

**TOOL ACCESSIBILITY WITH PATH AND MOTION PLANNING FOR
ROBOTIC DRILLING AND RIVETING**

By

David Dakdouk
Bachelor of Aerospace Engineering,
Ryerson University (2012)

A thesis presented to Ryerson University
in partial fulfilment of the requirement for the degree of
Master of Applied Science
In the Program of
Aerospace Engineering

Toronto, Ontario, Canada, 2016

© David Dakdouk, 2016

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

ABSTRACT

TOOL ACCESSIBILITY WITH PATH AND MOTION PLANNING FOR ROBOTIC DRILLING AND RIVETING

David Dakdouk

A thesis for the degree of

Master of Applied Science, 2016

Department of Aerospace Engineering, Ryerson University

Robotic applications in aerospace manufacturing and aircraft assembly today are limited. This is because most of the aircraft parts are relatively small or have complex shapes that make tasks like robotic drilling and riveting more challenging. These challenges include tool accessibility, path planning, and motion planning.

In this thesis, a process methodology was developed to overcome the tool accessibility challenges facing robotic drilling and riveting for aircraft parts. The tool accessibility was analyzed based on the Global Accessibility Area and the Global Accessibility Volume to determine the accessible boundaries for parts with zero, one and two surfaces curvatures. The path planning was optimized based on the shortest distance, least number of steps, and minimal tool orientation change. The motion planning was optimized based on the s-curve using the robot's maximum velocity and acceleration for minimum cycle time and maximum production rate. A software application was developed to simulate the tasks.

ACKNOWLEDGEMENTS

A very special thank-you to my supervisor Dr. Fengfeng (Jeff) Xi for the excellent supervision, support, guidance and kindly encouragement throughout the past years since my undergraduate. His motivation, innovative ideas, and strong academic background was a key factor for the progress of this thesis and a good support whenever I needed help. It has been my pleasure to work under his supervision.

I am also most grateful to Dr. Yu Lin for his strong academic background and knowledge with the robotic riveting project and research. He was a great source of information related to the robotic calibration.

Finally, I would like to thank my family and close friends sincerely for their encouragement, motivation, support, and understanding over the school years.

David Dakdouk, August 2016

CONTENTS

AUTHOR'S DECLARATION	ii
ABSTRACT	iii
ACKNOWLEDGEMENTS	iv
CONTENTS	v
LIST OF TABLES	ix
LIST OF FIGURES	x
NOMENCLATURE	xv
CHAPTER 1:INTRODUCTION	1
1.1. Background	1
1.1.1. Assembly Types in Aerospace Manufacturing.....	2
1.1.2. Robotic Automation in Aerospace Manufacturing.....	5
1.1.3. System Description.....	6
1.2. Problem Formulation and Objective	9
1.3. Outline of the Thesis	12
CHAPTER 2:LITERATURE REVIEW	14
2.1. Tool Accessibility	15
2.1.1. Tools Selection and Customization for Aerospace Applications	15
2.1.2. Tool Accessibility Approaches.....	17
2.1.2.1. Configuration Space (C-space) Approach.....	18
2.1.2.2. Visibility Map (VMap) Approach	19
2.1.2.3. Accessibility Cone Approach	19
2.2. Robotic Path Planning.....	21
2.2.1. Obstacle Avoidance	22
2.2.1.1. Visibility Graph Method.....	23

2.2.1.2.	Cell Decomposition Method.....	24
2.2.1.3.	Artificial Potential Method.....	24
2.2.2.	Optimal Path Planning.....	25
2.2.3.	Techniques of Programming Industrial Robots.....	26
2.2.3.1.	Robot Teaching Approach.....	26
2.2.3.2.	Real-Time Scanning and Guidance Approach.....	27
2.2.3.3.	Virtual Environment Simulation Approach.....	27
2.2.4.	Additional Consideration for Path Planning Optimization.....	28
2.3.	Robotic Motion Planning.....	29
2.3.1.	Motion Curves.....	30
2.3.1.1.	Trapezoidal Curve Model.....	31
2.3.1.2.	S-curve Model.....	32
2.3.2.	Motion Optimisation Consideration.....	33
2.3.2.1.	Minimum Execution Time.....	33
2.3.2.2.	Minimum Actuator Effort.....	34
2.3.2.3.	Minimum Jerk.....	34
2.3.2.4.	Minimum Cycle Time.....	35
2.4.	Application of Riveting Automation in Aircraft Design.....	35
2.4.1.	Riveting Automation.....	37
2.4.1.1.	MTorres™ Automated Riveting Systems.....	38
2.4.1.2.	Gemcor™ G100 Frame Fastener System.....	38
2.4.1.3.	Electroimpact™ E7000 Riveting Cell.....	39
2.4.1.4.	KUKA™ Drilling and Riveting Systems.....	40
2.4.2.	Challenges for Automated Riveting Systems.....	41
CHAPTER 3: ACCESSIBILITY ANALYSIS.....		43

3.1.	Accessibility Definitions	43
3.2.	Geometrical Classification	46
3.2.1.	Tool Geometry	47
3.2.2.	Part Geometry	48
3.3.	Global Accessibility Area (GAA)	50
3.3.1.	Geometrical Mapping	52
3.3.2.	GAA with Fixed Tool Orientation	53
3.3.3.	GAA with Variable Tool Orientation	58
3.4.	Global Accessibility Volume (GAV)	60
3.4.1.	Geometrical Volume Extraction	63
3.4.2.	GAV with Fixed Tool Orientation	64
3.4.3.	GAV with Variable Tool Orientation	68
3.5.	Accessibility Percentage	72
CHAPTER 4: PATH PLANNING		73
4.1.	Path Planning for Drilling and Riveting	73
4.2.	Path Selection Process for Robotic Systems	74
4.2.1.	Shortest Travel Distance	74
4.2.2.	Target Approachability Planning	76
4.2.3.	Variable Tool Orientation	80
4.3.	Path Planning Algorithm for Collision-Free Path	85
CHAPTER 5: MOTION ANALYSIS		89
5.1.	Motion Curves	89
5.2.	Robotic Cycle Time Analysis	100
CHAPTER 6: SIMULATION		105
6.1.	Algorithm Overview	105

6.2.	Simulation Test Cases	110
6.2.1.	2D Test Cases	110
6.2.2.	3D Test Cases	113
6.3.	Algorithm Implementation for Aircraft Assembly.....	116
6.4.	Test Cases of Aircraft Assembly Simulation	124
6.4.1.	Test Case with Wing Section.....	124
6.4.2.	Test Case with Fuselage Section	126
6.4.3.	Test Case with Cockpit Section.....	127
CHAPTER 7: CONCLUSION AND FUTURE WORK.....		130
7.1.	Conclusion.....	130
7.2.	Contribution	131
7.3.	Future Work	132
APPENDICES.....		134
Appendix A - Standard Riveting Calculations.....		134
A.1.	Rivet Patterns and Spacing Calculations.....	136
A.2.	Complex Riveting Patterns.....	139
Appendix B - Coordinate's Transformations.....		142
Appendix C - Software Application Representation.....		144
Appendix D - Robot Range Information		150
Appendix E - Engineering Drawings of the Robot.....		151
Appendix F - Mathematical Operations.....		152
REFERENCES		153

LIST OF TABLES

Table 3-1: The Faces, Sides, and Vertices Count for Some Geometries	47
Table 3-2: Tool Geometrical Classification.....	48
Table 3-3: Aircraft Part Geometrical Classification	49
Table 3-4: Crowdedness and Accessibility Check for each Aircraft Part	49
Table 6-1: Test Cases Sample of Inputs and Outputs for 2D	110
Table 6-2: Test Cases Sample of Inputs and Outputs for 3D	113
Table 6-3: Results for the Test Case with Skin Panel Section of a Wing.....	125
Table 6-4: Results for the Test Case with Fuselage Section.....	127
Table 6-5: Results for the Test Case with Cockpit Section	128
Table D-1: The Extreme Positions of the Robot Arms IRB 4400 45 kg [50]	150
Table D-2: Types and Ranges of the Robot Motion in Every Axis [50]	150

LIST OF FIGURES

Figure 1-1: a) Manual Riveting [5], b) Semi-automated Riveting [7], and c) Automated Riveting Systems [9].....	3
Figure 1-2: a) Manual Drilling [5], b) Semi-automated Drilling [10], and c) Automated Drilling Systems [11].....	5
Figure 1-3: Overview of the Robotic System	7
Figure 1-4: Overview of the Robot Tool System.....	8
Figure 1-5: Overview of the CNC Bucking Bar System	8
Figure 1-6: Overview of the RobotStudio™ Interface	9
Figure 2-1: A Sample of Drills Types Used in the Aerospace Industry [19]	16
Figure 2-2: Pneumatic Angled Drills [19]	17
Figure 2-3: Modified Bucking Bars to Fit in Different Flanges Forms [20]	17
Figure 2-4: C-Space Find-Path and Find-Space Problems in 2D [21].....	18
Figure 2-5: GAC Construction at the Fastener Location [17]	20
Figure 2-6: GAC Application, a) The 3D Representation, b) The Map Representation [17].....	20
Figure 2-7: Path Planning Problem Terminologies	22
Figure 2-8: The Cell Decomposition Approach.....	24
Figure 2-9: Trapezoidal Curve and S-Curve Motion Comparison [30].....	30
Figure 2-10: Overview of the Commercial Aircraft Main Components [28].....	36
Figure 2-11: Typical Skin-Stringer Panels [28].....	36
Figure 2-12: Gemcor™ G100 Frame Fastener [39].....	39
Figure 2-13: Electroimpact™ E7000 High-Speed CNC Fuselage Riveting Cell [40]	39
Figure 2-14: KUKA™ Fuselage Automated Upright Build (FAUB) [41].....	40

Figure 2-15: KUKA™ Mobile Riveting Platforms (MRPs) [42] & [43]	41
Figure 2-16: Riveted Skin Panel [44]	41
Figure 3-1: Accessibility Representation	44
Figure 3-2: Accessibility Representation in 2D	50
Figure 3-3: Global Accessibility Area Representation	51
Figure 3-4: Front Projection of a Curved Surface	55
Figure 3-5: Side Projection of a Curved Surface	55
Figure 3-6: Close View of the Z-beam on the Side Projection of a Curved Surface.....	55
Figure 3-7: Tool in GAA with Boundary Limits	56
Figure 3-8: Linear Intersection for: a) No Solution, b) One Solution, c) Infinite Solutions	57
Figure 3-9: Accessibility Representation in 3D	60
Figure 3-10: Inaccessible Zones Representation in 3D with Top, Side, and Left Views	61
Figure 3-11: Global Accessibility Volume Representation	62
Figure 3-12: Global Accessibility Volume of the Accessible Region for Different Beam Shapes	65
Figure 3-13: Tool in GAV with Boundary Limits	66
Figure 3-14: Faces Intersection for: a) No Solution, b) Linear Solution, c) Infinite Solutions	67
Figure 3-15: Euler Angles Representation.....	69
Figure 3-16: The Three Point Approach Diagram	70
Figure 4-1: Shortest Path Problem: a) Linear Path, b) Chain Path, c) Zig-zag Path	75
Figure 4-2: Predicted Path Maneuver for the Tool Around the Obstacle	77
Figure 4-3: Basic Target Approachability Layout for Drilling and Riveting Pattern.....	77
Figure 4-4: Target Approachability Planning Pattern Order	78

