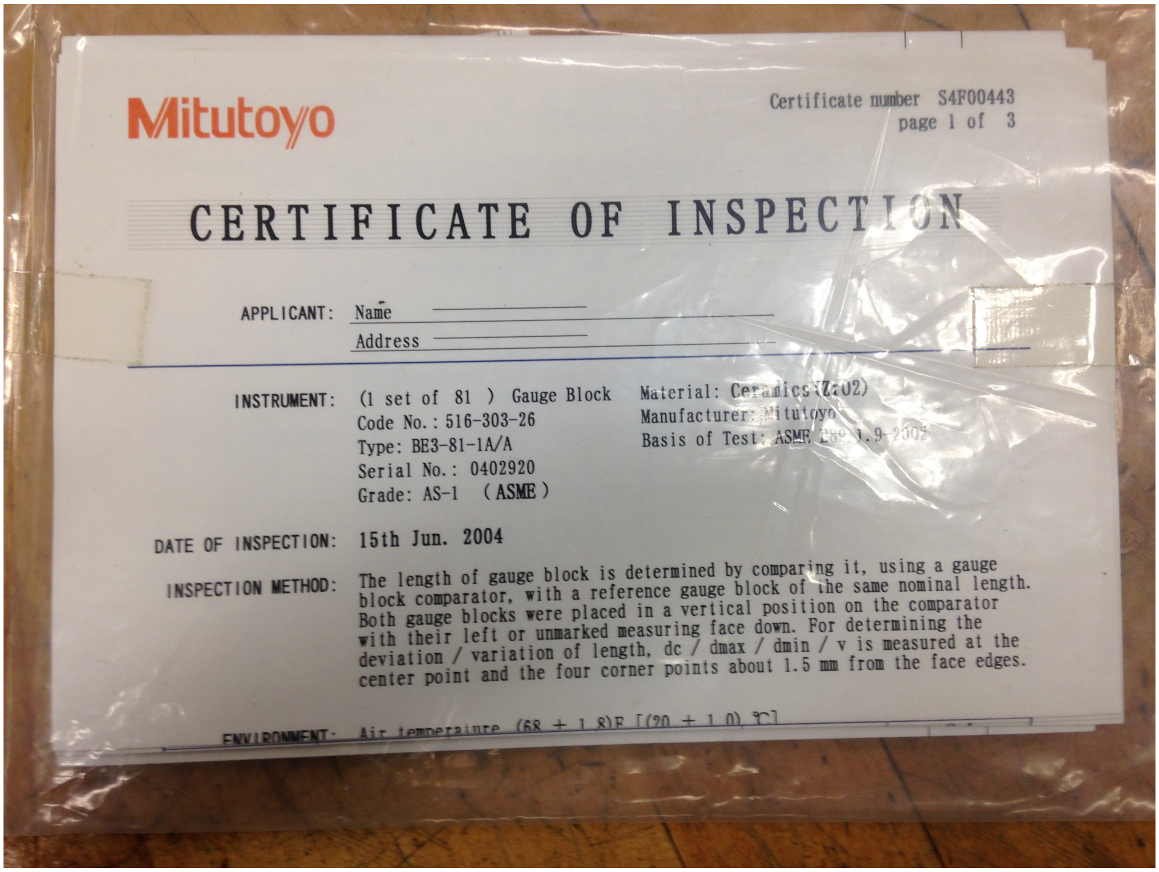
3.1 Introduction

In the process of producing this research, three potential problem areas were highlighted where errors could compromise the veracity of the research. First, user error could have compromised the research in that improper usage of the measuring tools would have compromised the research. Second, the measuring tools themselves could have provided erroneous readings that would result in faulty measurements. Third, the robot was potentially unable to produce the movements to the standard of accuracy required for the test. Combinations of these elements would have dramatically increased the unreliability of the study. As of completion of the study, the robot was found to be unable to produce the movements to the standard of accuracy for the test.

These issues led to the determination that a Gauge Repeatability and Reproducibility (Gauge RR) study was required to deal with potential gauge problems by the user and the gauges themselves. If a user found it difficult to properly utilize such tools in a standard gauge RR study, this would have certainly revealed itself over the course of such a study. Moreover, if the tools themselves had been faulty, then such deviations from accuracy would have been observable over the course of the study. Finally, in order to verify the veracity of the robot, measuring tools that were verified to be accurate were utilized to measure predetermined X, Y and Z movements individually on the robot. Simple movement programs were repeated as needed to reveal whether there was deviation above what the measuring tools are minimally capable of detecting and by how much.

3.2 Gauge Repeatability and Reproducibility for Dial Indicators

Three dial indicators were selected to perform the study and had been used in preliminary experimentation for this research. They were examined utilizing a set of Mitutoyo standard cera gauge blocks that had last been certified on 15JUN2004. (See Figure 3.1)



Mitutoyo

Certificate number S4F00443
page 1 of 3

CERTIFICATE OF INSPECTION

APPLICANT: Name
Address

INSTRUMENT: (1 set of 81) Gauge Block
Code No.: 516-303-26
Type: BE3-81-1A/A
Serial No.: 0402920
Grade: AS-1 (ASME)

Material: Ceramics (ZrO2)
Manufacturer: Mitutoyo
Basis of Test: ASME B89.1.9-2002

DATE OF INSPECTION: 15th Jun. 2004

INSPECTION METHOD: The length of gauge block is determined by comparing it, using a gauge block comparator, with a reference gauge block of the same nominal length. Both gauge blocks were placed in a vertical position on the comparator with their left or unmarked measuring face down. For determining the deviation / variation of length, dc / dmax / dmin / v is measured at the center point and the four corner points about 1.5 mm from the face edges.

ENVIRONMENT: Air temperature (68 ± 1.8)F [(20 ± 1.0) ℃]

Figure 3.1: (Gauge Blocks Certificate of Inspection).

Despite the long period of time since their last inspection, it was determined that they were still viable given their limited usage. The usage they were subjected to consisted of being used for educational purposes once a year and the occasional research project at the university where they were housed (Austin, 2014, p. 30). These cera gauge blocks were used in conjunction with a flat ceramic plate, an unknown brand dial height gauge, machined steel blocks to provide a stop and the three dial indicators with magnetic bases. (See Figure 3.2) The steel blocks allowed for the dial indicators to be set to where the zero would be established around 0.005" within the dial indicator's fullest extension. After having set the dial indicator's zero, the blocks would be inserted underneath the dial indicator to measure the blocks for ten individual tests of each cera block. The indicators were studied against five cera blocks that measured 0.050", 0.125", 0.150", 0.450" and 0.950". The various blocks were utilized so that all the dial indicators could be measured from the greatest common measurement value afforded by all three indicators to the smallest measurement value afforded by the cera blocks, with a divided range in between.

3.3 Robot Assessment

A preliminary series of tests were conducted to assess the overall repeatability of the robot by designing a simple program on the physical robot. This test was designed on the physical robot through online programming to assess whether or not the robot was displaying an obvious negative reaction to the OLP programs it was running contrasted to being programmed online. The three axes were tested by establishing an in and out movement to test the TCP's Z-axis, left and right movement to test the TCP's X-axis and finally, an up and down movement to test the TCP's Y-axis. Each movement had increments of 0.100", 0.250", 0.500" and 0.750". These increments were selected to display a broad extent of the smallest dial indicator size (1") and to assess whether a certain movement length was easier for the robot to replicate than another. The results of the initial assessment can be seen in Chapter 4.

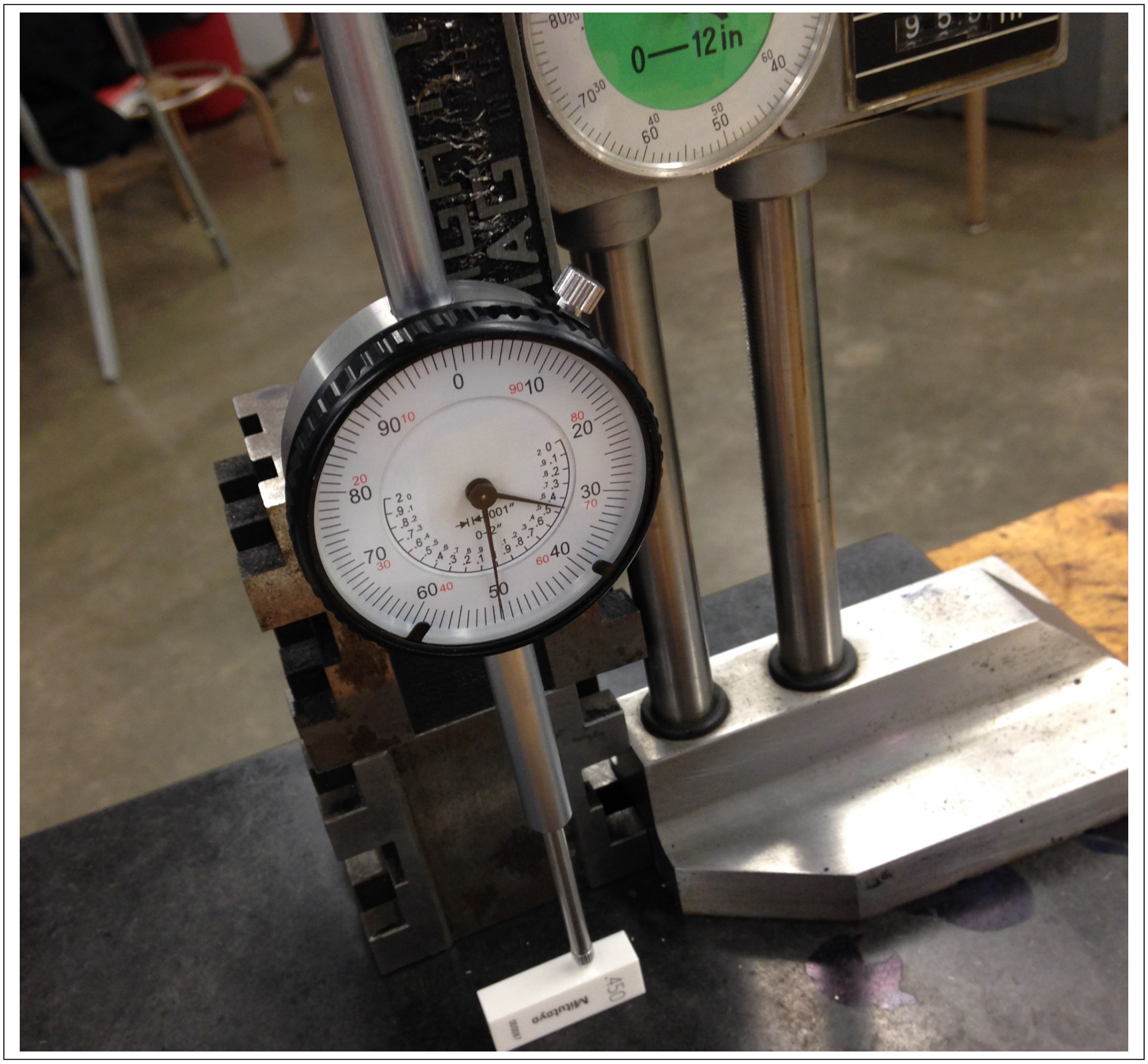


Figure 3.2: Gauge RR Study Layout.

It should be noted that the variations observed (see Figure 4.4) are inherent in the physical robot and that the accuracy of the robot alone are in play. No OLP software package or OLP generated program code was utilized for this small experiment and the variations are solely attributable to the robot as the dial indicators had already been checked for accuracy prior. Moreover, this initial test was simply to gain a general understanding of the robot's capabilities and to gain a plan to perform the actual experiment on the robot.

As such, the haphazard nature of poorly captured images and/or missing images were the early issues ironed out in this stage of experimentation.