Figure 4-5: Target Approachability Planning Pattern for Higher Orders	79
Figure 4-6: Curvature Presentation Diagram	80
Figure 4-7: Flowchart Diagram of the Path Selection and Optimization Process	86
Figure 4-8: Flowchart Representation of the Accessibility Validation along the Optimal Path ..	87
Figure 5-1: Generated Path Representation of Zero Order Approachability	91
Figure 5-2: Graphical Representation of Displacement.....	92
Figure 5-3: Graphical Representation of Velocity	93
Figure 5-4: Graphical Representation of Acceleration	94
Figure 5-5: Graphical Representation of Jerk.....	97
Figure 5-6: Total Cycle Time and Task Cycle Time Representation	101
Figure 5-7: Path Process Segmentation for Motion Planning.....	102
Figure 5-8: Flowchart Diagram of the Drilling Routine Cycle.....	103
Figure 5-9: Flowchart Diagram of the Riveting Routine Cycle	103
Figure 6-1: Algorithm Flowchart Representation of the Matlab Scripts for 2D and 3D	107
Figure 6-2: Matlab Riveting Process Check 2D	108
Figure 6-3: Matlab Riveting Process Check 3D	108
Figure 6-4: Matlab Accessibility Check 2D	109
Figure 6-5: Matlab Accessibility Check 3D	109
Figure 6-6: Matlab 2D Test Case 1	112
Figure 6-7: Matlab 2D Test Case 2.....	112
Figure 6-8: Matlab 2D Test Case 3.....	112
Figure 6-9: Matlab 3D Test Case 1 View at 40°	114
Figure 6-10: Matlab 3D Test Case 1 View at 20°	115

Figure 6-11: Matlab 3D Test Case 2.....	115
Figure 6-12: Matlab 3D Test Case 3.....	115
Figure 6-13: General Overview Flowchart Representation of the Add-In Code.....	116
Figure 6-14: Logic Flowchart Representation of the Add-In Block A.....	117
Figure 6-15: Logic Flowchart Representation of the Add-In Block B.....	118
Figure 6-16: Logic Flowchart Representation of the Add-In Block C.....	120
Figure 6-17: ABB Riveting Workstation.....	121
Figure 6-18: Riveting a Fuselage Section in Robotstudio.....	122
Figure 6-19: Riveting Planner [®] Add-In Tool in RobotStudio.....	122
Figure 6-20: RobotStudio Collision Avoidance Validation.....	123
Figure 6-21: Test Case with Wing Section.....	124
Figure 6-22: Test Case with Skin Panel Section of a Wing.....	124
Figure 6-23: Test Case with Skin Panel Section of a Wing ABB Workstation View.....	125
Figure 6-24: Test Case with Fuselage Section.....	126
Figure 6-25: Test Case with Fuselage Section ABB Workstation View.....	126
Figure 6-26: Test Case with Cockpit Section.....	128
Figure 6-27: Test Case with Cockpit Section ABB Workstation View.....	128
Figure A-1: Rounded Solid-head Rivet.....	135
Figure A-2: Layout Patterns for: a) Single Riveted Lap Joint, b) Double Chain Riveted Lap Joint, c) Double Zig-zag Riveted Lap Joint [54].....	136
Figure A-3: Layout Patterns for: a) Triple Chain Riveted Lap Joint, b) Triple Zig-zag Riveted Lap Joint [54].....	136
Figure A-4: Diagram of the Calculated Variables.....	137

Figure A-5: Riveting Diagram [5]	137
Figure A-6: Typical Layout for a Patch Repair for Damaged Fuselage Skin [20]	139
Figure A-7: Sample of Riveted Stringer with a Supportive Schematic Representation [51]	141
Figure B-1: Coordinates Representation.....	142
Figure C-1: Main window of the Riveting Planner [®] Software Application Add-in in RobotStudio [™]	144
Figure C-2: Settings and Calibration Window.....	145
Figure C-3: Three points Approach Calibration Window	145
Figure C-4: CNC Calibration Window	146
Figure C-5: Tools Dimensions Window	146
Figure C-6: Riveting Planner Window	147
Figure C-7: Curvature Input Window for One Curvature Selection.....	147
Figure C-8: Curvature Input Window for Two Curvature Selection.....	148
Figure C-9: Obstacles Window.....	148
Figure C-10: Riveting Planner Window with Results	149
Figure C-11: Riveting Repairs Window	149
Figure E-1: Working Range and Load Diagram for the Robot Arms IRB 4400 45 kg [50]	151
Figure E-2: Illustration of Extreme Positions of the Robot Arms IRB 4400 45 kg [50].....	151

NOMENCLATURE

Symbols	Definition
α, β, γ	Euler Angles for Roll, Pitch, and Yaw
$\hat{\mathbf{i}}, \hat{\mathbf{j}}, \hat{\mathbf{k}}$	Unit Vectors
n, m	Number of an Item
u, v, w	Unit Vectors
ρ, ϕ, h	Cylindrical Coordinate System for Radius, Azimuth, and Height
r, θ, φ	Spherical Coordinate System for Radius, Colatitude, and Longitude
X, Y, Z	Cartesian Coordinate System
x, y, z	Local Cartesian Coordinate System
A	Area
a	Acceleration
B	Distance Between Rivets
b	Point in local Coordinates
C	Local Rotation Matrix
c	Surface Curvature
ζ	Configuration Space (<i>C-Space</i>)
D	Diameter
\mathfrak{D}	Displacement
d	Distance
E	Local Rotation Matrix
ξ	Free Configuration Space (<i>C-Free</i>)
F	Targets
f	Tool Faces
G	Tool
g	Tool Tip Point
H	Height
\mathbf{h}	Translation Vector
J	Jerk
L	Length
ℓ	Line

Symbols	Definition
M	Matrix
N	Normal Vector
O	Obstacle
\emptyset	Null
σ	The Nominal Tensile Strength
P	Workpiece
p	Point in local Coordinates
Q	Objects
q	Quaternions
R	Rotation Matrix
\mathcal{R}	Rate
Γ	Not Accessible Zone
r	Radius
S	Distance on the Side of Rivets
s	The Generated Path
T	Time
\mathbf{T}	Homogeneous Transformation Matrix
\mathcal{T}	Normalized Time
t	Thickness
τ	Shear Strength of Rivet
V	Volume
\vec{V}	Vector
v	Velocity
W	Width
ψ	Distanced Between Rows of Rivets
Y	Safety Zone
I/O	Input/output

Sub/Superscripts	Definition
<i>c</i>	Center
<i>g</i>	Gun
<i>h</i>	Homogenous
<i>home</i>	Home Position
<i>i</i>	i^{th} element
<i>j</i>	j^{th} element
<i>k</i>	k^{th} element
<i>max</i>	Maximum
<i>min</i>	Minimum
<i>R</i>	Robot System
<i>r</i>	Robot
<i>T</i>	Tool Module
<i>t</i>	Tool
<i>tip</i>	Variables at the tip of module
<i>w</i>	Work Peace
<i>z</i>	Zig-zag Riveted Lap

CHAPTER 1: INTRODUCTION

This thesis presents a novel method for robotic drilling and riveting based on tool accessibility analysis with path and motion planning for aircraft assembly applications. This chapter provides a general overview of the topic presented in the thesis.

In the background section, a brief review of robotic applications background in aircraft assembly and automation is presented. It provides an overview of multiple systems and assembly techniques used for aerospace applications. Further, it provides a description of existing robotic manipulators, tools, and fasteners while highlighting the advantages and disadvantages of each system. Lastly, a complete description of the system that was used for the development of this thesis is provided.

In the problem formulation section, the challenges faced by robotic assembly and automation are presented. This section also highlights all the assumptions and considerations used as the foundation of the thesis.

In the outline section, the full thesis outline is presented with a summary explanation of the content of each chapter.

1.1. Background

A machine is an assemblage of parts that transmit forces, motion, and energy one to another in a predetermined manner. In general, robots are machines that are made of a number of links connected together with joints. They can be operated mechanically using gears and engines, or electrically using electronically controlled motors and actuators to perform any given task. These

machines are used to replace humans in performing various applications that can be repetitive or have a risk potential [1].

Today, many industries like automotive and aerospace adopt high manufacturing standards in accuracy, high quality, and efficiency for mass production when it comes to assembly processes. This makes machines and robots excellent candidates as the choice for assembly automation. Industrial robots in particular are controlled by computer systems that can be programmed to perform complicated tasks with high accuracy. This high accuracy enables them to perform tasks that result in high quality finished products, which makes industrial robots more cost and scrap efficient for mass production.

1.1.1. Assembly Types in Aerospace Manufacturing

Aerospace manufacturing is the task of assembling a number of parts, components, and systems to make an aircraft. Assembly is known as joining two or more components to create a new entity using tasks like welding or mechanical fasteners [2]. The aircraft assembly go through three stages that starts with subcomponent assembly, then component assembly, and ends with line assembly. The subcomponent is the early stage in the assembly process, where parts are joint together to form the base for the component structure. The main components that form the structure of the aircraft are the cockpit, fuselage, wing, empennage (Horizontal and vertical stabilizer), engines, and landing gears. These main components are then joined together to make a complete aircraft through the line assembly [3].

A number of joining technologies are used in the aerospace industry that include welding and mechanical fasteners, where each technology is used for a specific application. Welding is used in joining structures that require permeant joining, as it is hard to remove the welding without impacting the structure integrity of the joint material. The welding process also involves fusion

method, which causes thermal deformation to the molecular structure of aluminum alloys [3]. Further, welding restricts the degree of freedom of the welded structure, and it increases the residual stress at the joined section especially if the weld involves two different materials. On the other hand, mechanical fasteners are used for the parts that require easy access and replacement. Mechanical fasteners includes screws, bolts, nuts, studs, and rivets. Screws, bolts, nuts, and studs are ideal for repetitive use and provide structural support to load bearing structures. However, rivets are considered permanent mechanical fasteners that can be removed using drilling process.

Rivets are widely used to join two or more panels together in aircraft assembly. In the aerospace industry, the riveting of components and sub-components plays a major role in aircraft assembly. They are used to join fuselage skin to the fuselage structure, wing skin to the wing structure and empennage skin to the empennage structure.

There are three types of riveting systems used in aerospace manufacturing sectors: *manual riveting*, *semi-automated riveting*, and *automated riveting* (Figure 1-1) [4].



Figure 1-1: a) Manual Riveting [49], b) Semi-automated Riveting [53], and c) Automated Riveting Systems [35]

The manual riveting system (Figure 1-1.a) is widely used in the aerospace manufacturing field, as it is ideal for small sub-component assembly due to the large variety and geometric complexity of the parts. However, manual riveting methods tend to be quite labour intensive,

tedious, error prone and pose health hazards like joints pain, ear ringing, and long-term hearing loss to the pneumatic gun operators [5].

On the other hand, semi-automated and fully automated riveting systems require fewer trained workers to operate the system and load parts to the machine, thereby reducing any health hazards. These riveting systems are only limited to component assembly, such as fuselage skin panels and wing skin panels [3]. However, semi-automated riveting machines with automatic feeder like Superior's™ P-18 Riveter (Figure 1-1.b) are rather limited for use in assembling larger components with simple shapes because of their lack of adaptability and physical constraints, where for each specific riveting task a specific machine is used [6].

Unlike the semi-automated riveting systems, the automated riveting systems like MTorres™ (Figure 1-1.c), have the ability to accomplish more than one riveting task with minimal human interference, as these systems can be added as end-effector to any semi-automated or automated industrial manipulator. Moreover, these automated riveting systems are relatively small when compared to the semi-automated riveting systems, and can be adjusted easily to accomplish several tasks at once due to the flexibility of the robotic manipulators. Thus, fewer riveting machines are needed to accomplish riveting tasks in the automated riveting system than with the semi-automated, thus increasing the facility space to be used for other manufacturing tasks. This latter system also reduces the required time to accomplish each task, which leads to an increase in production rate as well as a reduction in the capital cost and the number of operators.

Drilling a through hole is performed before the application of mechanical fasteners. Similar to rivets, there are three types of drilling systems used in aerospace manufacturing sectors: *manual drilling*, *semi-automated drilling*, and *automated drilling* (Figure 1-2).



Figure 1-2: a) Manual Drilling [49], b) Semi-automated Drilling [57], and c) Automated Drilling Systems [36]

The manual drilling system (Figure 1-2.a) is similar to the manual riveting system. Both are labour intensive, tedious, error prone and pose health hazards [5]. Similar to the semi-automated riveting system, the semi-automated drilling system like Harsle™ Z3063 (Figure 1-2.b) is limited for use in drilling larger components with simple shapes because of their lack of adaptability and physical constraints. The automated drilling systems like MTorres™ Flexible Drilling Head (Figure 1-2.c) have the ability to accomplish more than one drilling task.

1.1.2. Robotic Automation in Aerospace Manufacturing

Robots exist in various categories, including *manually controlled*, *semi-automated*, and *fully automated*. Each category is used in the manufacturing and assembly process to perform different tasks based on the manufacturing process and requirements. These systems vary in size and control system complexity based on the expected efficiency and mass production requirements.

Manually controlled robots use a set of controllers like I/O switches and cameras in some cases to monitor progress and are controlled by an operator to perform various tasks. This type of robots depend on human factors to operate while executing each task or performing a given operation. These manually controlled robots are used in assembly tasks that involve heavy objects and require a human supervision to avoid damaging the parts, where tasks are irregular and cannot be automated.

Semi-automated industrial robotic manipulators are used specifically to perform repetitive complex mechanical tasks that require a high level of accuracy, task repeatability, and low margin of error. These manipulators are widely used for automation in various sectors like the automotive and aerospace industry. They are mainly used to perform tasks like heavy lifting, moving, assembling, welding, and riveting. Similar to the human arm, a manipulator is made of a series of links that are connected by joints, which are controlled by an advanced control system [1].

Semi-automated robots use a pre-programmed set of commands that can be changed, manipulated and executed based on the required task. Generally, these robots are equipped with sensors and gauges that can control a set of pre-programmed tasks in given situations. For stability purposes, these manipulators are required to have a fixed base to increase the accuracy and reduce any possible deflection. However, being stationary limits the ability of the robot to move freely in the 3D space and limits the working range into a small space [7]. Such limitations in general cause challenges to perform tasks like riveting in confined spaces or in environments that have a large number of obstacles. These semi-automated robots include industrial robots like ABB™, and KUKA™ robots.

The fully automated robots are totally preprogrammed and uses a set of sensors, cameras, and gauges to manipulate, change and execute a set of codes based on a large number of variables and conditions without human intervention. These smart robots are the most sophisticated type of robots that are still in development and will be the future of smart manufacturing.

1.1.3. System Description

Ryerson University is developing a new automated percussive drilling and riveting system in an effort to make automated drilling and riveting systems as efficient and compact as possible while reducing manufacturing time and capital cost. This percussive drilling and riveting system

functions as an end-effector attachment for any small size semi-automated light to medium-payload industrial robotic manipulators with six degree of freedom like the ABB™ IRB 4400. The advantage of developing this new compact system is that it can access tight and awkward areas easier than most of the existing systems now available in market which are limited to component assembly [3].

This system uses various components, including a Chicago™ pneumatic drill, Chicago™ pneumatic rivet gun, a rivet feeder unit, a tool exchanger unit ATI™ QC-41-Master and Tool units, attachment jig, and a Delta-TAU™ CNC frame machine that is used to carry a gantry that hold a bucking bar [8]. In addition, this system is equipped with Laser 3D point coordinate measurement system Boulder Innovation Group (BIG)™ 3D Creator FP 7000 that is used for calibration and depth calculation, and a camera system attached to a camera mount [9]. The system overview is presented in Figure 1-3, Figure 1-4, and Figure 1-5.

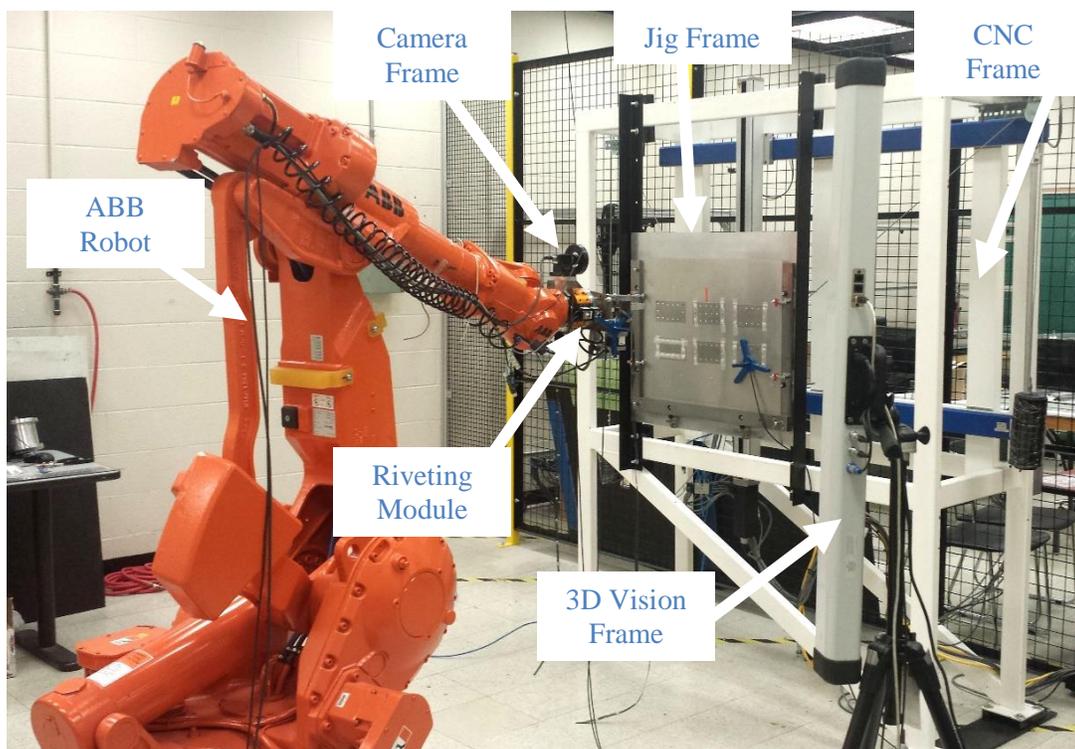


Figure 1-3: Overview of the Robotic System

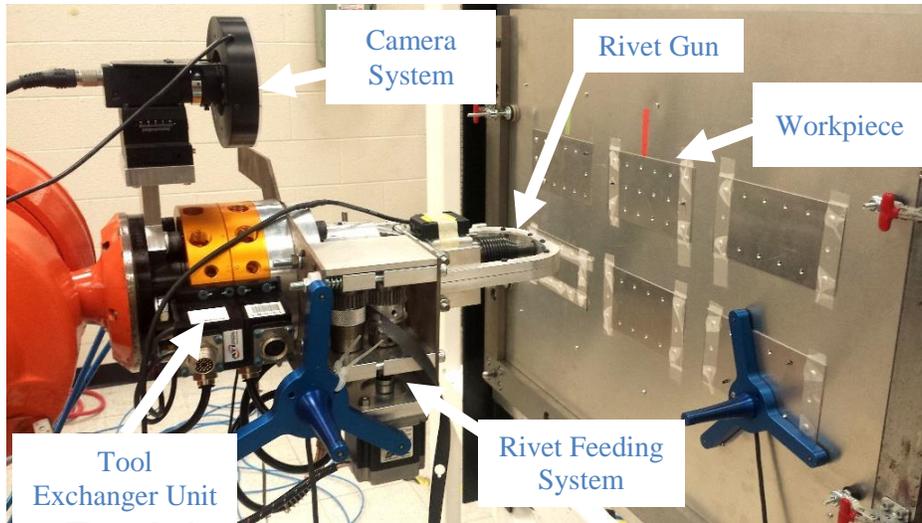


Figure 1-4: Overview of the Robot Tool System

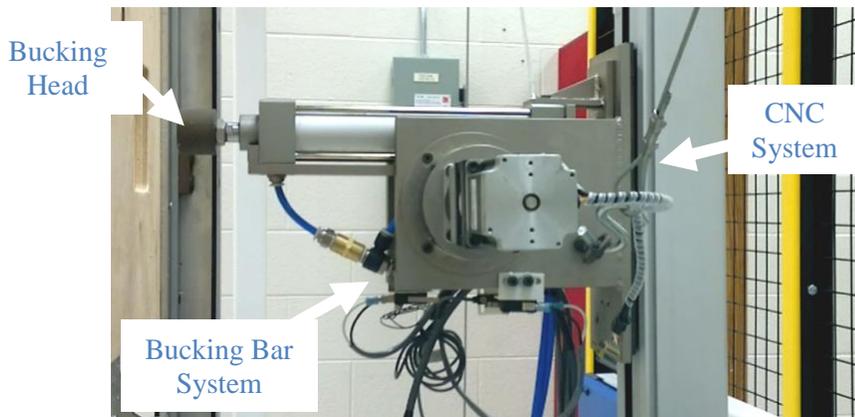


Figure 1-5: Overview of the CNC Bucking Bar System

This new automated drilling and riveting system uses a number of controllers that communicate together to allow appropriate drilling and riveting task execution. Major controllers are the *Robotic Percussive Cell Controller*, the *CNC Bucking Bar Gantry Controller*, The tooling System Controller, the 2D Camera controller, and the 3D Vision Scanner Controller [3]. All these controllers are connected to a central workstation computer system that provides each controller with the required task to execute through the drilling and riveting process. The main software package that is used on the workstation is known as RobotStudio™ that is developed by ABB™

Robotics (Figure 1-6) to provide users with a better ability to graphically control the robotic system and additional add-ons (known as Add-ins in RobotStudio™) like the controllers mentioned above.

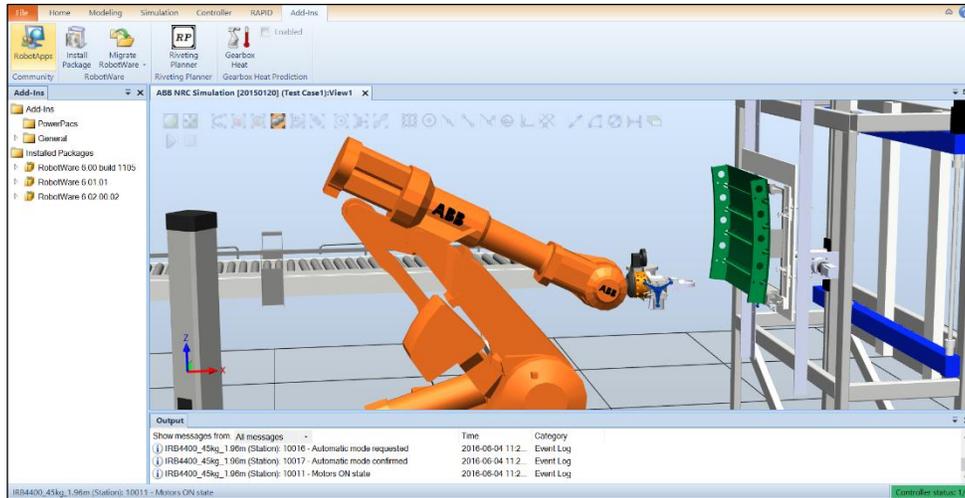


Figure 1-6: Overview of the RobotStudio™ Interface

In order to achieve a fully automated system, a user interface (UI) software application that enables the user to accomplish multiple drilling and riveting tasks by simple data input, was developed as part of the scope of this thesis. This software application has the ability to function as an add-on in RobotStudio™. This add-on provides the user with the path planning required for drilling and riveting tasks. In addition, it generates a virtual simulation that simulates the tasks process before proceeding with the actual task execution with the calculated path to the robot. This enables the user to validate the calculated path and to easily visualize the process before compiling it. This project and software are patent under the *Automated Percussive Riveting System* paten number WO 2013152440 A1 [10].