3.4 Robot Experiment Methodology

3.4.1 Robot Program Methodology

The core experiment of the robot utilized three programs with the first designed in FeatureCAM, imported into RoboDK, imported into Roboguide before then being uploaded to the physical robot. A second program was designed in RoboDK, imported into Roboguide and then uploaded to the robot. The final program was developed solely on Roboguide before uploading to the physical robot. Each program was designed to replicate the actions made by the first program (developed in FeatureCAM) so as to produce mirroring robot movements on the physical robot. The programs consisted of a "home" zone from where the robot could begin the program and return in between each axis test. Then the program would move to a safe position off of the Z-axis dial indicator (the 1" Fowler dial indicator). Next, the robot would move to touch off on the dial indicator and the dial indicator would be set to zero. Then the robot would move into further contact with the dial indicator for a distance of 0.5". It is at this point that the dial would be read and the results recorded. As will be noted later, it was determined to also record the positional data from the robot's return to the zero position as well. Both sets of data was recorded by taking pictures of the dial indicator so as to prevent excessive time from being consumed recording the data in hand writing as well as the possibility of recording the measurement incorrectly (see Figure 3.3).

After the data was recorded, the line of code would be repeated in order to record the movements of the robot. These repeats were carried on for a total of twenty-five mea-

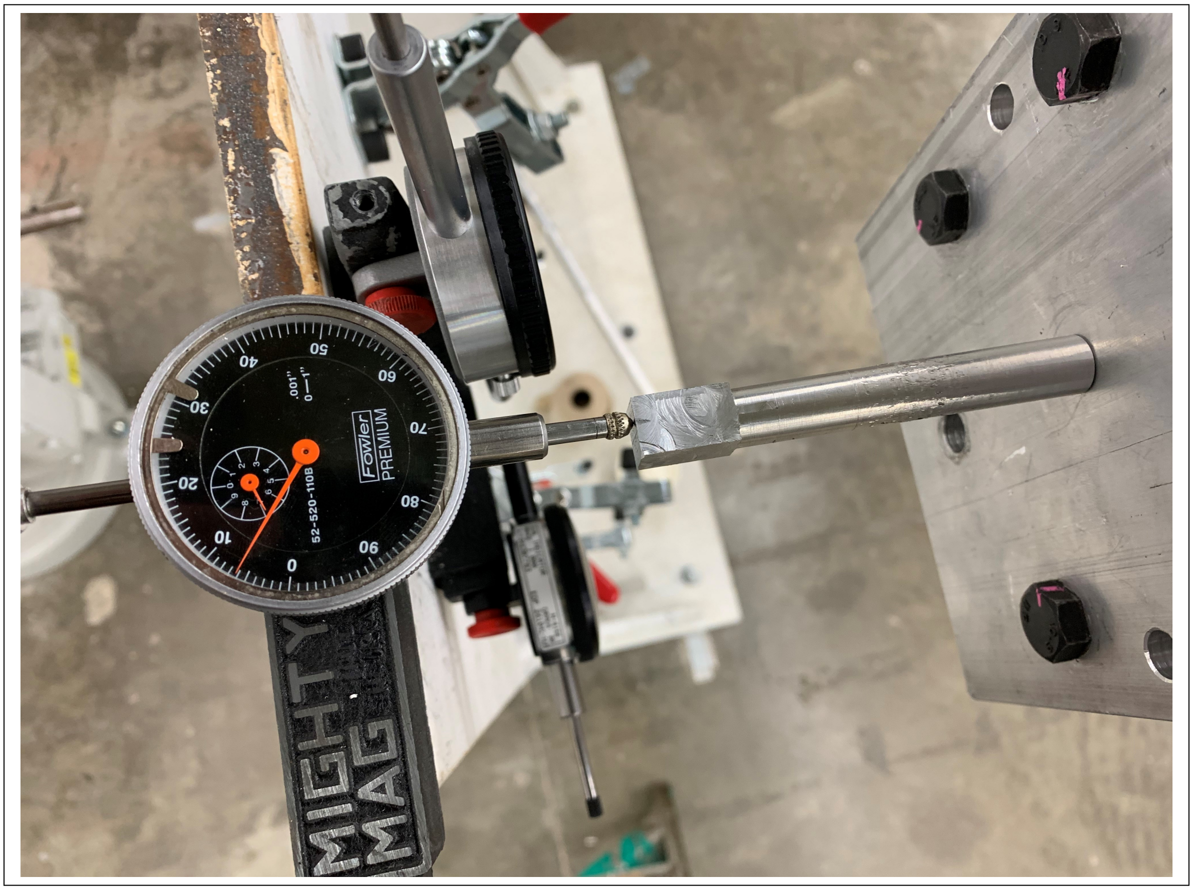


Figure 3.3: Example of a single Z-Axis movement being recorded.

surements of the dial indicator on the 0.5" of inward travel and twenty-five measurements of the set zero. After the Z-axis measurements had been taken, the program returned to the same "home" position where it could safely travel to the next axis to be measured. The X and Y axes' measurements were conducted in the same manner as the Z axis and this process was repeated for all three programs (the results can be seen in Chapter 4.3).

It was noted in the initial robot tests that there was a certain variation when the robot would return to the zeroed position. As opposed to the initial tests where only the variation from the destination was considered, it was determined that it could be beneficial to include the variation from the set zero as well. These results were similarly recorded as variations in thousandths of an inch from the established zero (those results may be seen in Chapter 4.3).

3.5 The Physical Components of the Experiment

The experiment consisted of a PC with the FeatureCAM software package from the Autodesk software suite, the software package of RoboDK and the Fanuc software package of Roboguide. A Fanuc R-2000iC/165F industrial robot was also utilized in conjunction with a steel frame to construct a roboforming work area. Three dial indicators were utilized to measure the Cartesian coordinate movements of the robot's tool. These movements were along the X-axis, Y-axis and Z-axis. The length of these movements were made to match and the indicators were placed in line with those paths. The indicators were set to zero off of the first test so as to provide a baseline for the following tests. Finally, the tool utilized was of a flat and square construction so as to reduce the slipping of the dial indicators off of the current tool's spherical end. Preliminary testing revealed this flaw and a square end of the same dimensions but with a different shape allowed for the tests to be more accurate.

The physical components were verified to reduce the possibility of human error, robot error and measurement error in the experiment. This was accomplished by first performing a Gauge RR study upon the dial indicators to ensure that they were operating to the factory assured standard. If they were found to be defective, they would have been replaced with properly functioning dial indicators. Second, the robot was tested to ascertain the margin of error within the robot. The robot in question was donated by an auto manufacturer and was used to hold vehicle frames as they were being welded. This processing history made it possible that the robot was not up to the factory standard and this experiment's success was based upon whether or not the robot could maintain a consistent level of accuracy.

Finally, the experiment itself was carried out over iterations of measurements. If there was very little variation in the measurements recorded, then the experiment's tests would have been over fewer iterations. The larger variation that was detected required more testing to produce a usable set of data for study.

3.6 The Virtual Components of the Experiment

Each of the programs had mirroring programs developed in order to provide the robot with similar movements. In FeatureCAM, this entailed directing the program to move what the program believed to be a Computer Numeric Control (CNC) mill end to perform some basic movements for the X, Y and Z axes of the Cartesian coordinate system. In RoboDK and Roboguide, this involved creating a pair of digital twins to replicate the robot and the steel frame so as to match the physical workspace's layout. The robot assets within the software packages are proprietary assets developed by RoboDK and Roboguide and as such, were not and could not be modified. The steel frame assets were made by generating and re-purposing assets to match the dimensions of the physical frame as well as its distance and angle from the robot in the physical space. Through each DT, matching programs moved the robot's tool to match the movements of the other programs so as to attempt to produce mirroring robot programs.

3.7 The Environment of the Experiment

First, the FeatureCAM program was loaded into RoboDK to translate the CNC code into a usable Karel robot language. Then, the movements were matched to the digital robot workspace before that language was run through the Roboguide program to produce usable binary code. This was necessary to produce code that the physical robot could read and the program could then be tested on the robot. Second, the RoboDK DT was used to create a second matching program within the digital robot workspace and the Karel code was again translated into usable binary code. This was then uploaded to the robot in order to test. Finally, the Roboguide DT was utilized to create a third matching program that was then translated within Roboguide to the proper binary format, uploaded to the robot and tested.

3.8 Methodology of Processing the Results of the Experiment

After performing each test, the measurements of the physical robot's movement were compared with one another to determine whether or not the robot's movements matched the movements of the robot from the other tests for X, Y and Z. This was done by using the three dial indicator gauges. If deviations were detected, they were measured to see how much the deviation was within each of the Cartesian movements. This was accomplished by setting the gauges to zero off of the first test and the values that are more or less of that value were recorded as deviations. If the value didn't shift from zero, then the result was recorded as a lack of variation. If variations were detected, then the experiment was analyzed to determine whether the cause of the deviation was mechanical, code-based or a result of human-error in the construction of the DTs.

3.9 Conclusion

As stated in previous chapters, no other research has examined deviations in movement for robots within robot programs generated by OLP software. Also, no other research been conducted to provide a low-cost, low-tech solution for this or other robot motion testing. This research attempted to fill this void by utilizing tools that any machinist shop has readily available and software packages that merely require self-training to utilize.

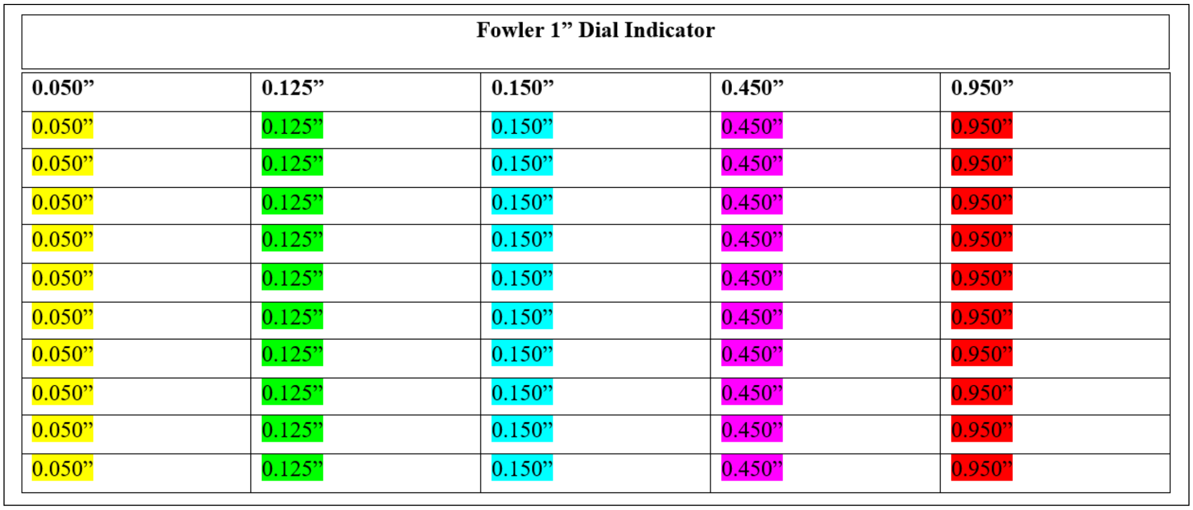
Chapter 4

EXPERIMENT RESULTS

The following chapter is structured to follow the order of operations of the experiment and the results.

4.1 Gauge RR Results

The first tool studied was a Fowler 1" dial indicator that was studied against all five of the measuring blocks and with each measurement block being tested no less than ten times individually. The tests resulted in exact measurements of each block that strictly adhered to all cera blocks’ dimensions without fail for all fifty tests. None of the findings were rounded or averaged as all testing sequences produced mirroring results (see Figure 4.1).