1.2. Problem Formulation and Objective

Most of the semi-automated and automated robotic drilling and riveting systems available in the market today, like the MTorres™, KUKA™ drilling and riveting systems, have large size tool head module that is limited to some simple assembly applications. The large size of the tool head module attached as an end-effector presents an accessibility constraint limiting the usage of the

system to perform tasks like attaching skin panels to wing structure and fuselage sections. Therefore, these large tool head systems cannot perform tasks in confined spaces like cockpit and empennage segments. For this reason, large systems still depends on workers to perform all the sub-components assembly tasks, which tend to be quite labour intensive and pose health hazards.

Ryerson University percussive automated system is considered an ideal drilling and riveting system that has a small tool head size and can perform tasks like drilling and riveting for small and complex geometry parts. This is considered an advantage in design and concept when compared to KUKA™ and MTorres™ drilling and riveting systems. This percussive automated system is suited to replace manual applications in sub-component assembly. The ultimate goal of this new system is to be as compact, efficient, and accurate as possible in order to operate in confined spaces and crowded environment like aircraft wings and empennage structures with a complex geometry. However, this system requires some study analysis regarding the accessibility of the tool to tight and confined spaces.

The goal of this thesis is to improve the robotic accessibility, by proposing new approaches, methodologies and techniques that support the efforts of the path planning process. This involves the ability to perform the drilling and riveting task in the RobotStudio™ simulated environment, while taking in consideration all the obstacles available in the operational volume of the robotic tool head.

Four main objectives were set for this thesis to achieve tool accessibility with path and motion planning for the robotic drilling and riveting system. These four objectives are highlighted as follows.

The first objective of this thesis is to develop an accessibility analysis process that improves the robotic drilling and riveting accessibility for small parts and panels. This process is based on

geometrical accessibility of the tool and part geometry. In this process, a standard approach is defined to classify tool and part geometry based on shape and radius of curvature. This classification would reduce the number of curvatures and simplify the solution for the least amount of curvatures. Then, the Global Accessibility Area (GAA) and Global Accessibility Volume (GAV) new approaches are used to determine the accessibility boundaries for 2D and 3D applications. The Global Accessibility Area (GAA) would determine all accessible boundaries using the geometrical projection approach of the obstacles. While the Global Accessibility Volume (GAV) would determine all accessible boundaries using the geometrical volume extraction approach.

The second objective of this thesis is to develop a methodology that generate and select the optimal path for robotic drilling and riveting process based on the standard equations for rivet spacing and size. This methodology is developed based on the GAA and GAV to determine all possible paths for the required tasks. The optimal path is selected based on these three factors: the shortest distance, the least number of steps, and the least number of tool orientation change. These factors are important to ensure that the optimal path will guarantee the fastest time, which will increase the production efficiency.

The third objective of this thesis is to analyze the motion of the robot through the execution of the optimized path based on the robot's maximum velocity and acceleration. This analysis would provide a calculated estimate of the number of rivets that can be completed per minute taking in consideration the robotic velocity and acceleration constraints.

The fourth and final objective of this thesis is to develop a software application for RobotStudio™ that can be used as an add-on to simulate the drilling and riveting process through the generated path based on techniques and methodologies proposed in this thesis. This software

application would have a user-friendly interface and would provide the user with a calculated path for the location of each rivet. Further, this drilling and riveting path generator will have to generate all the needed files required for other systems as well as the 3D virtual simulation of the generated tasks.

1.3. Outline of the Thesis

The thesis is organized and divided into six main chapters in addition to the introduction. The thesis is outlined as follows:

Chapter 2: presents a literature review that covers all the topics discussed in this thesis as well as a review of previous research and key technologies that exists today. This chapter starts with definition of tool accessibility. Then, a review of tool accessibility techniques used and developed by researchers. Next, a review of the existing robotic path planning and collision-free path approaches used in robotics manufacturing. Then, another section is dedicated to motion curves and motion planning. Finally, a review of the applications of robotic drilling and riveting automation in the aerospace industry.

Chapter 3: presents a detailed accessibility analysis. The first section introduces the definition of tool accessibility and the criteria for accessibility analysis. Then, a review of the technique used in the thesis to classify the tool and part geometry. Next, the geometrical accessibility analysis with the geometrical projection and volume extraction concepts are introduced. These concepts are introduced as the Global Access Area (GAA) and Global Access Volume (GAV) approaches. Finally, the percentage accessibility is presented.

Chapter 4: presents a detailed path planning analysis for robotic system where a new pathfinder logic approach is defined for collision-free path. The first section reviews the rivet spacing

requirements. This section is followed by the path selection process section that uses the shortest path, the approachability profile, and the tool orientation criteria as the foundation to select the optimal path. The last section presents the new pathfinder methodology that is developed based on the GAA and GAV to determine optimal collision-free paths for the required drilling and riveting tasks to increase production efficiency.

Chapter 5: presents a detailed motion analysis based on displacement, velocity and acceleration of the robot during the execution of the planned path as well as cycle time. The first section presents the motion curves for the robotic path displacement, velocity, and acceleration analysis. This section investigate the motion curves to quantify the velocity and acceleration required by the robot at each step of the optimal path. The optimal path selected through the pathfinder methodology is then validated through the review of these curves and their maxima when related to the manufacturer boundaries. Last section will determine the cycle time that the robot take to archive the required tasks successfully and determine the productivity rate.

Chapter 6: presents the simulation application generated based on the methodology established in this thesis for tool accessibility. The first section presents an overview of the algorithms used to develop the software application tools to support the scope of the thesis. Then, the second section reviews the application of the developed algorithm to test some 2D and 3D cases using a MatLab based scripts as an application tool. The following section includes an overview of the algorithm implementations for aircraft assembly using the developed add-in software application. The last section involves some test cases and their simulation validation using some aircraft parts as a sample of aircraft assembly applications.

Chapter 7: presents the conclusion and the contribution of the thesis, and proposes future work to be conducted.

CHAPTER 2: LITERATURE REVIEW

This chapter presents a general coverage of the topics discussed in this thesis as well as a review of previous research that have been done on these topics. The literature review covers the following topics: tool accessibility, robotic path planning, robotic motion planning, and application of riveting in aircraft design.

The first section will cover a brief description about tool accessibility and accessible volume. It will start by tool accessibility definition followed by a review of tools designs used for aerospace drilling and riveting applications to address limited accessibility and obstructions. Then, a review of a number of a number of previous research conducted in tool accessibility and the techniques used to determine the accessible volume for the tool.

The second section will cover a brief review of the robotic path planning techniques and the different approaches used for determining the best path for the robot. Further, this section will review the criteria for collision-free path and minimal distance optimization.

The third section will cover the robotic motion planning techniques and define the different types of motion curves and cycle times. More, the section will cover the known approaches to plan the motion of a robot.

The last section will review the applications of riveting automation in aircraft design to relate the knowledge of path planning with the constraints of aircraft design and environment. This section will include a review of the applications of robotic drilling and riveting in the aerospace industry and the systems used in the industry.

2.1. Tool Accessibility

Tooling and tool accessibility are two important factors in planning a complete assembly or disassembly sequence from tool selection and design perspective. Embedding a tool selection in assembly planning as a process is a challenging task as it depends on the tool feasibility, which is complex, dynamic, and variable [11]. An incomplete plan generated for product assembly or disassembly may lead to retooling, special tooling, long lead time, and even design changing at the production stage [12].

Tool accessibility is defined as the ability of a given tool to reach the required position on the workpiece and can be operated while experiencing the minimal location and path interference of any obstacles such as neighboring parts, tools and fixtures that could prevent the tool from completing the required task as desired. The tool accessibility depends on three factors: the tool shape, the complexity of the workpiece design, and the obstacles in the workpiece surrounding.

2.1.1. Tools Selection and Customization for Aerospace Applications

Tools are equipment designed to facilitate the process of performing specific tasks. The size and shape of the tool can present a challenge in the accessibility of the tool to perform a task like drilling or riveting. In previous research, Peng [13] identified a number of key factors that dictates the selection of the correct tool to perform any given task. These factors includes the type, size, and visibility of the fastener location, the effective tool placement, the volume and shape of the tool, the weight size, material and shape of the workpiece, and the minimum space that should be free in an assembly to apply the tool. Therefore, for an optimal tool accessibility planning, it is preferred to reason the available space for a tool before planning its application during the assembly process [13].

Most of the aerospace structures are complex in nature, where many design constraints and access limitations prevent tools from easily accessing the required locations to perform given tasks. This complexity makes the space availability of the tool to reach the desired location more limited. For instance, the spaces between stringers and stiffeners in the assembly process of fuselage panel are considered major obstacles that dictate the tool insertion with a specific way; otherwise, it can damage the stringer or skin of the fuselage [14]. Another challenge for tool accessibility is that sometimes the fasteners are not visible to the observer or the camera [11]. Therefore, the shape and size of the workpiece has a major impact as well in dictating the tool shape and size based on the number of the available obstacles available in the workpiece design.

The decision of selecting the proper tool is often determined by the workpiece designer or the assembly planners' experience and testing. Through the design process, the designer or assembly planner conducts accessibility analysis to ensure that the recommended tool for the task is appropriate for the workpiece design. If the tool is not able to reach the required location, the designer modify the workpiece design or design a unique tool for the required task.

In Aerospace applications, unique tools are designed for complex workpiece to overcome challenges related to tool accessibility. For instance, DOBCO™ Equipment Ltd. designs specific tools for drilling and riveting fuselage and wing panel in aerospace assembly as in Figure 2-1.

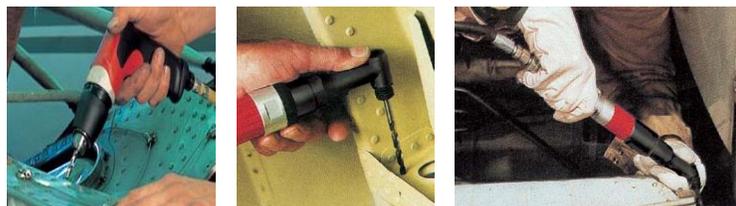


Figure 2-1: A Sample of Drills Types Used in the Aerospace Industry [15]

Multiple drill designs are used to perform drilling operations in confined spaces that cannot be easily accessed using conventional drills. Some of these modifications can be in a form of

removable extensions, as shown in Figure 2-2, where the extension can be replaced based on the task to perform [15].



Figure 2-2: Pneumatic Angled Drills [15]

Another example from the aerospace industry is the customized bucking bar shapes that are used when performing a riveting task to attach the aircraft skin to stringers. These bucking bars are selected based on the stringers shape as shown in Figure 2-3. This enables the bucking bar to perform its intended task while bucking the tail of the rivet to hold it in place [16].

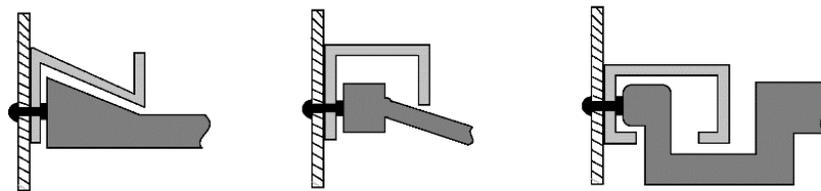


Figure 2-3: Modified Bucking Bars to Fit in Different Flanges Forms [16]

Thus, tool design customisation plays a major role into overcoming the majority of limitations that can prevent the standard tools from performing their task.

2.1.2. Tool Accessibility Approaches

There are a number of tool accessibility approaches that have been developed since the 1980s. These approaches include: the Configuration Space (C-space) approach [17], the Visibility Map (VMap) approach [18], and the Accessibility Cone approach which includes the Local Accessibility Cone (LAC) and Global Accessibility Cone (GAC) [19].

2.1.2.1. Configuration Space (C-space) Approach

The first approach requires the computation of the Configuration Space (C-space) representation of obstacles, which could be considered very complex when dealing with variations of used volumes and constraints. This method combines collision detection methods to solve the Find-Space problem and randomised target point path planners to solve the Find-Path problem [17]. As presented in Figure 2-4, finding the space problem is to determine if the object G when placed in region ζ , which is the Configuration Space, can collide with any of the objects O_j available in the region ζ . Finding the path problem is to determine if the object G when moved from one location to another will collide with any of the other objects O_j [17].

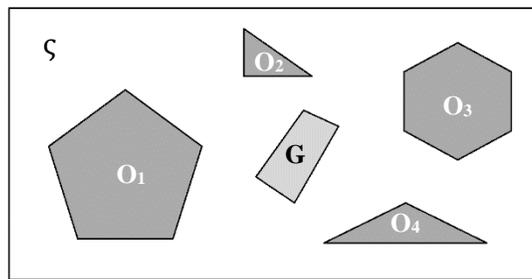


Figure 2-4: C-Space Find-Path and Find-Space Problems in 2D [17]

This method assumes that the tool place and use volumes are not variable during a tool application. Which means if the tool changes in shape or size through its application, the C-space computation becomes no longer valid. Further, once the C-space is created for a given tool size or type, it cannot be reused for examining another tool for different size or type. More, this approach consider that all obstacles are in static condition. This approach however, is considered to be computationally expensive to implement, at it requires a large number of iterations to validate if the tool is moving freely. The computational complexity of this approach especially for 3D planning when combined with collision detection becomes more challenging [13].

2.1.2.2. Visibility Map (VMap) Approach

The second approach is the Visibility Map (VMap) approach, which is based on Gaussian Map (GMap) [18]. A Gaussian map can be represented in the form of a spherical shape where the surface of this sphere is composed of the set of end points of the unit vectors normal to the surface. Woo et al. [20] expanded the idea behind Gaussian maps and developed the visibility map approach, which is defined as a set of points on a spherically convex region. They pointed that the local accessibility of a point on a surface is defined by the hemispherical region constructed by using the surface normal at the point as the pole [21]. However, Savinder et al. [21] mentioned that the visibility map cannot be used to determine the global accessibility of an individual facet within a concave region. This is because the VMap of an object is constructed using the local accessibility information for various facets.

2.1.2.3. Accessibility Cone Approach

The third approach is the Accessibility Cone approach, which is based on Gaussian image-Minkowski sum [19]. The Accessibility Cone approach can be divided into two categories: the *local accessibility cone* (LAC) and *global accessibility cone* (GAC). Spyridi et al. [19] defined the local and global accessibility in their paper. The local accessibility of a point on a surface is a hemispherical region, where the pole of which is the normal at that point. The global accessibility is defined as all possible accessible directions to reach the workpiece surface without having any interference with the workpiece structure [22].

Peng [13] adopted and improved the global accessibility cone (GAC) approach through his tool space reasoning and accessibility planning studies. He pointed out that the GAC approach reduces the complexity of existing planning systems and makes the tool assembly reasoning

computationally inexpensive. This approach proved success over the other one as it takes the fastener type and size as the only non-variable factor of the analysis.

The GAC is based on the geometrical mapping of a discrete unit sphere that represents a set of directions, where the sphere center is the center of the fastener head at the flat section to the workpiece surface. The discrete unit sphere is represented by 180×360 pixels, which create total 64,800 directions on the sphere with 180 colatitude angles (θ) and 360 longitude angles (φ) in a spherical coordinate as shown in Figure 2-5.

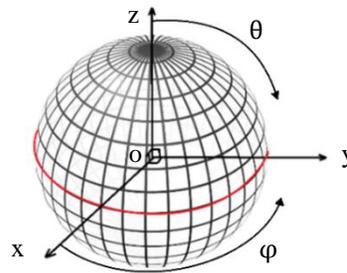


Figure 2-5: GAC Construction at the Fastener Location [13]

Therefore, there is a one-to-one mapping between directions in the GAC and unit vectors in 3D space, which are defined by the angles θ s and φ s, as shown in Figure 2-6. Thus, φ and θ in this research are used to calculate a unit vector and to represent the direction at an equivalent pixel (φ, θ).

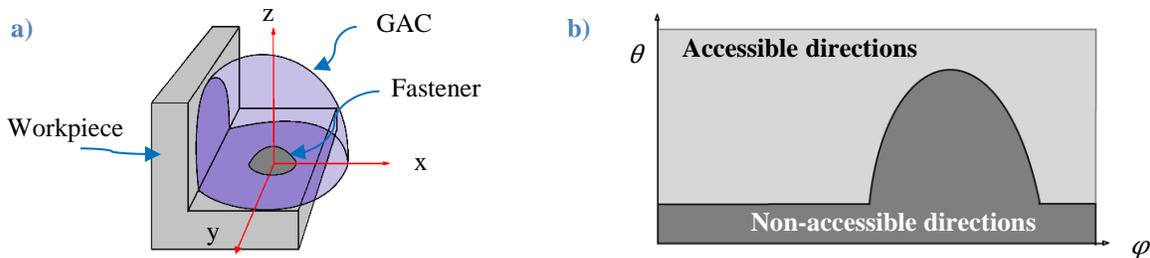


Figure 2-6: GAC Application, a) The 3D Representation, b) The Map Representation [13]

As an improvement to the GAC approach, Peng introduced a new methodology called Triangle Patch to reduce the complexity of depth analysis, as it is time-consuming process in a

complex product consisting of many parts. This methodology was proven to be fast, efficient and practical for assembly tool reasoning based on his research [13]. Peng used virtual environment and 3D simulations to verify whether the assembly approaches are appropriate or not, due to the tool space reasoning computational complexity [23]. Further, he adopted a via-Web and Java DB system that depends on ‘HandTool’ standard tool catalogue tables made by Snap-on Company for tools geometric information. This geometry information of each basic tool is used to define the tool parameters for a complete tool that is stored in the ‘HandTool’ table, which helps the user to examine the tool interference with obstacles during the simulation of a tool application. Despite being successful, this method is mainly valid when considering the accessibility from the fastener perspective as it consider the tool shape and design to be the only variable to validate. In addition, it depends on a predefined library set of tool to select the best tool to reach the fastener location despite the part geometry or the path required to reach the fastener head.

2.2. Robotic Path Planning

Robotic assembly is the most sophisticated task that an industrial robot can perform. This task combines path control of continuous path with the precision of machining tool, where the robot have to work fast, while performing smaller, smoother, and more intricate motions [24]. Despite the fact that each joint has fewer degrees of freedom, the industrial robot manipulator can move through greater angles in a spherical working range [25]. They are able to move in at least three directions: vertical, horizontal, and around the vertical axis. Another three axes of motion yaw, pitch and roll are added once an end-effector is added to the arm [24].

Due to their maximum flexibility, industrial manipulators have many advantages including the ability to cover a large workspace relative to volume of robots, and can reach over and under workpiece. The robotic workspace is defined as the envelope that contains the set of all points,

which the robot can reach, to perform a given task without losing its accuracy or reaching singularity.

On the other side, manipulators are known to have complex kinematics, which makes it difficult to control their linear motion. Further, having complex kinematics makes it difficult to visualize easily the motion of the manipulator without the support of a computer simulator. From a structural perspective, the rigidity of the manipulators tends to decrease at full reach affecting accuracy and creating a given margin of error specified by the manufacturer [25].

2.2.1. Obstacle Avoidance

There are a number of terminologies that are used when solving the path planning problem. These terminologies are the configuration space ζ (C-space), the space of free configuration ξ (C-free), and the obstacles representation in the C-space (C-obs). The C-space ζ is the space of all possible configuration relative to a reference point (Figure 2-7). The C-obs is the space occupied by the obstacle, while C-free is the other space that is not occupied by the obstacle in C-space as shown in Figure 2-7 [17].

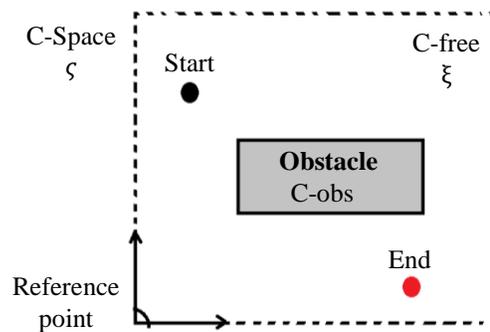


Figure 2-7: Path Planning Problem Terminologies

Gasparetto et al. [26] defined that there are number of methodologies used for path planning to generate the required geometric path, which includes the roadmap techniques, the cell decomposition algorithms, and the artificial potential methods.

2.2.1.1. Visibility Graph Method

The roadmap or visibility graph technique is based on the geometrical projection of the obstacles that are available in C-space. In this approach the free space connectivity is mapped into a one-dimensional curves in the C-free space. This generates a number of curves or paths that links the initial configuration to the final configuration. Then, the generated graph can be searched for the optimal shortest path also known as the Euclidean path. This approach generate a path that passes as close as possible from the obstacle. This approach is very successful for 2D path planning as it consider the projection of all obstacles in the C-space of the robot. Further, the path optimization is not challenging and it considers the Euclidean path approach.

Another derivative of the visibility graph method is called Voronoi diagrams, where the spaces between obstacles are divided to the equidistant point. Each segment has a defined boundary that is far enough from the obstacles, while allowing an optimized path between the obstacles. The Voronoi diagram is ideal for path planning when considering a large clearance distance between the tool and the obstacles, which is not like the visibility graph approach which create the path in a very small clearance distance.

Even though the roadmap method is in general ideal and simple for all 2D path planning application, the 3D path planning is very complex. For 3D applications, this approach follows a random search process to generate a wide number of proposed paths. All the proposed solutions to the path planning can be presented in a tree form that starts from the initial point and ends at the final point. Then, this tree of proposed paths has to be optimized trough a complex computational process. This optimization process becomes even more complex as the number of the obstacles in C-space increase, making it a very slow computational process [26].

where the robot always has to lean toward. On the other side, the obstacles are surrounded by repulsive fields, where the highest repulsive field is the closest one to the obstacle. The generated path tends to go from the initial position to the final position, while avoiding the repulsive fields. Even though this method provides a simple path planning approach for practical purpose, it has a major problem related to local minima. This problem happens when the robot is trapped at a local minimum before reaching the final point [27].

2.2.2. Optimal Path Planning

Path planning is a critical issue in the field of robotic automation, as explained by Gasparetto et al. [26]. The ultimate goal of robotic automation is to perform the required task in the fastest possible speed for the shortest production time. This means that the distance traveled by the manipulator shall be optimal to ensure the task is properly executed as planned. However, the faster the manipulator goes the more the control systems and actuators are at risk of losing accuracy. For this reason, it is important to consider a particular care when generating the path, especially if it will be executed at high speed, in order to avoid any harm to the control systems of the robot.

The path planning algorithm generate a geometric path in 3D space, for a start and an end point, through which the robot will path through a number of points to execute a given task like drilling or riveting within the workspace of the robot. The path planning is a geometric problem that define a geometric path with no time factor. There are two types of path planning than can be generated standard path and dynamic path. The standard path deals with the required task and the steps required to accomplish this task, this includes tasks like riveting pattern, drilling sequence and riveting approach. While the dynamic path deals with obstacle avoidance like avoiding a spar or a flange.

The robotic path planning is defined as the ability to define a collision-free path between a start point and a final point within the robot workspace envelope. If all the obstacles in the workspace are known then the generated path will be considered static. If the obstacles in the workspace are unknown, then the generated path will be dynamic as the path will keep changing based on the obstacles encountered through the process. The static path is much simpler and require less computation and processing time when compare to the dynamic approach [26].