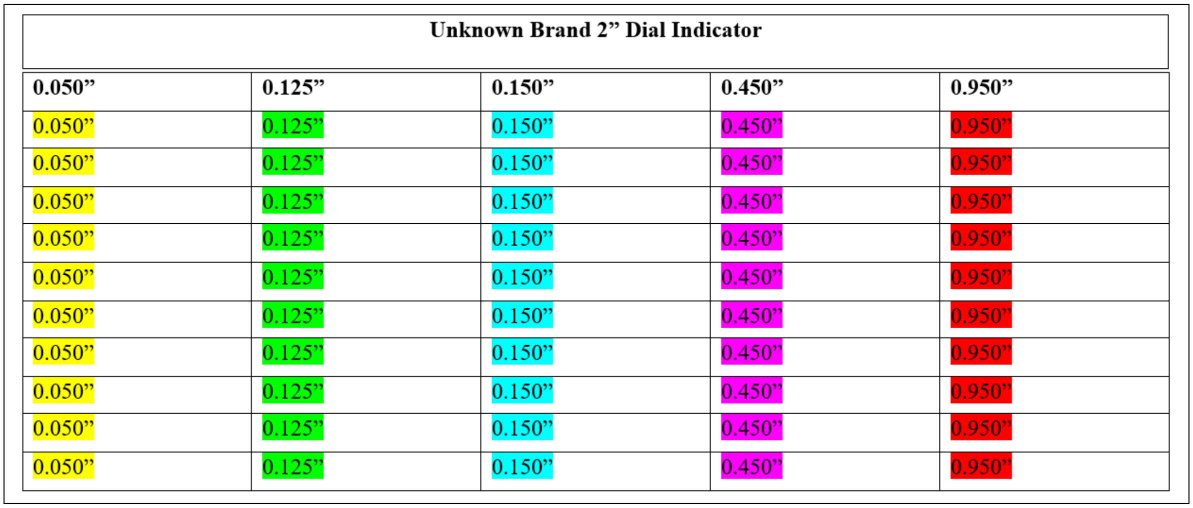


Fowler 1” Dial Indicator

0.050”	**0.125”**	**0.150”**	**0.450”**	**0.950”**
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”

Figure 4.1: Gauge RR results for Fowler 1" dial indicator. Black and bold text indicate target with color coded results in the column below.

The second tool studied was an unknown brand 2" dial indicator that was studied against all five of the measuring blocks and with each measurement block being tested no less than ten times individually. The tests resulted in consistent measurements of each block that did not strictly adhere to all cera blocks’ dimensions. Although the measurements were consistent for all fifty tests, there was an observed variance of less than 0.001" for forty of the tests. It should be noted that the dial indicator has accuracy to a 0.001" and the variances fell underneath that threshold. None of the findings were rounded or averaged as all testing sequences produced mirroring results (see Figure 4.2).

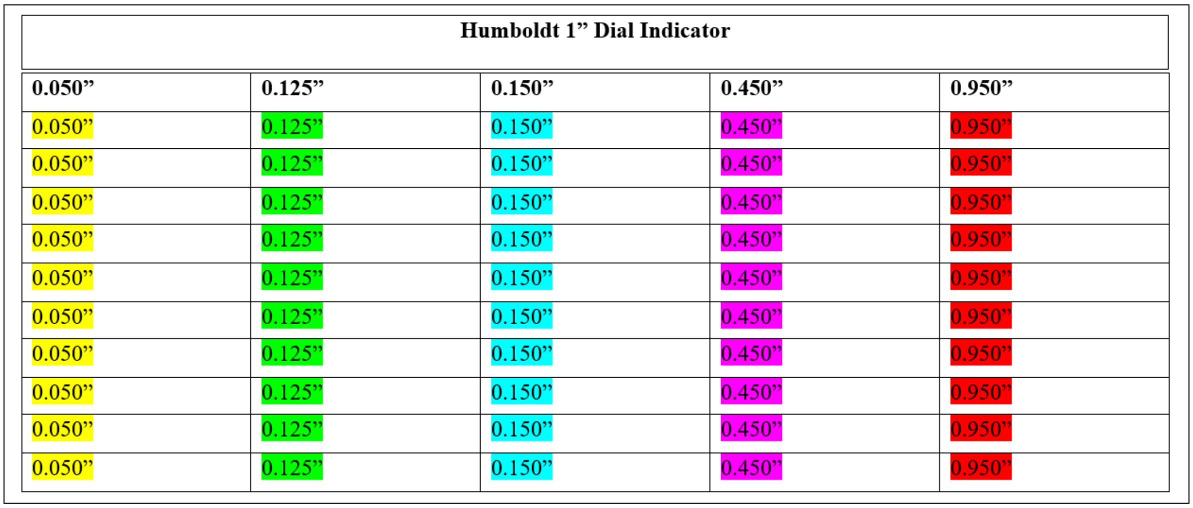


Unknown Brand 2” Dial Indicator

0.050”	**0.125”**	**0.150”**	**0.450”**	**0.950”**
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”

Figure 4.2: Gauge RR results for Unknown brand of 2" dial indicator. Black and bold text indicate target with color coded results in the column below.

The third tool studied was a Humboldt 1" dial indicator that was studied against all five of the measuring blocks and with each measurement block being tested no less than ten times individually. The tests resulted in consistent measurements of each block that did not strictly adhere to all of the cera blocks’ dimensions. Although the measurements were consistent for all fifty tests, there was minute variance of less that 0.001" for forty of the tests. It should be remembered that the dial indicator has accuracy to a 0.001" and the variances fell underneath that threshold. None of the findings were rounded or averaged as all testing sequences produced mirroring results (see Figure 4.3).

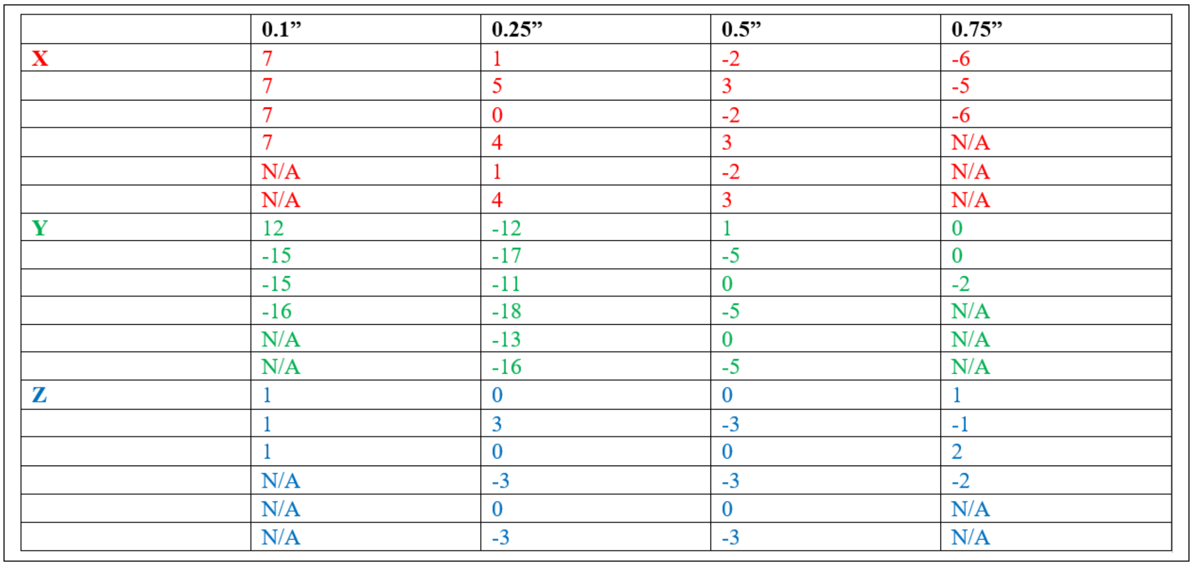


Humboldt 1” Dial Indicator

0.050”	**0.125”**	**0.150”**	**0.450”**	**0.950”**
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”
0.050”	0.125”	0.150”	0.450”	0.950”

Figure 4.3: Gauge RR results for Humboldt 1" dial indicator. Black and bold text indicate target with color coded results in the column below.

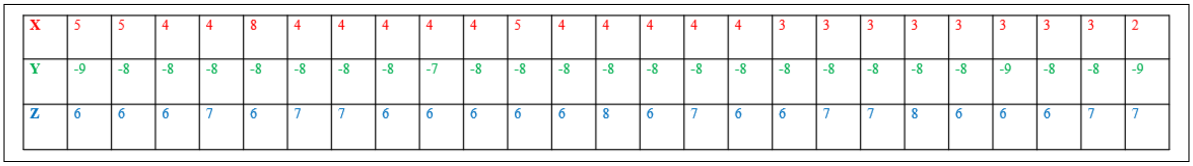
4.2 Robot Assessment Results



	0.1”	**0.25”**	**0.5”**	**0.75”**
X	7	1	-2	-6
	7	5	3	-5
	7	0	-2	-6
	7	4	3	N/A
	N/A	1	-2	N/A
	N/A	4	3	N/A
Y	12	-12	1	0
	-15	-17	-5	0
	-15	-11	0	-2
	-16	-18	-5	N/A
	N/A	-13	0	N/A
	N/A	-16	-5	N/A
Z	1	0	0	1
	1	3	-3	-1
	1	0	0	2
	N/A	-3	-3	-2
	N/A	0	0	N/A
	N/A	-3	-3	N/A

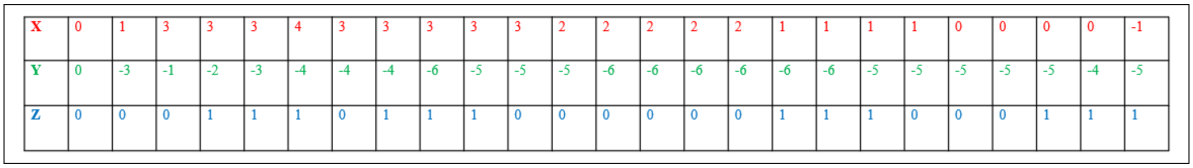
Figure 4.4: The left column denotes the axis along with the matching color scheme. The upper row denotes how far the robot was set to move and is in black and bold. The measurements reported are shown in how many thousandths of an inch the robot deviated from the set destination on that axis. The nomenclature "N/A" denotes measurement values whose images were accidentally missed or illegible.

4.3 Robot Experiment Results



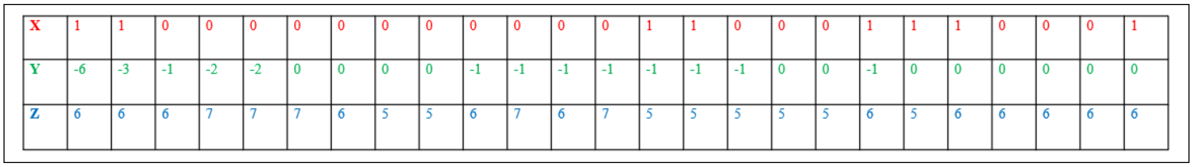
X	5	5	4	4	8	4	4	4	4	4	5	4	4	4	4	4	3	3	3	3	3	3	3	3	2
Y	-9	-8	-8	-8	-8	-8	-8	-8	-7	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-9	-8	-8	-9
Z	6	6	6	7	6	7	7	6	6	6	6	6	8	6	7	6	6	7	7	8	6	6	6	7	7

Figure 4.5: Variations in thousandths of an inch from the 0.5" set travel distance from an established zero for FeatureCAM. Left column letters in bold denotes axis along with color coded results in following rows.



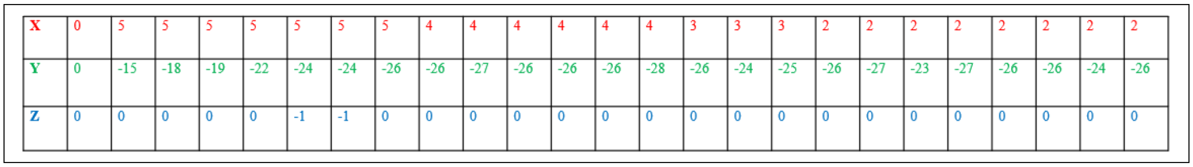
X	0	1	3	3	3	4	3	3	3	3	3	2	2	2	2	2	1	1	1	1	0	0	0	0	-1
Y	0	-3	-1	-2	-3	-4	-4	-4	-6	-5	-5	-5	-6	-6	-6	-6	-6	-6	-5	-5	-5	-5	-5	-4	-5
Z	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	1	1	1

Figure 4.6: Variations in thousandths of an inch from the set zero for FeatureCAM. The first entry is the zero being set with all following entries being results from robot's return movement. Left column denotes axis along with color coded results in following rows.