2.2.3. Techniques of Programming Industrial Robots

There are a number of approaches used in the industry to program industrial robots, validate the generated path, and verify that the robot will perform the task as required. These include the human control robotic teaching approach in semi-automated systems, the real-time scanning and guidance approach, and the virtual environment simulation approach.

2.2.3.1. Robot Teaching Approach

The human control robotic teaching approach is used widely when programing industrial robots today. In this approach, the robotic manipulator is first taught the required target position by adjusting manually the robot's coordinates. Through this process, the manipulator coordinates are changed until the robot reaches the required target point, which is then stored in the robotic logic. This task is then repeated for every point that the robot has to be at. Once all the points are entered, the robot compile the best path and propose the ultimate speed to accomplish the required task. This approach is mainly used for the simple repetitive tasks that does not contain a variety of obstacles, like spot welding and line assembly. This approach is considered the simplest method to program the robot; however, this approach depends mainly on human expertise and is prompt to error.

2.2.3.2. Real-Time Scanning and Guidance Approach

Another approach is the real-time scanning and guidance. In this approach, the robotic manipulator is guided using a sensor input like 3D cameras or laser scanning system that does a real-time analysis of the surrounding environment. In case of unexpected situations in workspace, the robot manage the situation by dynamically updating the proposed initial path in a fast way. Jang and Lee [28] were able to develop a 3D camera system that was able to analyse the surrounding environment, then analyze its contents, determining all its obstacles. Next, the system then plans the proper path that the manipulator has to take to reach the workpiece. This approach is relatively complex, as it depends on a large number of sensors and image analyzation as well as continues human monitoring through the programing phase.

2.2.3.3. Virtual Environment Simulation Approach

The last approach is the virtual environment simulation. This approach depends on offline programing the robot to reach specific target points and to follow a specific path that can be generated and modified in virtual environment. The generated path is then calibrated relative to a reference point then the generated scripts are sent to the manipulator to be executed. Systems like the ABB industrial manipulators are equipped with simulation software like RobotStudio™ that can simulate the full manufacturing process in a virtual environment. The manipulator, the tool, and the workpiece can be imported to this software, and the path of the task can be simulated in the virtual environment. Such approach enables the full visualization of the process with the ability to detect any obstacle or accessibility issues. This approach also allow operators trough the collision avoidance logic to verify if the required task will be possible to achieve using the current tools or tools need to be changed or redesigned [29]. This approach is the best approach to verify

and validate the generated path, as it is feasible and does not require more than basic knowledge of computer programming skills.

2.2.4. Additional Consideration for Path Planning Optimization

When considering path planning, the manipulator constraints and range boundaries, the end-effector shape and size, as well as workpiece shape and size, should be taken into consideration. For the manipulator, the manipulator constraints are considered to avoid singularity. Also the shape of each link and its accuracy in movement and deflection tolerance is considered to avoid harm to workpiece [30]. For the end-effector, the shape and size are considered to ensure that the end-effector will be able to fit in the accessible volume and reach the required target position. For the workpiece, the shape and size are considered to ensure that the end-effector and the manipulator are not damaging or in collision with the workpiece.

When dealing with path planning for riveting process, a number of considerations are taken into account. One of which is the acceptable standard riveting patterns defined by the Federal Aviation Administration (FAA). These regulations defines a specific edge distance, minimum and maximum pitch distance or distance between rivets, number of rows of rivets, and the installation sequence and procedures [31].

Having the standard riveting pattern in mind is an important aspect while generating the path through the path planning process. This ensures that the generated path will comply with the FAA standard regulations. More, considering the type of obstacles faced the riveting process is as important as the generated pattern. These obstacles include the shape of the beams used for stringers and stiffeners, shape of spars and joining sections, as well as the geometry of the workpiece. Further, the optimal goal of the generated riveting path is to be for the shortest distance

to ensure fast production rate. Therefore, it is important to consider all these elements through the path planning process, as they add to the complexity of the path planning problem.

2.3. Robotic Motion Planning

Robotic motion planning is an important aspect to consider when planning for robotic assembly tasks like riveting or when conducting tool accessibility analysis. This studies the dynamic aspect of the planned path, and validate if the robot is capable of delivering the required task within a minimal execution time as required, without exceeding the actuators design limits or harming the robotic manipulator. Therefore, understanding the robotic motion planning algorithms is an important aspect to consider when performing tool accessibility analysis, and the riveting pattern through the path planning process.

Motion planning algorithm, also known as trajectory planning, is the process of combining the time information with the generated geometrical path. This algorithm combines the planned path with the kinematic and dynamic constants of the robot to provide the required motion profiles for the manipulator position, velocity, acceleration, and jerk. Through this process, the target point of the end-effector is extracted, the corresponding values at each robot joint is determined using kinematic inversion, and last the motion plan is generated using interpolation function in the joint space. It is important that the generated profiles must be smooth and within the robotic design constraints. This to prevent situations like excessive forces on joints or actuators, over-torque, or mechanical resonance. Further, it is important to keep in mind that abrupt motions at high velocity can damage the actuators at the joints. Therefore, the ultimate goal of the motion planning is to ensure a continuous acceleration of the joints with a limited jerk as it is crucial to reduce induced vibrations to prevent the wear of the mechanical structures [26].

2.3.1. Motion Curves

Motion of a manipulator can be presented by building a grid of points in term of profiles as shown in Figure 2-9. These profiles depict the position, velocity, acceleration, and jerk of the manipulator. The curvilinear presentation of the path parameter is the abscissa s , the velocity is the derivative s' of the path, the acceleration is the second derivative s'' of the path, and the jerk is the third derivative s''' of the path [26]. There are two types of motion curves representation, the trapezoidal curve, and the high order polynomial curve also known as S-curve.

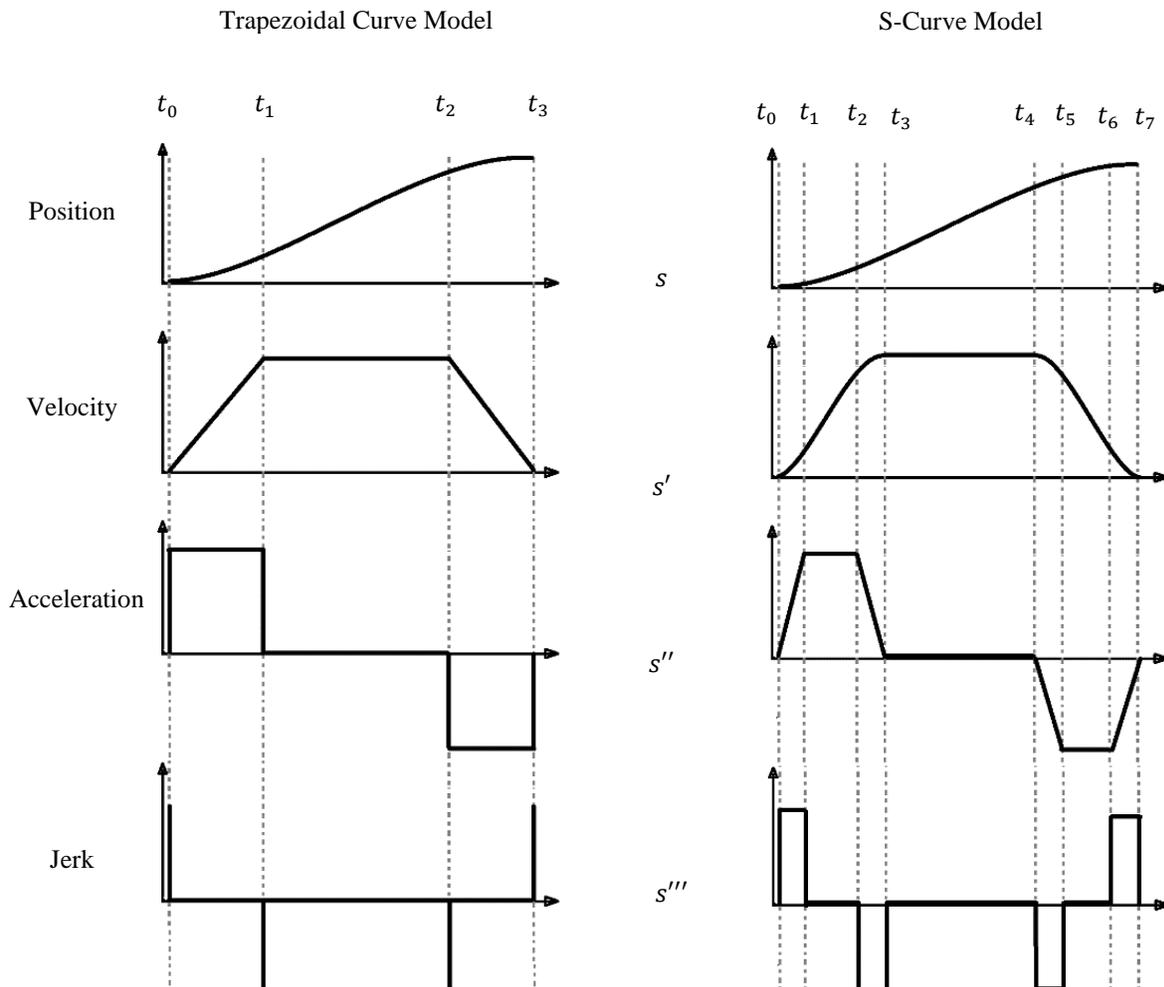


Figure 2-9: Trapezoidal Curve and S-Curve Motion Comparison [26]

2.3.1.1. Trapezoidal Curve Model

The trapezoidal curve is used to represent a velocity model that can achieve fast motion. In this model, the manipulator motors experience a constant acceleration when the velocity increase linearly. Then, it experience zero acceleration when the motor has a constant velocity. Last the motor experience a constant deceleration when the velocity decrease linearly until it reaches a complete stop. This curve has a linear velocity representation and the position is defined as a second order polynomial equation. In addition, it has a sharper representation of acceleration profile, where the jerk factor is at its maximum. As shown in Figure 2-9, the motion trajectory to be connected can be presented into four segments, where the acceleration jumps to a constant value t_0 and t_2 , then retune back to zero at t_1 and t_3 . The jump in acceleration creates an infinite increase in jerk at t_0 , t_1 , t_2 , and t_3 points. That infinite increase in jerk causes major disadvantages like velocity profiles overshoots, residual vibrations excitement, reduction in desired precision, and inaccurate reach to required final destination [32].

This type of curves neglects the links flexibility of the manipulator and the dynamics of the actuators when applying sudden increase in acceleration, causing a number of undesired effects that can harm the actuator motors. For instance, as the robotic actuators cannot generate discontinuous torques, this causes a delay between the joint motion and the reference trajectory to follow, which has an impact on the accuracy of the tracking control. Another undesired effect is the chatter phenomenon at the actuators, where the high sudden change in motion induce a mechanical structure vibration. This chatter effect causes wearing of the mechanical components of the robot and decrease its accuracy over time, also it stress the actuators out. Another undesired effect is that the added error at each joint will prevent the controller from correcting the tracking error [26].

2.3.1.2. S-curve Model

On the other hand, the S-curve is much smoother motion control profile where the acceleration gradually ramp-up to reach the desired velocity then ramp-down to reach a complete stop, making the jerk profile is less sharp. This approach ensure a steady continuity in acceleration throughout the proposed path. As shown in Figure 2-9, the motion trajectory to be connected can be presented into eight segments, where the acceleration gradually increase to accelerate at t_0 to reach a constant acceleration rate at t_1 then gradually decrease at t_2 to reach a steady velocity at t_3 . Then, the acceleration gradually increase in opposite direction to decelerate at t_4 to reach a constant acceleration rate at t_5 then gradually decrease at t_6 to reach a complete stop at t_7 . This curvilinear approach has no oscillation effects and can follow the proposed path with high accuracy. This curve has a smooth linear velocity representation and the position is defined as a third order polynomial equation, which causes a reduction in the jerk effect as jerk will have a finite values [32]. As there is a correlation between the low values of jerk and the accuracy in trajectory following, the reduction in jerk allow the manipulator controls to better perform when tracking the position of each link, mostly through linear interpolation applications, and it reduces the peak-power values [33]. This type of curves takes into consideration the flexibility of the manipulator links as well as the dynamic stresses applied on the actuators. The gradual increase in acceleration creates less torque resistance at the joints, making the motion much smoother. Therefore, the s-curve approach for motion planning is considered the safest approach for the mechanical controls of the robot, and the most precise approach when taking in consideration the reduced jerk factor [26].

There are two approaches used when optimizing the s-curve, the kinematic trajectory algorithm, and the dynamic trajectory algorithm. The kinematic algorithm take in consideration

the design constraints defined by the manipulator manufacturer for the velocity, acceleration, and jerk, where in this algorithm the maximum values allowed for each is considered the ceiling for each profile that the manipulator can follow. This algorithm is considered simple, mathematically feasible, and easy to apply for motion analysis, which reduces the total execution time for the desired task. However, it does not yield a better use of the robotic actuators. On the other hand, the dynamic algorithm considers the dynamic model of the manipulator and takes into consideration all the dynamic constraints like actuator torques, and any actuator jerks at each step throughout the path. This algorithm is much more computationally complex as it features a better capacity of the robotic actuators [26]. For this reason, the kinematic model is better to be followed from motion analysis perspective, where the dynamic model is useful when studying the behavior of capacity of the actuators.

2.3.2. Motion Optimisation Consideration

Gasparetto et al. [26] defined that there are number of optimal criterion that has to be taken in consideration when generating the motion planning. These criteria include the minimum execution time, the minimum actuator effort, and the minimum jerk.

2.3.2.1. Minimum Execution Time

The minimum execution time is an important aspect as it is directly related to high productivity for atomized systems. The faster the manipulator is when accomplishing a required task, the higher the productivity that it will deliver. This aspect take in consideration the position of the manipulator at each target point, the velocity required to reach each target point, as well as the acceleration at each step. However, when considering the minimum execution time, it is important to keep the manipulator acceleration within the manufacturer mechanical kinematic constraints. These kinematic constraints are the maximum value of velocity, acceleration, and jerk

at the joint. If the acceleration exceeds the actuators constraint, the precision in following the desired path will be reduced, and the actuation system can fail due to excessive stress. Therefore, it is important when optimizing the minimum execution time for the manipulator, to be within the mechanical constraints specified by the manufacturer.

2.3.2.2. Minimum Actuator Effort

The minimum actuator effort is another important aspect. It ensures that the power consumption by each joint actuator and motor system is not more than the required power for the task. This aspect focus on the torque at each joint, where the less torque needed at the joint the less power required. This is achieved when generating much more smooth acceleration trajectories that are easier to track and reduce the stress on the robot structure. Considering the energy consumption is much more important for mobile robots, where the available battery power is relatively limited. However, it is also important for the ecofriendly manufacturing model, where designing the required task to consume less energy is much more desirable and economical. Further, having a very smooth acceleration profile reduces the jerk experienced by the actuator, which prevent any damage to the mechanical systems. Therefore, it is important to consider the power consumption and design a smooth torque profile along the whole trajectory to reduce the torque stress at the joints and to ensure a more economical and ecofriendly optimal path.

2.3.2.3. Minimum Jerk

The minimum jerk is one more aspect to consider when optimizing the trajectory planning. Jerk is defined as the time derivative of the acceleration. This jerk effect is generated when the actuators at the joints suddenly accelerate or decelerate in the form of torque vibration. Such vibration can be reduced by making the acceleration profile much smoother. In order to optimize a trajectory with a smooth start and stop, the values of the velocity, acceleration and jerk must be

zero at the start and end points. It was found that reducing the jerk has impact on reducing the error of the trajectory during the tracking phase, as well as reduction in resonance frequency through the manipulator structure. Further, reducing jerk reduces the stresses that the mechanical structure of the robot and actuators experience [26].

2.3.2.4. Minimum Cycle Time

The last factor to consider when optimizing the motion planning for high productivity rate is the cycle time analysis. As time is considered the most important factor in mass production, it is important to track the time required to complete a given task cycle. The cycle time is defined as the average time required to complete a given task from start to finish. There are two levels of cycle time the task cycle time, and the total cycle time. The task cycle time is the duration required to complete a given specific task within a total cycle time. The total cycle time is the duration that the manipulator takes from leaving the home position to perform a number of tasks and returning to the same home position at the end of the manipulator control code [26]. Reducing the task cycle time is important as it will reduce the total cycle time, and it will increase the production rate. The production rate is defined as the number of workpiece completed per unit time.

2.4. Application of Riveting Automation in Aircraft Design

The structure of an aircraft consists of six main components as shown in Figure 2-10. They are the fuselage, wings, horizontal stabiliser, vertical stabiliser, engines and landing gear. Each component is covered by metallic skin that provides a structure support and is aerodynamic efficient.

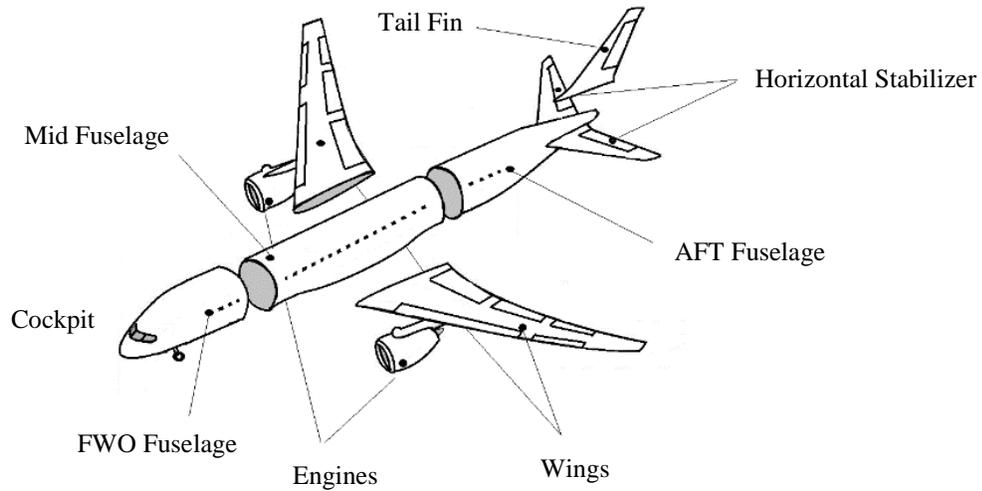


Figure 2-10: Overview of the Commercial Aircraft Main Components [24]

This skin is attached to stringers as shown in Figure 2-11, spars, and flanges using fasteners like rivets, bolts, nuts, where rivets consists the majority of fasteners that are used in skin attachment [24]. As it is shown Figure 2-11, each shape of these stringers has a number of associated challenges related to the way that they are attached to the aircraft skin. Some of these stringers have complex shapes that it is not easy to use standard tools to perform simple riveting tasks.

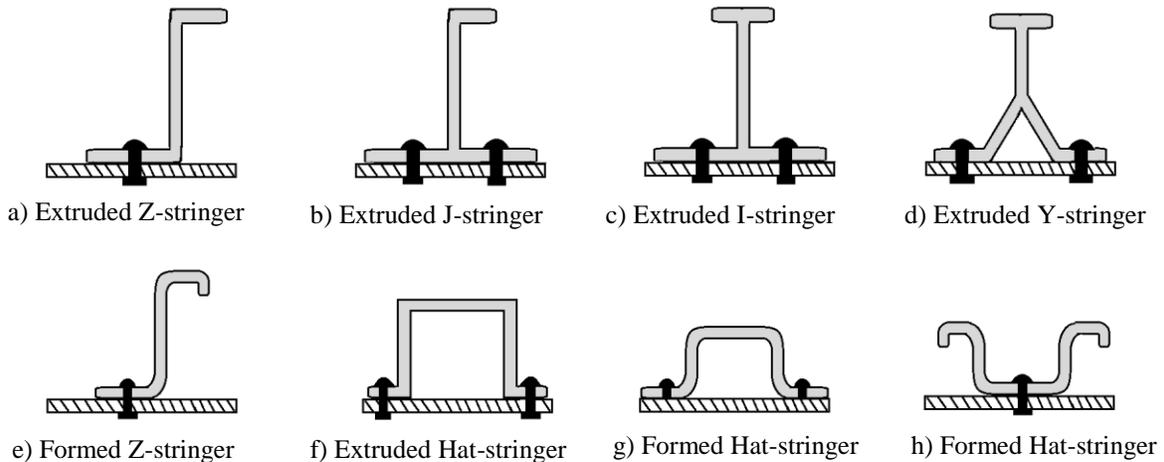


Figure 2-11: Typical Skin-Stringer Panels [24]

2.4.1. Riveting Automation

Rivets are one of the most common types of fasteners used in the aerospace industry. This is due to their mechanical properties and low cost, as well as they are easily to be installed. Currently, manual riveting is the only common approach that is used to perform riveting in confined spaces like aircraft wings and tails. However, manual riveting techniques have much less accuracy when compare to robotic riveting, due to human error and tool calibration. In addition, manual riveting have long-term health impact on joints and cusses long-term hearing loss due to the exposure to the pneumatic gun noise and vibration.

Robotic riveting, on the other hand, has been in use in the automotive industry for the last 20 years and have proven success, so that companies like Jaguar have adopted this technique in their production line [34]. As semi-automated and automated robotic manufacturing systems gets more and more developed and much accurate, they will perform high accuracy riveting at low cost. As a result, the assembly times will be reduced and these robotic systems would be ideal candidates to be widely used in the aerospace industry.

Many attempts were conducted to modify industrial manipulators to include riveting guns positioned at the end-effector of the robot. Jayaweera and Webb [14] created an automated riveting system for aircraft fuselage panels by attaching an interchangeable riveting gun to the end-effector of an industrial robot. Other attempts of using robotics in aerospace assembly for processes like drilling and joining includes ICAM system, Grumman Aerospace Corporation robotic cell, Airbus floor-grid structure assembly cell, Dassault Aviation assembly cells. All of these systems required accurate positioning and fixturing of the components within the cell to overcome the inherent flexibility of robot structures [14].

Today, there are a number of successful automated robotic drilling and riveting systems that are used in the aircraft component assembly stage. Some of these systems are developed by companies like MTorres™, Gemcor™, Electroimpact™, KUKA™, or by aircraft manufacturers.

2.4.1.1. MTorres™ Automated Riveting Systems

The MTorres™ automated riveting systems (Figure 1-1.c [35]) and MTorres Flexible Drilling Head (Figure 1-2.c [36]) are automatic CNC based systems. This system consists of two major components, a stationary gantry and a movable CNC based drilling or riveting systems. The stationary gantry holds the workpiece like wing segments or fuselage segments, and the CNC based system 5 interpolated axis drills or installs the rivets. This system is used for programs like Airbus A350, A380, and Comac C919 [36]. The major disadvantage of this system is that the drilling system is independent from the riveting system, where two separate CNC machines are used to perform riveting installation task. Another disadvantage is the size of the tool complex is large, which limits the usage of this system to installing wing panels and fuselage panels only.

2.4.1.2. Gemcor™ G100 Frame Fastener System

The Gemcor™ Advanced All Electric Aerospace Fastening G100 Frame Fastener system (Figure 2-12) is an industrial manipulator based system. This semi-automatic system is a dual encoder feedback system that is coupled with advanced vision resynchronization technology and burr-free drilling and vision resynchronization. This system is used for riveting door, leading edge, pylon, fuselage, nacelle, frames, and door-frames. This system has the ability of installing solid rivets and 2-piece fasteners [37]. However, the Gemcor™ system has a relatively large tool unit that makes riveting in confined spaces relatively challenging.