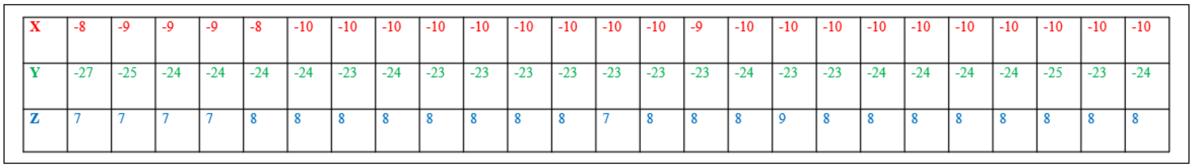
X	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	1	1	0	0	0	1
Y	-6	-3	-1	-2	-2	0	0	0	0	-1	-1	-1	-1	-1	-1	-1	0	0	-1	0	0	0	0	0	0
Z	6	6	6	7	7	7	6	5	5	6	7	6	7	5	5	5	5	5	6	5	6	6	6	6	6

Figure 4.7: Variations in thousandths of an inch from the 0.5" set travel distance from an established zero for RoboDK. Left column denotes axis along with color coded results in following rows.



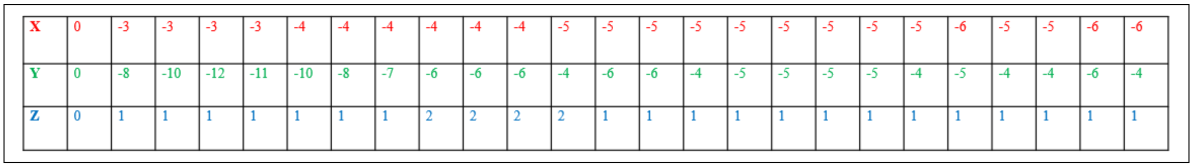
X	0	5	5	5	5	5	5	5	4	4	4	4	4	4	3	3	3	2	2	2	2	2	2	2	2
Y	0	-15	-18	-19	-22	-24	-24	-26	-26	-27	-26	-26	-26	-28	-26	-24	-25	-26	-27	-23	-27	-26	-26	-24	-26
Z	0	0	0	0	0	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 4.8: Variations in thousandths of an inch from the set zero for RoboDK. The first entry is the zero being set with all following entries being results from robot's return movement. Left column denotes axis along with color coded results in following rows.



X	-8	-9	-9	-9	-8	-10	-10	-10	-10	-10	-10	-10	-10	-10	-9	-10	-10	-10	-10	-10	-10	-10	-10	-10	-10
Y	-27	-25	-24	-24	-24	-24	-23	-24	-23	-23	-23	-23	-23	-23	-23	-24	-23	-23	-24	-24	-24	-24	-25	-23	-24
Z	7	7	7	7	8	8	8	8	8	8	8	8	7	8	8	8	9	8	8	8	8	8	8	8	8

Figure 4.9: Variations in thousandths of an inch from the 0.5" set travel distance from an established zero for Roboguide. Left column denotes axis along with color coded results in following rows.



X	0	-3	-3	-3	-3	-4	-4	-4	-4	-4	-4	-5	-5	-5	-5	-5	-5	-5	-5	-5	-6	-5	-5	-6	-6
Y	0	-8	-10	-12	-11	-10	-8	-7	-6	-6	-6	-4	-6	-6	-4	-5	-5	-5	-5	-4	-5	-4	-4	-6	-4
Z	0	1	1	1	1	1	1	1	2	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure 4.10: Variations in thousandths of an inch from the set zero for Roboguide. The first entry is the zero being set with all following entries being results from robot's return movement. Left column denotes axis along with color coded results in following rows.

Chapter 5

CONCLUSION

5.1 Introduction

Overall, the results display a high level of inconsistency in regards to the Y-axis measurement results that indicates that the robot used for this study was unsuited to perform the experiment. Consistent errors would not have been problematic as the margin of error could potentially have been taken into account when generating code or some other modifications could have been applied to remove the consistent errors. However, with the inconsistency displayed by these results when performing the same sequences of code, a clear solution to ameliorate errors was and is not present. Such actions to ameliorate those inaccuracies for those specific movements are likely to create problems within other areas of code for other movements. As such, the experiment as it stands will have to be treated as a failure due to the utilized mechanism to perform the test having been incapable of performing the experiment reliably.

5.2 Examining the Results of the Study

The results derived under the subsection entitled "Robot Assessment" were not originally intended to be factored into the actual study. Instead, those examinations were intended to serve as a general litmus test to ascertain if the study could be carried out safely and to ensure that the results could be recorded properly. Some of the images were rendered useless due to a combination of blur and/or angle from the dial indicator that made the results unusable in an objective study. The discovery of these initial problems enabled the researcher to find better angles and positions to produce usable images. However, after examining the results of the core experiment of the study, it was decided to include those results to demonstrate the inconsistency within the tests along the Y-axis.

New or refurbished Fanuc R-2000iC/165F robots have a repeatability accuracy of +/-0.00787402" (RobotWorx, 2022). The robot utilized for this experiment is a robot donated to Western Kentucky University by Nissan North America. This robot was used to lift car frames while they were being welded and there is no indication that the robot is refurbished.

5.2.1 X-Axis Variation

The X-axis study utilizing the FeatureCAM generated robot program when travelling to 0.5" had an overall average variation of 0.00388" and a range of 0.003". Although the range and average was higher than that of the Z-axis results (seen later in the chapter), these numbers are still well within +/-0.00787402". The most frequent nonconforming result measured was 0.004" a relative frequency of 48 percent. However, a single result of 0.008" did occur with a relative frequency of 4 percent. In the case of RoboDK, the program produced results that had an average variation of 0.00032" and range of 0.001". The most frequent result was of perfect conformance that occurred with relative frequency of 68 percent. Roboguide's version of the program produced an average variation of -0.00968" with a range of 0.002". This average fell below the lower range of +/-0.00787402" by almost 0.002". The most frequent nonconforming result was 0.010" that occurred with a relative frequency of 76 percent. By comparison, the only result that fell within the +/-0.00787402" range was 0.007" whose relative frequency was a mere 20 percent.

The results of the robot returning to its zeroed position during the X-axis tests revealed that FeatureCAM's program produced an average variation of 0.00168". The range produced by this series of tests stands at 0.005" with the most frequent nonconforming result of 0.003" having a relative frequency of 32 percent. The highest and lowest reaches of the results range didn't come close to the +/-0.00787402" zone and the furthest reaching result of 0.004" represented a relative frequency of only 4 percent. RoboDK's results when returning to its zeroed position showed an average variation of 0.00336" with a range of 0.005". It should again be noted that the first result from every result table reflects the zero having been set. However, this result is still valid as a result as the robot achieved the position where the zero was set. With this set of results, the results deviated significantly more than others and it should be noted that the only conforming result was the first result when the zero was set on the dial indicator with a relative frequency of 4 percent. The most frequent nonconforming result of 0.002" occurred with a relative frequency of 32 percent and the most extreme nonconforming result of 0.005" occurred with relative frequency of 28 percent. Roboguide's program when returning to a zeroed position yielded results with an average of -0.00436" and whose range was 0.006". The most frequent nonconforming result of 0.005" occured with relative frequency of 44 percent and the only conforming result occurred with relative frequency of 4 percent.

5.2.2 Y-Axis Variation

The Y-axis study on the travel to 0.5", FeatureCAM's program resulted in an average variation of -0.00808" with a range of 0.002". The most frequent nonconforming result of 0.008" was noted to have occurred with a relative frequency of 84 percent. This is set outside of the +/-0.00787402" zone with only a single result occurring within that zone for a relative frequency of 4 percent for 0.007". RoboDK's program produced an average variation of -0.00088" with a range of 0.006". The most frequent result was perfectly conforming with a relative frequency of 48 percent and the most excessive nonconforming result being -0.006" with a relative frequency of 4 percent. No measured result occurred outside of the +/-0.00787402" zone. Roboguide's version produced an astonishing average variation of -0.02376" with a range of 0.004". The most frequent nonconforming results were recorded to be both -0.023" and -0.024" with a relative frequency of 44 percent to produce a combined relative frequency of 88 percent. It should be noted that the smallest deviation is more than double the +/-0.00787402" zone and to produce a minimum distance of over 0.015" below that zone.

When the results were examined for the robot returning to its zeroed position from the Y-axis test location, one can see that FeatureCAM's program resulted in an average variation of -0.00448". These results had a range of 0.006" with the most frequent nonconforming result being -0.005" at a relative frequency of 36 percent. This is shortly followed by -0.006" having a relative frequency of 28 percent. These values still fell within the +/-0.00787402" zone. RoboDK's program return to zero were in sharp contrast with the same group's travel to 0.5". This can be noted in that the average variation stood at -0.02348" with a range of 0.028". Even if one factored out the initial set zero, the range was still a dramatic 0.013". The most frequent nonconforming result of -0.026" stood at a relative frequency of 40 percent and was more than triple the +/-0.00787402" repeatability standard. Roboguide also displayed sharp contrast to its 0.5" travel results when one observes that the previously large variations came significantly closer to conformance when returning to the zeroed position. The average variation of said results were recorded to be -0.00604".

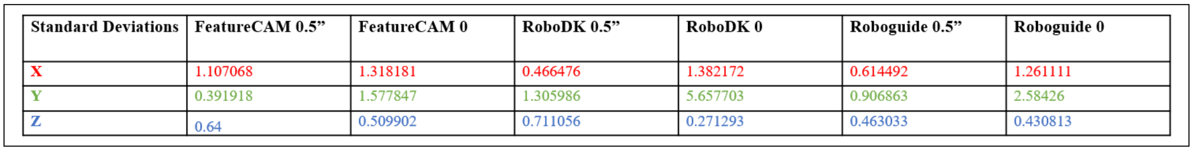
However, the range increased to triple its previous range (0.004") and was recorded at 0.012". Even if the initial result when the dial indicator was set to zero is removed, the range would still be 0.008". The most frequent nonconforming results were recorded to be -0.006" and -0.004" with matching relative frequencies of 24 percent. Although these measures are within the +/-0.00787402" zone, the relative frequency of results outside of that zone produced a matching relative frequency of 24 percent.

5.2.3 Z-Axis Variation

The FeatureCAM developed program, the results of the study show that the Z-axis variation when traveling to 0.5" had an average variation of 0.00648" and a range of 0.002". Its most frequent nonconforming result was 0.006" at a relative frequency of 60 percent. Although its variation did extend higher than the repeatability accuracy of a new/refurbished robot with its +/-0.00787402" tolerance, the relative frequency of those results was a mere 8 percent by comparison. RoboDK's program's Z-axis variation for traveling to 0.5" had an average variation of 0.00588" and a range of 0.002". Its most frequent nonconforming result was also 0.006" with its relative frequency being 48 percent. However, its highest variation did not extend higher than 0.007" and the relative frequency of that variation was lower than the relative frequency of the 0.005" variation (20 percent compared to 32 percent). Roboguide's program Z-axis variation for 0.5" travel had an average variation of 0.00784" and a range of 0.002". Although that was within the repeatability accuracy of +/-0.00787402", the relative frequency of measurements past that limit was significant with 0.008" measurements having a relative frequency of 76 percent. By contrast, 0.007" had a relative frequency of a mere 20 percent and a single measurement of 0.009" still being present for a relative frequency of 4 percent.

When examining the variations in the robot's movement when returning to its zero position, it was found that FeatureCAM's program had an average variation of 0.00048" and a range of 0.001". The most frequent nonconforming result was 0.001" at a relative frequency of 48 percent. This was overshadowed in that the relative frequency of a perfectly conforming result (0) was 52 percent and 0.001" is well within the +/-0.00787402" repeatability zone. RoboDK's version showed that the average was -0.00008" with a range of 0.001" below the zeroed position. Its most frequent nonconforming result of -0.001" occurred with relative frequency of 8 percent and is significantly under the +/-0.00787402" repeatability standard. Perfect accuracy was achieved at a relative frequency of 92 percent. Roboguide's iteration had higher variation when returning to the zeroed position similar to its relatively high variation traveling to the 0.5" mark. Roboguide's average variation was 0.00112" with a range of 0.002". The most frequent nonconforming result was also 0.001" at a relative frequency of 84 percent. Although 0.002" of variation occurred unlike the previous two programs, it was still well within the +/-0.00787402" standard.

5.2.4 Examining the Standard Deviation of the Results



Standard Deviations	FeatureCAM 0.5"	FeatureCAM 0	RoboDK 0.5"	RoboDK 0	Roboguide 0.5"	Roboguide 0
X	1.107068	1.318181	0.466476	1.382172	0.614492	1.261111
Y	0.391918	1.577847	1.305986	5.657703	0.906863	2.58426
Z	0.64	0.509902	0.711056	0.271293	0.463033	0.430813

Figure 5.1: Standard deviation values for 0.5" travel and return to zeroed position. Top row in black indicates the program and process with color coded axes listed in the left column.

At a glance one can plainly see from Figure 5.1 that the Y-axis movements displayed the most dramatic nonconforming results from the various positions. Those positions being not only the point of 0.5" travel, but also the original position where the zero was set on the dial indicators. Although the zeroed position results might not be considered important by the casual observer, when one considers that those positions are generated within the code in the same manner as the positions for the 0.5" travel, then the variation of results holds the same relative weight within the research. Although variations occurred within the other tests of the axes of movement, the variations were not as broad of variations and generally displayed a higher level of consistency. An example would be the standard deviation for RoboDK's X-axis return to zero movement, 1.382172, was the highest of the standard deviations outside of the Y-axis standard deviations. By contrast, the highest for the Y-axis movements, being the RoboDK return to zero, was calculated to be 5.657703. Furthermore, for all but one of the sets of tests (that being the FeatureCAM travel to 0.5") the standard deviation for the Y-axis movements were higher than the standard deviations for the other axes movements.

5.2.5 Bias Testing of the Results

It should be noted that bias testing (t-Test: paired two sample for means) was conducted for the data garnered from this experiment. These tests were processed in Excel and the results of these tests can be found in the appendix in figures A.1 through A.6. Each figure includes all three axes for one of the three program iterations and for travel to either the 0.5" joint target or the 0" joint targets. The tests showed that there was statistical significance in the data and is strong evidence to reject the null hypothesis. This confirms what is plainly seen in the data examined and further cements the failure of the experiment.

5.3 Potential Causes of Excessively Nonconforming Results

Exact causes are difficult to determine but some potential causes can be ruled out in their entirety. First, the idea that these variations were caused by deteriorations within the robot programs due to iterations of translation and/or importing, is not a viable cause.

After a program is imported from FeatureCAM into RoboDK, one can immediately check the positional coordinates in the targets for the robot under "Options" in the target's drop-down menu. Additionally, after the program has been generated through RoboDK's post processors into the Karel language, one can read all of the positional coordinates at the base of the program code in the text file. This can be seen in the Roboguide version at the base of the program code as well. Finally, these coordinates can be seen in the loaded program onto the physical robot through the teach pendant by selecting each individual positional coordinate and selecting "Position" in the screen's menu. The coordinate will be displayed in the format that it reads from the base of the code (i.e. a joint target in degrees or a Cartesian coordinate in millimeters).

After fully examining each coordinate that was output from the various programs and the physical robot, it was determined that the positional coordinates were exactly the same without any deviation from iteration of code to iteration of code. Another manner whereby this possibility was eliminated was the fact that the results do not have reliable deviation from their positional coordinates. The standard deviation results from RoboDK's iteration of the program when returning to zero, range from a Z-axis standard deviation of 0.271293 to a Y-axis standard deviation of 5.657703. The same program had a standard deviation for all three axes of movement that was less than half of their counterparts when the robot was traveling to the 0.5" mark. The lack of consistent deviations are not indicative of an error being produced through the program's coordinates. By both comparison of the results and through the verification of the positional data within the program code, this potential cause was eliminated fully as a possibility.

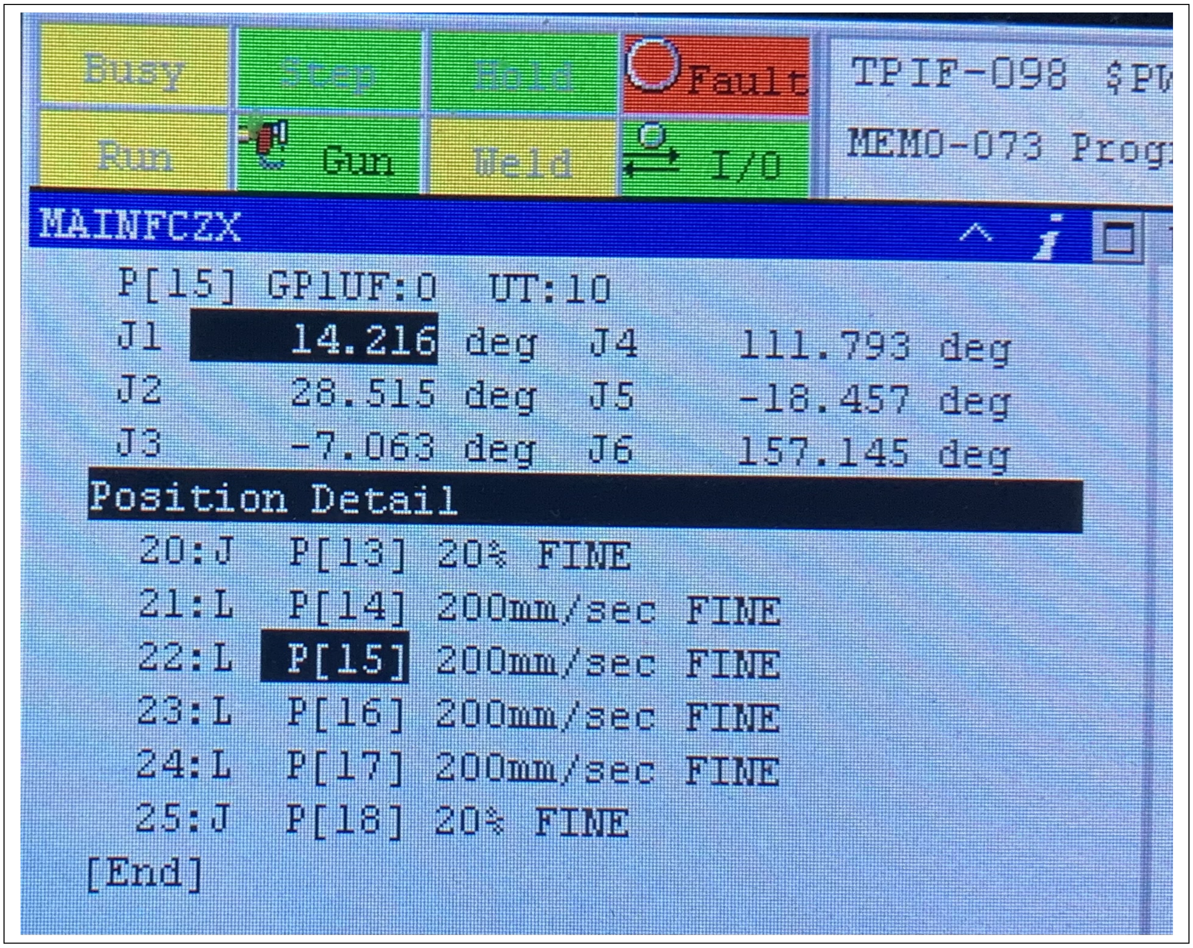
Causes such as the dial indicators being damaged or malfunctioning were eliminated at the outset with the gauge repeatability and reproducability study conducted prior to the experiment. Poorly captured data was eliminated by taking photographs of the results as opposed to hand-writing them down. Moreover, when poorly captured images were taken during the actual experiment, the process was repeated as needed to have a minimum of 25 legible images of all positions. Each image was legible to examine the results of both the 0.001" indicator needle and the 0.1" indicator needle. In Figure 5.2 one can see a less legible image of the last test of FeatureCAM's program returning to zero. The 0.1" indicator needle was still legible to indicate the movement type being examined and the 0.001" indicator needle was still legible to clearly see the value of -0.005" of variation from the target.

Figure 5.2: Final image of FeatureCAM's program's Z-axis test returning to its zeroed position for a value of -0.005".

Another less legible picture shows a less legible 0.1" indicator needle being slightly obscured by angle and the 0.001" indicator (see Figure 5.3). However, this image clearly showed that the robot was pressing in upon the dial indicator's plunger indicating the movement type. Moreover, despite the angle, the 0.001" indicator needle could still be seen clearly sitting upon the mark of 0 indicating that there was no variation from the X-axis target coordinate. This image was captured for the twelfth measurement of RoboDK's program traveling to 0.5".

Figure 5.3: Twelfth image of RoboDK's program's X-axis test traveling to 0.5" for a value of 0".

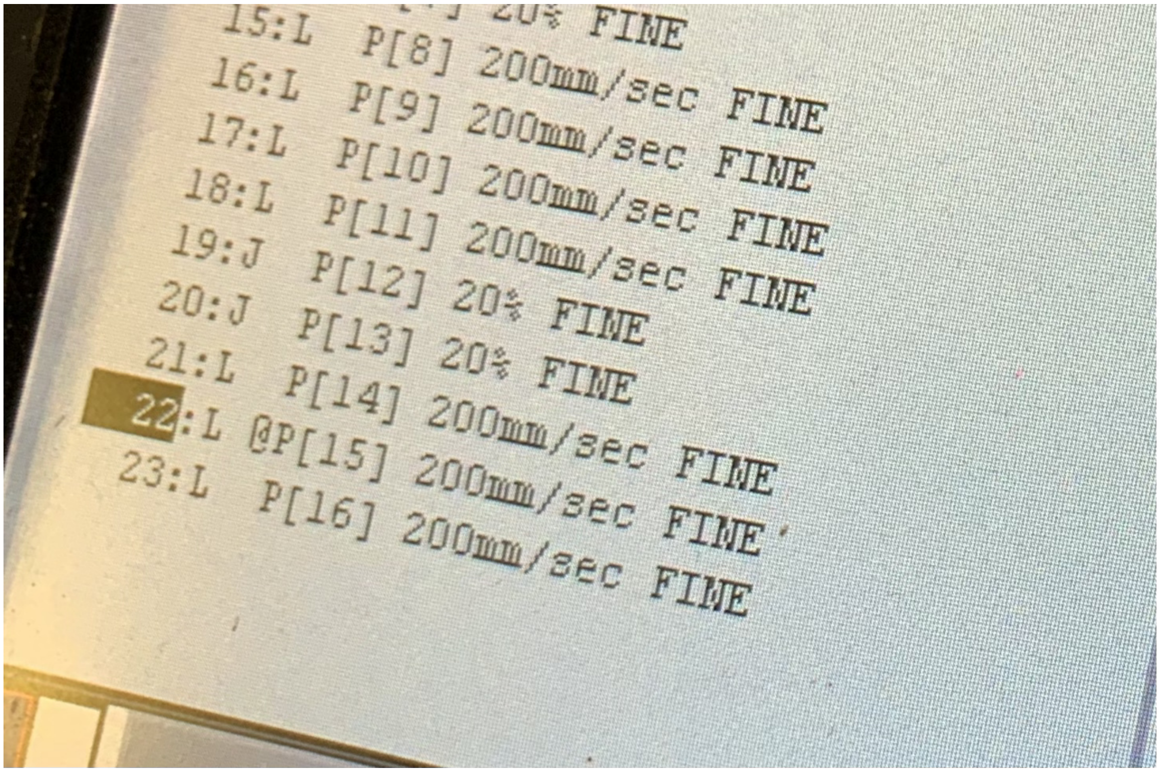
What became apparent as the potential causes were eliminated was that the robot was the last possibility of the erratic variation observed in the study. As stated earlier, all of the positional data was consistent and accurate through the iterations of software to the robot. However, the robot's teach pendant has an option that shows the position of the robot's TCP (in millimeters) and/or the position of the robots joints (in degrees of rotation). When this data was examined after the tests were concluded, one could see that the positional data the robot received was a mirror of the target data sent through both RoboDK and Roboguide (see Figure 5.4).



```
MAINFCZX
  P[15] GP1UF:0  UT:10
  J1      14.216 deg  J4    111.793 deg
  J2      28.515 deg  J5    -18.457 deg
  J3      -7.063 deg  J6    157.145 deg
Position Detail
   20:J  P[13] 20% FINE
   21:L  P[14] 200mm/sec FINE
   22:L  P[15] 200mm/sec FINE
   23:L  P[16] 200mm/sec FINE
   24:L  P[17] 200mm/sec FINE
   25:J  P[18] 20% FINE
  [End]
```

Figure 5.4: Line 22 of the FeatureCAM program shows that position 15 (P[15]) is set to execute joint rotational values enumerated above the "Position Detail" section of the screen. Such values are listed in order from J1 (joint 1) through J6 (Joint 6).

Although the data was an exact reflection of the data from the robot program and that the robot program had maintained its data values throughout the translation and uploading process, the robot's positional data showed the robot was not arriving at the given location. This was observed by going through the robot's teach pendant and selecting the "POSN" (position) button. When this button is selected, the display selected (multiple displays can be set up upon the same teach pendant's screen) will show the positional data for the robot specifically. As one can observe in Figure 5.5, the robot was at the same line of code and attempting to execute the same movement to the same position.

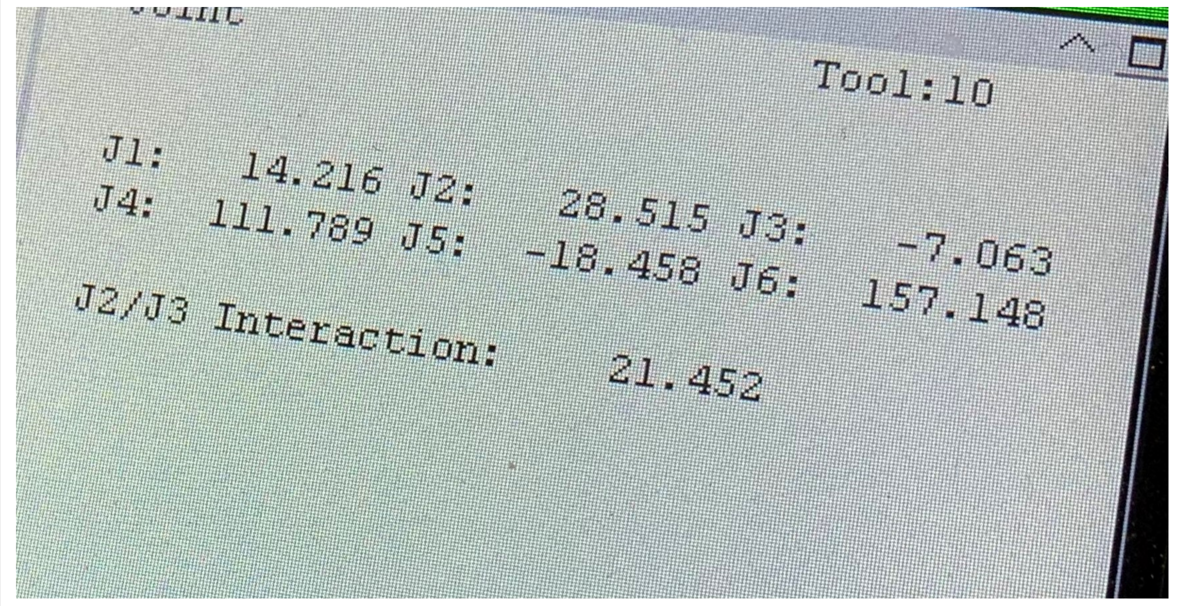


```
            20% FINE
  15:L  P[8] 200mm/sec FINE
  16:L  P[9] 200mm/sec FINE
  17:L  P[10] 200mm/sec FINE
  18:L  P[11] 200mm/sec FINE
  19:J  P[12] 20% FINE
  20:J  P[13] 20% FINE
  21:L  P[14] 200mm/sec FINE
  22:L @P[15] 200mm/sec FINE
  23:L  P[16] 200mm/sec FINE
```

Figure 5.5: Line 22 is attempting to make a linear move to P[15] (position 15) at 200mm/sec using FINE movement.

However, the robot's end position when the robot ceased movement after executing the instruction, was demonstrably different. This can be observed in Figure 5.6 where joints 4 through 6 were all at different locations to the targets enumerated in the code. It should be noted that this captured image of the execution of code was not taken during one of the experiment's tests. This was taken after concluding all of the experiment's iterations and the program was run in steps to observe how the robot executed the instructions.

As one can see in Figure 5.7, the robot's last three joints are clearly seen as two rotational joints and an elbow joint at the end of the robot arm.



```
                                        Tool:10
J1:    14.216 J2:    28.515 J3:    -7.063
J4:   111.789 J5:   -18.458 J6:   157.148
J2/J3 Interaction:      21.452
```

Figure 5.6: J4's value should be 111.793, J5's value should be -18.457 and J6's value should be 157.145.

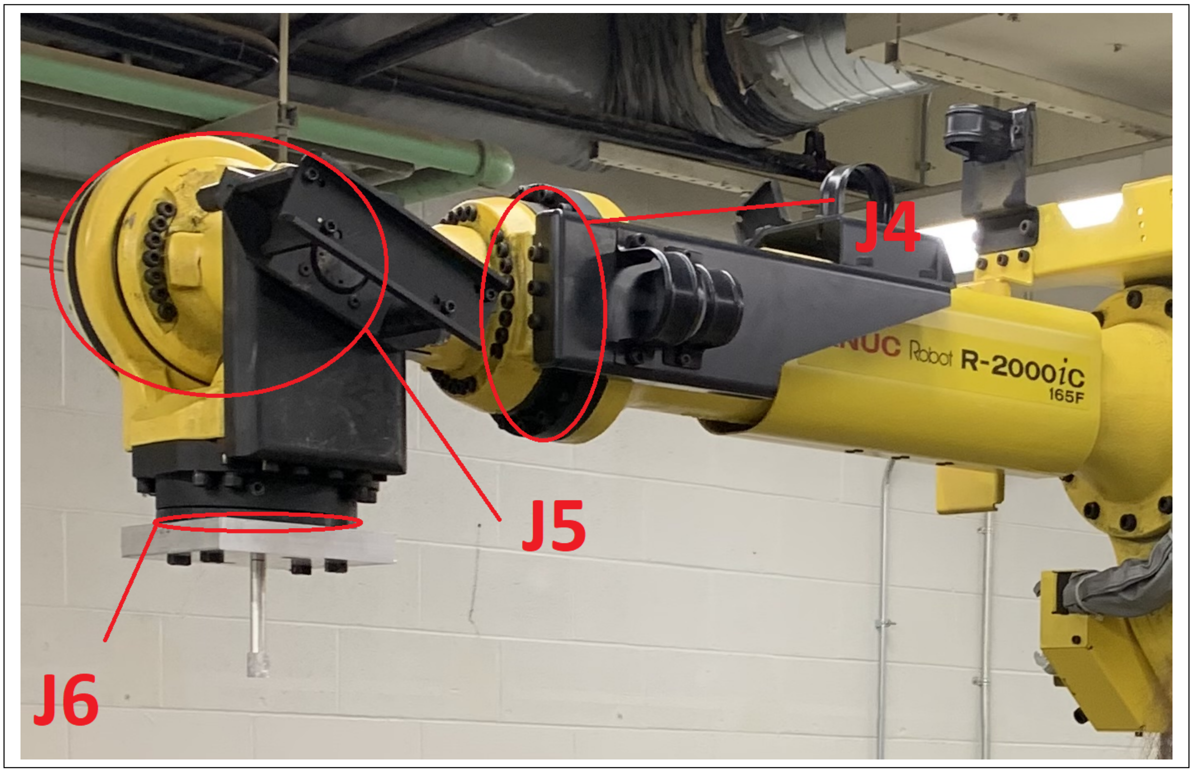


Figure 5.7: Joints 4 through 6 listed in red.

With these three joints being affected for the point of 0.5" travel, it is understandable how there was great potential for the inconsistent variance within the robot's movement. It should also be noted that there were other deviations from the program's positional coordinates by the robot. However, the most dramatic is the one listed above affecting the Y-axis 0.5" travel positional coordinate. An important caveat was that this information was gleaned from the FeatureCAM designed program. As one can see from the results, the FeatureCAM program had the closest results to the intended destinations.

5.4 Question of the Study

The goal of the research to this point has been to produce a low-cost and low-tech solution to test for variations between multiple iterations of offline programming software packages for industrial robots. Although there are various avenues whereby one can produce roboforming and/or robomachining products outside of the outlined FeatureCAM, RoboDK and Roboguide methods, the aforementioned grouping of software packages is the manner that one can bring such products to production utilizing Fanuc robots. ABB robots have built in ISF/SPIF options within their software package of RobotStudio, Fanuc does not at this time have such capability with Roboguide. Moreover, in as far as this concerns SMEs, Western Kentucky University represents an entity that can be considered an SME, being a regional, publicly-funded university that is not the largest in the state let alone the country. As such, an SME could potentially find itself in a circumstance where such Fanuc robots could be used to perform roboforming/robomachining operations and this research's findings would serve to advance such an entity in pursuing Industry 4.0 through these methods.

However, given that the physical robot utilized within this experiment failed to be able to produce adequately reliable results for this research, the experiment can only be classified as a failure under the terms it was proposed. An SME could not be reasonably expected to purchase a used robot and all the given software packages only to hope that the robot would be able to accurately and reliably perform roboforming/robomachining products. Moreover, if an SME were to pursue a roboforming/robomachining center for production within their business, a more reasonable choice would be an ABB type robot whose software package, RobotStudio, already includes such options within it.

5.5 Future Research

5.5.1 Program Degradation Testing

This research initially began as part of a larger research project into an XR robot-training module. The original parameters were to determine whether or not there was a degradation of the program code through iterations of software packages. Under that framework, a preliminary conclusion would be that there is not degradation in the program iterations as all positional coordinates matched throughout each iteration of software to the robot controller itself.

However, more tests of more complex programs should be tested to observe if deviations from the positional coordinates occur in the programs themselves. This would not require a physical testing method to check for deviations. However, to answer this question fully, it would require more than three different programs initiated at varying stages of software package iterations in order to give a confirmed result. In order to conduct such testing, one would merely need to generate the program within a CAM software, import the program through RoboDK and Roboguide and then check that all the positional coordinates match throughout the iterations of the software down to the robot controller.

5.5.2 Variation Testing

In as far as SMEs and the need to adapt, adopt and improve in the new global economy of Industry 4.0, a physical test will be required to ensure that the robot, the virtual twins, the user frames and the programs' iterations are all functional. Although in the prior paragraph one could be reasonably confident that their software packages do not produce conflicts, it is still essential to test for other problems. Given the issues produced by the physical robot in this research, robomachining with such a robot could potentially produce a large amount of broken mill-ends and damaged products as a result of inconsistent variation in the robot's TCP. Roboforming's process requires similar small increments of feed or else tearing/puncturing of the sheet can occur during ISF/SPIF processing. As such, a physical test is still required to ensure that all elements of the digitalized workspace work harmoniously to produce results that are consistent and within the level of accuracy required by the given process. Moreover, such research should be geared toward greater cost-effectiveness in order to enable SMEs to garner the benefits of innovative technology as the world slips into another recession.

Such a physical test would still require testing for the three axes of movement in order to detect variations. In preliminary testing for this research, a test was conducted using only Z-axis movement testing in three different locations. This manner of testing would have failed to highlight the issues that were brought to light with the experimentation finally used. A researcher could potentially assume that the robot had a higher level of accuracy as the Z-axis results of this research were some of the most consistent and accurate in the experiment. Major inconsistency and inaccuracy within the Y-axis results would not have been noticed or tested for. As such, it is vital to ensure that future testing observes those three axes of movement and following research could test for issues within the other areas of the workspace. This research utilized three dial indicators placed on the top of the workspace in the left corner. Future research could assemble three sets of three dial indicators along the top of the workspace to factor in the angle of the workspace so that the research concurrently marries the digital twin's accuracy with the physical workspace. Moreover, future research could also examine the remaining three axes of movement (roll, pitch and yaw) in order to determine the accuracy levels inherent in those movement types.

5.5.3 Localized Future Research for Western Kentucky University

In order to advance research goals within Western Kentucky University and WKU's XR Lab, assessing the full extent of the physical robot's limitations is necessary. This is not only important for this research and the WKU XR Lab's research, but also for any other research project and/or class that will utilize the robot for scholastic purposes. Given that the same three robot joints (J4, J5 and J6) were the joints whose values were not equivalent to the values dictated by the joint positional targets, it is possible that the robot has certain joint problems. Joint wear is caused by load, high temperatures, bad lubrication and defective parts (Kot, 2021, p. 2). Such wear typically manifests within the gearboxes of robots in both the gears and the bearings (Kot, 2021, p. 2). Additionally, wear can introduce debris within the joint that can mix into the joint's lubricant (Bittencourt, 2011, p. 1). Such additions can create additional friction and/or wear that can further exacerbate joint issues already being experienced. Given that the robot was used for some years lifting car frames, it is quite possible the inconsistent results were caused by large loads being manipulated by the robot over its years of service. In order to ensure the robot can be utilized appropriately at WKU, it is important to isolate the joints that are most affected and diagnose to what extent they are affected, so that a resolution plan may be devised. Potential resolutions could include simply isolating the affected joints during precision movements or conducting component replacements. However, cost may prohibit repairs and a total isolation of affected joints may not be possible for all programs regardless of the precision required by their movements. Nonetheless, ascertaining the joints that are affected, to what degree they are affected and having contingency plans in place to ameliorate the problems, are still important for future work with the robot.

Methods whereby one can ascertain the joints that are most affected can be accomplished by running movement programs and examining the robot's positional data in comparison to the program positional coordinates. As long as the positional coordinates are joint coordinates, it would be simple to see the joints that have deviations and by how much. Repeats of the program movements can allow for one to ascertain the consistency of the deviations.

5.6 Summary

The goal of this research was to produce a low-cost, low-tech option for assessing whether or not program degradation has occurred between multiple iterations of offline programming (OLP) software. Due to the inconsistent nature of the results garnered within the experiment, the research cannot formally be assessed to have brought that research goal to fruition. Although the robot utilized had relatively close tolerances to new/refurbished robots' repeatable accuracy, the inconsistent variation of the Y-axis tests render the research inconclusive to that goal. Given the potential costs of the system utilized, it is most likely that an SME seeking to employ Industry 4.0 technologies would pursue a robot option with higher accuracy and OLP software that incorporates robomachining/roboforming technology within the same software suite.

Although the research conducted resulted in failure, it should be reiterated that such research does not exist in prior literature. Testing for variations in subsequent iterations of OLP software packages is not a focus of other research to this point. This is despite the fact that Western Kentucky University, a relative SME, is in the position where the utilization of the current Fanuc robot for ISF/SPIF research will require software combinations like the one used in this research. As such, further study should look to fill in this research gap so as to ensure that SMEs, universities and others will have more cost effective manners of working with and educating others on the new and innovative technology that is emerging and being demanded in the 21st century's high-tech and global economy.

REFERENCES

Afzal, A., C. L. Goues, M. Hilton, and C. S. Timperley (2020, October). A study on challenges of testing robotic systems. In *2020 IEEE 13th International Conference on Software Testing, Validation and Verification (ICST)*. IEEE.

American-National-Standards-Institute. (1998). Manipulating industrial robots - performance criteria and related test methods. *ISO 9283:1998*.

American-National-Standards-Institute. (2022).

Austin, A. (2014). Process capability in a computer integrated manufacturing cell. *Computer Integrated Manufacturing Cell*.

Autodesk-Inc. (2021). Featurecam: Cam automation for cnc programming.

Bahloul, R., H. Arfa, and H. BelHadjSalah (2014, September). A study on optimal design of process parameters in single point incremental forming of sheet metal by combining Box–Behnken design of experiments, response surface methods and genetic algorithms. *Int. J. Adv. Manuf. Technol. 74*(1-4), 163–185.

Banabic, R. (2015). *Techniques for identifying elusive corner-case bugs in systems software*.

Bi, Z. M., Z. Miao, B. Zhang, and C. W. J. Zhang (2020, June). The state of the art of testing standards for integrated robotic systems. *Robot. Comput. Integr. Manuf. 63*(101893), 101893.

Bittencourt, A. C., P. Axelsson, Y. Jung, and T. Brogårdh (2011, January). Modeling and identification of wear in a robot joint under temperature uncertainties. *IFAC proc. vol. 44*(1), 10293–10299.

Burghardt, A., D. Szybicki, P. Gierlak, K. Kurc, P. Pietruś, and R. Cygan (2020, January). Programming of industrial robots using virtual reality and digital twins. *Appl. Sci. (Basel) 10*(2), 486.

Bârsan, A., M. O. Popp, G. P. Rusu, and A. I. Maroșan (2021). Robot-based incremental sheet forming – the tool path planning. *IOP Conference Series: Materials Science and Engineering, 1009*.

Choi, H., C. Crump, C. Duriez, A. Elmquist, G. Hager, D. Han, F. Hearl, J. Hodgins, A. Jain, F. Leve, C. Li, F. Meier, D. Negrut, L. Righetti, A. Rodriguez, J. Tan, and J. Trinkle (2021, January). On the use of simulation in robotics: Opportunities, challenges, and suggestions for moving forward. *Proc. Natl. Acad. Sci. U. S. A. 118*(1), e1907856118.

Deans, M. (2021). What is cam (computer-aided manufacturing)? *Fusion 360 Blog*.

Dennis, P. (2017). *Lean production simplified: A plain-language guide to the world's most powerful production system*. Boca Raton, FL: Taylor & Francis Group.

Erdogmus, H., J. Favaro, and W. Strigel (2004, May). Return on investment. *IEEE Softw. 21*(3), 18–22.

FANUC-America-Corporation. (2021a).

FANUC-America-Corporation. (2021b). Roboguide robot simulation software: Fanuc america.

Garbev, A. and A. Atanassov (2020, October). Comparative analysis of RoboDK and robot operating system for solving diagnostics tasks in off-line programming. In *2020 International Conference Automatics and Informatics (ICAI)*. IEEE.

Ghobakhloo, M. and M. Fathi (2019, July). Corporate survival in industry 4.0 era: the enabling role of lean-digitized manufacturing. *J. Manuf. Technol. Manag. 31*(1), 1–30.

Goldratt, E. (2006, August). *Goal: A process of ongoing improvement - revised 3rd edition*. Highbridge Company.

Józwik, J., D. Ostrowski, P. Jarosz, and D. Mika (2016, December). Industrial robot repeatability testing with high speed camera phantom v2511. *Adv. Sci. Technol. Res. J. 10*(32), 86–96.

Klitou, D., J. Conrads, M. Rasmussen, L. Probst, and B. Pedersen (2017). Industry 4.0 initiatives in the cosme countries. *Digital Transformation Monitor*.

Kot, T., Z. Bobovský, A. Vysocký, V. Krys, J. Šafařík, and R. Ružarovský (2021, June). Method for robot manipulator joint wear reduction by finding the optimal robot placement in a robotic cell. *Appl. Sci. (Basel) 11*(12), 5398.

Łobos, K., V. Malátek, and M. Szewczyk (2020). Management practices in area of human resources and monitoring results as determinants of SME's success in poland and the czech republic. *E+M Ekon. Manag. 23*(2), 114–125.

Moynihan, T. (2014). A slo-mo camera with an insanely high frame rate (and price tag). *Wired*.

Oh, Y. T. (2019, November). Study of orientation error on robot end effector and volumetric error of articulated robot. *Appl. Sci. (Basel) 9*(23), 5149.

Pagliosa, M., G. Tortorella, and J. C. E. Ferreira (2019, October). Industry 4.0 and lean manufacturing. *J. Manuf. Technol. Manag. 32*(3), 543–569.

Pan, J.-N. (2006, August). Evaluating the gauge repeatability and reproducibility for different industries. *Qual. Quant. 40*(4), 499–518.

Radwell-International-LLC. (2021).

RoboDK-Inc. (2021).

RobotWorx-Inc. (2022).

Ross, L. T., S. W. Fardo, and M. F. Walach (2018). *Industrial robotics fundamentals: Theory and applications*. The Goodheart-Willcox Company, Inc.

Sanders, A., C. Elangeswaran, and J. Wulfsberg (2016, September). Industry 4.0 implies lean manufacturing: Research activities in industry 4.0 function as enablers for lean manufacturing. *J. Ind. Eng. Manag. 9*(3), 811.

Slavkovic, N. R., D. S. Milutinovic, and M. M. Glavonjic (2014, February). A method for off-line compensation of cutting force-induced errors in robotic machining by tool path modification. *Int. J. Adv. Manuf. Technol. 70*(9-12), 2083–2096.

Ustundag, A. and E. Cevikcan (2017, September). *Industry 4.0: Managing the digital transformation* (1 ed.). Springer Series in Advanced Manufacturing. Basel, Switzerland: Springer International Publishing.

Vocetka, M., R. Huňady, M. Hagara, Z. Bobovský, T. Kot, and V. Krys (2020, December). Influence of the approach direction on the repeatability of an industrial robot. *Appl. Sci. (Basel) 10*(23), 8714.

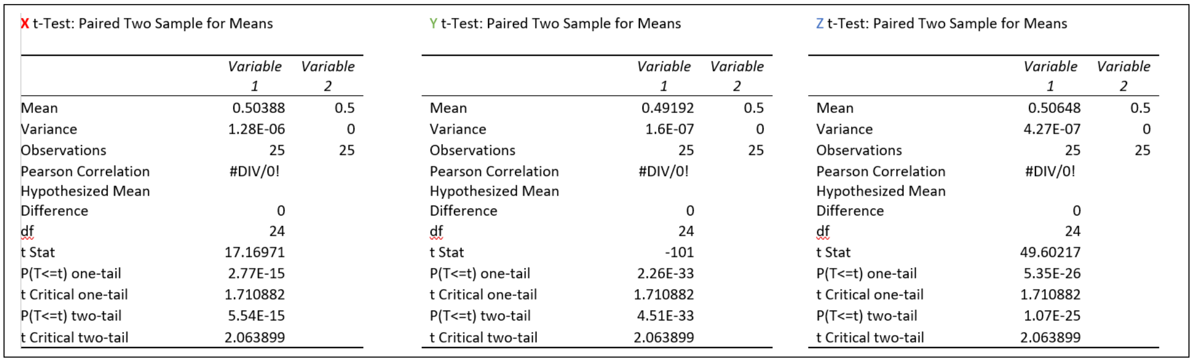
Vuola, A. and R. Tuokko (2012). Accuracy measurements of miniature robot using optical CMM. In *Precision Assembly Technologies and Systems*, pp. 126–133. Berlin, Heidelberg: Springer Berlin Heidelberg.

Womack, J. P. and D. T. Jones (2003). *Lean Thinking: Banish Waste and Create Wealth in Your Corporation*. New York, NY: Free Press.

Zhuang, C., T. Miao, J. Liu, and H. Xiong (2021, April). The connotation of digital twin, and the construction and application method of shop-floor digital twin. *Robot. Comput. Integr. Manuf. 68*(102075), 102075.

Appendices

Appendix A



X t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.50388	0.5
Variance	1.28E-06	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	17.16971	
P(T<=t) one-tail	2.77E-15	
t Critical one-tail	1.710882	
P(T<=t) two-tail	5.54E-15	
t Critical two-tail	2.063899	

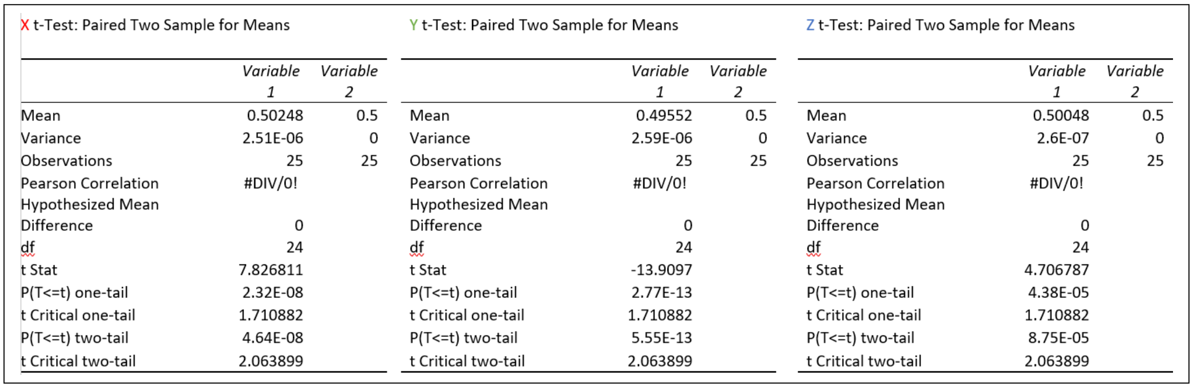
Y t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.49192	0.5
Variance	1.6E-07	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	-101	
P(T<=t) one-tail	2.26E-33	
t Critical one-tail	1.710882	
P(T<=t) two-tail	4.51E-33	
t Critical two-tail	2.063899	

Z t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.50648	0.5
Variance	4.27E-07	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	49.60217	
P(T<=t) one-tail	5.35E-26	
t Critical one-tail	1.710882	
P(T<=t) two-tail	1.07E-25	
t Critical two-tail	2.063899	

Figure A.1: FeatureCAM X, Y and Z axes for 0.5" Travel t-Test Table.



X t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.50248	0.5
Variance	2.51E-06	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	7.826811	
P(T<=t) one-tail	2.32E-08	
t Critical one-tail	1.710882	
P(T<=t) two-tail	4.64E-08	
t Critical two-tail	2.063899	

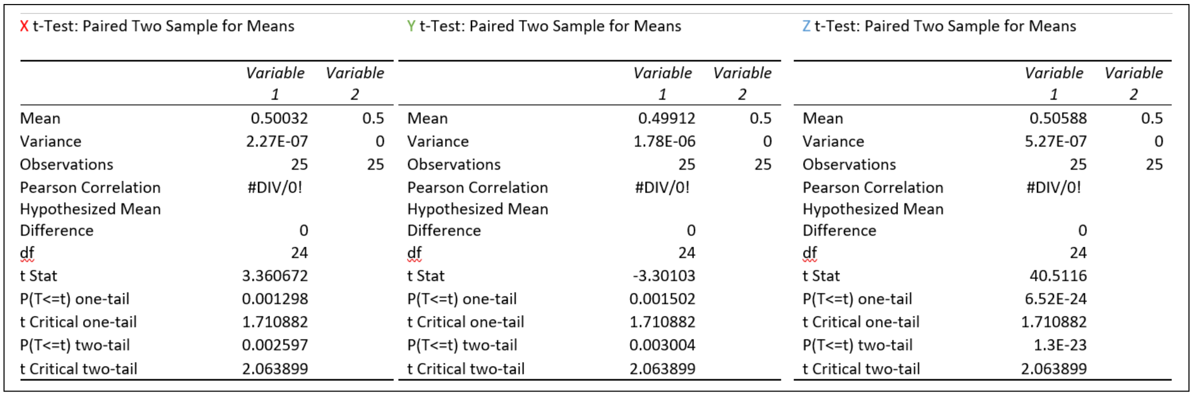
Y t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.49552	0.5
Variance	2.59E-06	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	-13.9097	
P(T<=t) one-tail	2.77E-13	
t Critical one-tail	1.710882	
P(T<=t) two-tail	5.55E-13	
t Critical two-tail	2.063899	

Z t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.50048	0.5
Variance	2.6E-07	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	4.706787	
P(T<=t) one-tail	4.38E-05	
t Critical one-tail	1.710882	
P(T<=t) two-tail	8.75E-05	
t Critical two-tail	2.063899	

Figure A.2: FeatureCAM X, Y and Z axes for 0" Travel t-Test Table.



X t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.50032	0.5
Variance	2.27E-07	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	3.360672	
P(T<=t) one-tail	0.001298	
t Critical one-tail	1.710882	
P(T<=t) two-tail	0.002597	
t Critical two-tail	2.063899	

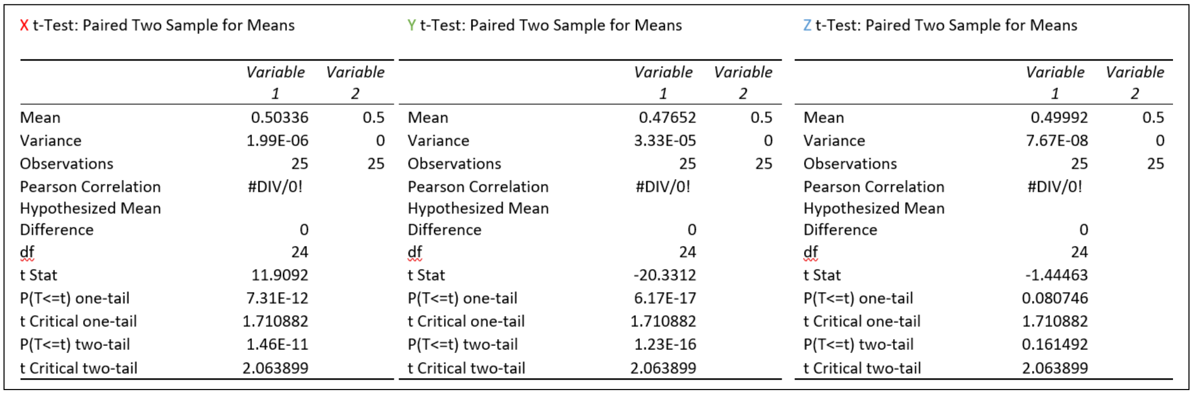
Y t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.49912	0.5
Variance	1.78E-06	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	-3.30103	
P(T<=t) one-tail	0.001502	
t Critical one-tail	1.710882	
P(T<=t) two-tail	0.003004	
t Critical two-tail	2.063899	

Z t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.50588	0.5
Variance	5.27E-07	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	40.5116	
P(T<=t) one-tail	6.52E-24	
t Critical one-tail	1.710882	
P(T<=t) two-tail	1.3E-23	
t Critical two-tail	2.063899	

Figure A.3: RoboDK X, Y and Z axes for 0.5" Travel t-Test Table.



X t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.50336	0.5
Variance	1.99E-06	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	11.9092	
P(T<=t) one-tail	7.31E-12	
t Critical one-tail	1.710882	
P(T<=t) two-tail	1.46E-11	
t Critical two-tail	2.063899	

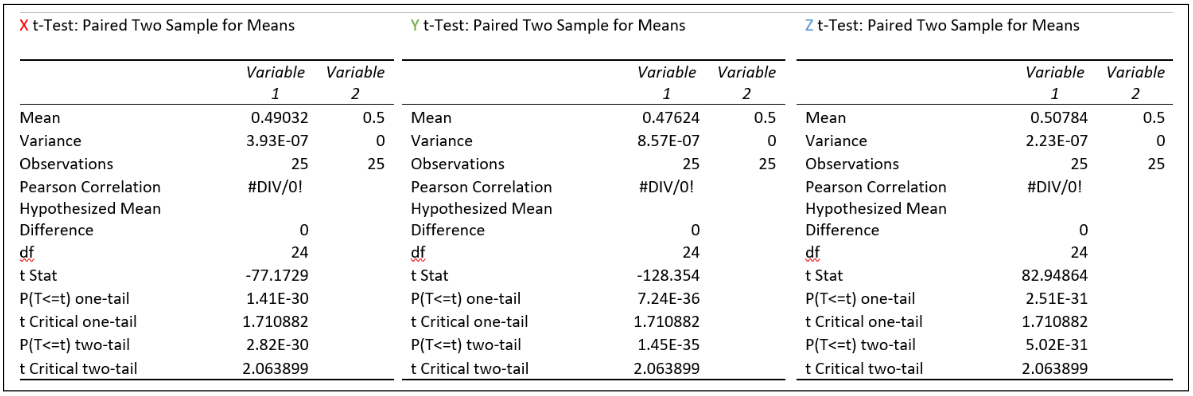
Y t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.47652	0.5
Variance	3.33E-05	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	-20.3312	
P(T<=t) one-tail	6.17E-17	
t Critical one-tail	1.710882	
P(T<=t) two-tail	1.23E-16	
t Critical two-tail	2.063899	

Z t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.49992	0.5
Variance	7.67E-08	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	-1.44463	
P(T<=t) one-tail	0.080746	
t Critical one-tail	1.710882	
P(T<=t) two-tail	0.161492	
t Critical two-tail	2.063899	

Figure A.4: RoboDK X, Y and Z axes for 0" Travel t-Test Table.



X t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.49032	0.5
Variance	3.93E-07	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	-77.1729	
P(T<=t) one-tail	1.41E-30	
t Critical one-tail	1.710882	
P(T<=t) two-tail	2.82E-30	
t Critical two-tail	2.063899	

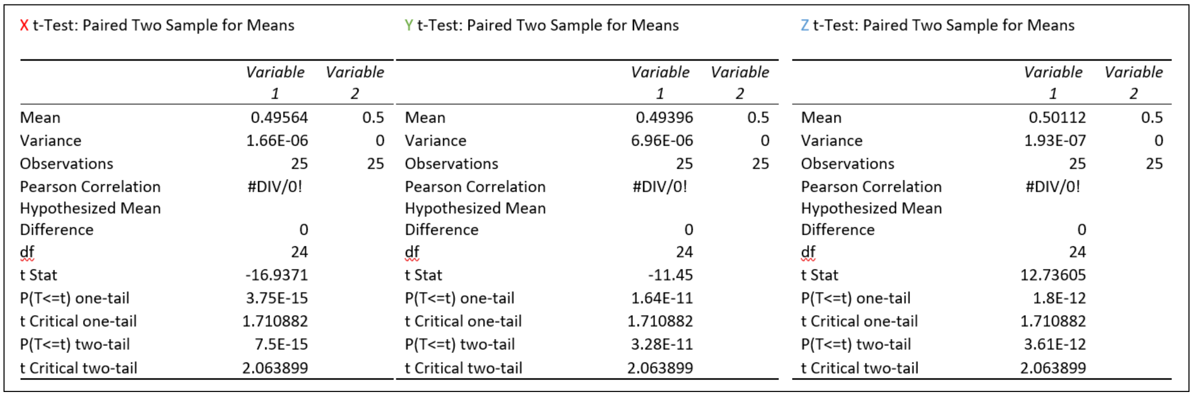
Y t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.47624	0.5
Variance	8.57E-07	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	-128.354	
P(T<=t) one-tail	7.24E-36	
t Critical one-tail	1.710882	
P(T<=t) two-tail	1.45E-35	
t Critical two-tail	2.063899	

Z t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.50784	0.5
Variance	2.23E-07	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	82.94864	
P(T<=t) one-tail	2.51E-31	
t Critical one-tail	1.710882	
P(T<=t) two-tail	5.02E-31	
t Critical two-tail	2.063899	

Figure A.5: Roboguide X, Y and Z axes for 0.5" Travel t-Test Table.



X t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.49564	0.5
Variance	1.66E-06	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	-16.9371	
P(T<=t) one-tail	3.75E-15	
t Critical one-tail	1.710882	
P(T<=t) two-tail	7.5E-15	
t Critical two-tail	2.063899	

Y t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.49396	0.5
Variance	6.96E-06	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	-11.45	
P(T<=t) one-tail	1.64E-11	
t Critical one-tail	1.710882	
P(T<=t) two-tail	3.28E-11	
t Critical two-tail	2.063899	

Z t-Test: Paired Two Sample for Means

	Variable 1	*Variable 2*
Mean	0.50112	0.5
Variance	1.93E-07	0
Observations	25	25
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	24	
t Stat	12.73605	
P(T<=t) one-tail	1.8E-12	
t Critical one-tail	1.710882	
P(T<=t) two-tail	3.61E-12	
t Critical two-tail	2.063899	

Figure A.6: Roboguide X, Y and Z axes for 0" Travel t-Test Table.