Figure 2-12: Gemcor™ G100 Frame Fastener [37]

2.4.1.3. Electroimpact™ E7000 Riveting Cell

The Electroimpact™ introduced the E7000 High-Speed CNC Fuselage Riveting Cell (Figure 2-13) for the Airbus A320 and A400M [38]. This CNC cell system has the ability to perform a drilling and riveting cycle to install 15 rivets per minutes using squeeze process. This system is designed to fasten skin panels to stringers, shear ties, and other internal fuselage components. This system is equipped with a fast tool changer mechanism that enables the system to drill and rivet efficiently.

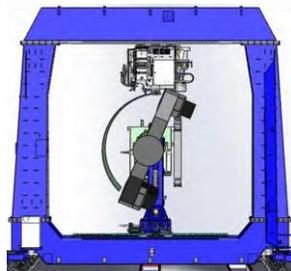


Figure 2-13: Electroimpact™ E7000 High-Speed CNC Fuselage Riveting Cell [38]

In addition, it has a fastener verification laser curtains that validate the installment of the rivets in place and the quality of the rivet. The workpiece is attached to a moving gantry and the CNC drilling and riveting system performs the riveting process. This system has more than one disadvantage, including the large tool size, which presents a challenge in riveting in confined spaces. Another disadvantage is the cell system is only designed for fuselage skin panel assembly, which limit the purpose and application of the system to a very narrow field of applications.

2.4.1.4. KUKA™ Drilling and Riveting Systems

The KUKA™ has a number of drilling and riveting systems that are used by different aerospace manufacturers like Boeing, Bombardier, and Gulfstream. The Fuselage Automated Upright Build (FAUB) system (Figure 2-14) was developed to be used for the Boeing 777, 777X and 737MAX series [39]. This system is used mainly in the fuselage assembly and it consists of two robotic manipulator systems. One system is holding the Multi-Functional End-Effector (MFEE) drilling riveting tool from outside and the other is holding the bucking bar from inside. This system is considered very effective in mass production as it has the ability to install 60,000 rivets per aircraft. However, the large size of the MFEE tool unit is considered a disadvantage when the application is related to compact areas like wings and cockpit skin panel installations.

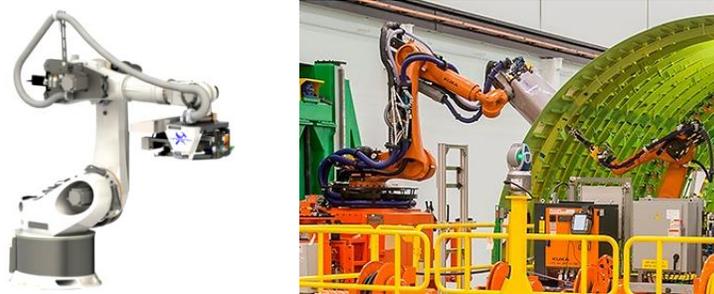


Figure 2-14: KUKA™ Fuselage Automated Upright Build (FAUB) [39]

Another KUKA™ system was developed for Gulfstream and Bombardier is the KUKA Mobile Riveting Platforms (MRPs) as shown in Figure 2-15 [40] & [41]. This system consists of a mobile robot unit that travels along the fuselage and the wing components, and a mobile gantry unit that holds the bucking bar and the workpiece. This system is currently used in the manufacturing of the Bombardier C-Series. Similar to the FAUB, the MFEE tool unit used in this system is considered large in size and performing riveting in confined spaces is considered a challenge. Another disadvantage is that the riveting process has to be tested and tuned manually for the robot to be able to perform a complete riveting task on the following workpiece.

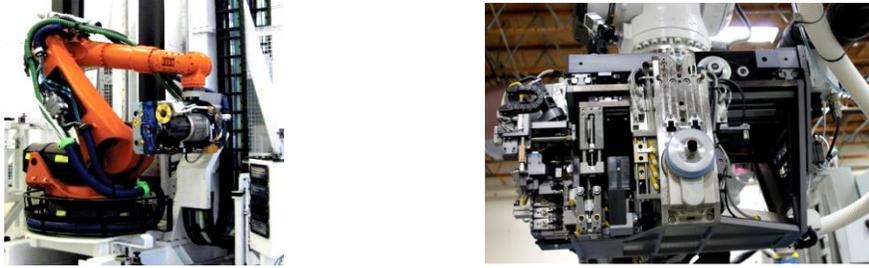


Figure 2-15: KUKA™ Mobile Riveting Platforms (MRPs) [40] & [41]

2.4.2. Challenges for Automated Riveting Systems

The common challenges that face most of the automated robotic riveting today are the large size and weight of the end-effector tool. Aircraft manufacturers are encouraging research and innovation to develop robotic systems with small tool size of less weight to increase the accuracy of the drilling and riveting processes and the ability to access tight spaces as shown in Figure 2-16. For instance, Airbus has conducted the ‘*Airbus Shopfloor Challenge*’ competition as a part of the annual IEEE 2016 International Conference on Robotics and Automation (ICRA) in Stockholm, Sweden. In the objective of this competition, Airbus have highlighted the urge and need to find innovative ways to improve the manufacturing process through the integration of the robotic automation technologies into the production line.



Figure 2-16: Riveted Skin Panel [42]

Airbus have highlighted that despite having some automated processes in their assembly lines, many operations remain manual because of the high number of constraints. These constraints

include space with limited accessibility due to the presence of obstacles like stringers, spars and flanges, and the challenge of large size and weight end-effector. Airbus is currently using systems that use end-effectors capable of performing multiple operations as presented in sub-sections (2.4.1.1) and (2.4.1.3) previously. As a result, the size and weight of the end-effectors become high, which increase the weight to payload ratio.

Their ultimate goal is to be able to develop a lightweight automated robotic system that can perform tasks like drilling, sealing, and riveting, to reduce the repetitive physical stress experienced by workers through aircraft assembly. Further, these systems and tools have to be very accurate, can be integrated easily into their production lines, cost effective, and meet the FAR high reliability standards [42].

For this reason, it is essential to improve robotic riveting system and reduce the size of the end-effector, which will reduce its weight and increase its accuracy, while adopting advanced path-planning techniques that avoid collision with the workpiece while performing the riveting task. This includes the development and adoption of special tooling for some cases where robotic constraints prevent the use of standard tooling. Such improvements will ensure that the robotic riveting system is accurate and able to function in many different scenarios where a small space exists.

CHAPTER 3: ACCESSIBILITY ANALYSIS

This chapter presents a detailed accessibility analysis from the tool (end-effector) accessibility perspective. The objective of the accessibility analysis is to determine a feasible systematic approach to map the accessible volume, allowing the tool to access and reach its target. This new approach will be used when planning the required robotic drilling and riveting tasks to avoid collision with obstacles.

This chapter starts by discussing the tool accessibility and the accessible volume definition. Then, the geometry of the drilling and riveting tools as well as the aircraft parts are classified based on their geometrical shapes. Next, the Global Access Area (GAA) is introduced and defined. The GAA section includes accessibility analysis based on the projection mapping approach, and describes the process to determine the accessible area. This section review the 2D accessibility based on fixed and variable tool orientation. Then, the Global Access Volume (GAV) is introduced and defined. The GAA section includes the accessibility analysis based on the volume extraction approach, and describes the process to determine the accessible volume. This section review the 3D accessibility based on fixed and variable tool orientation. Last, the percentage accessibility definition and calculations are presented.

3.1. Accessibility Definitions

In general, accessibility is defined as the ability of an object to move freely in any direction within a given space that is limited with a defined number of constraints, without any interference. Accessibility is apparent when considering a pair of objects, where one of these objects is in motion relative to the other one with the availability of a number of constraints in a configuration space ζ (*C-Space*).

Let the configuration space ζ (*C-Space*) be within the working range of the robot as presented in Figure 3-1 below. Let the moving source object be the tool G of a given shape and size, such as drilling gun, sealing gun, or a riveting gun with g as the tool's tip. Let the static target be the workpiece (work-object) P , such as a flat skin panel, wing skin, or fuselage skin. Let the static constraints be m number of obstacles O_j of a given shape and size that obstruct the ability to reach target point F_i , such as beams, spars, and trusses. Moving the tool G from its initial position to become in contact with workpiece P at the target point F_i . The point F_i is considered accessible if the tool G can reach point F_i without colliding with any obstacles O_j along the path s .

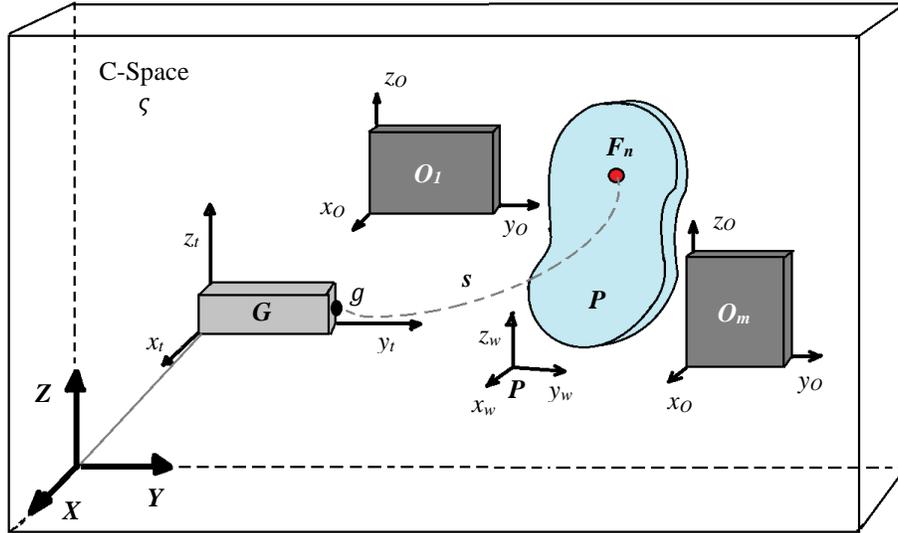


Figure 3-1: Accessibility Representation

Thus, the tool G will not collide with the workpiece P only if:

$$G \cap P = \emptyset \quad (3.1)$$

Further, the tool G will not collide with any obstacles O in the confined space only if:

$$G \cap O_j = \emptyset \quad \forall \quad j = 1, \dots, m \quad (3.2)$$

Therefore, when combining equation (3.1) with (3.2), the general Accessibility *Acc* function representation can be expressed as follows:

$$Acc(F_i, \zeta) = \{s : G \cap P = \emptyset \wedge G \cap O_j = \emptyset\} \quad \forall \begin{cases} i = 1, \dots, n \\ j = 1, \dots, m \\ F_i \in P \end{cases} \quad (3.3)$$

where, s is the path between the tool's tip g position and the target point F_i . This function depicts the accessibility of the tool G as it approaches the workpiece P at target point F_i in the C-Space set ζ . The tool follows the path s without colliding with any of the surrounded obstacles O_j .

To define accessibility in terms of geometry, the geometry of the tool, the robot, the workpiece, and the obstacles can be presented in terms of convex polyhedral. The tool G can be presented in a union of convex polyhedral as follows [43]:

$$G = \bigcup_{i=1}^{nG} G_i \quad \forall \quad i = 1, \dots, nG \quad (3.4)$$

with

$$G_i = \{g \in \mathbb{R}^3 | E_i g \leq b_i\} \quad \forall \quad \begin{cases} i = 1, \dots, nG \\ E_i \in \mathbb{R}^{f_i \times 3} \\ b_i \in \mathbb{R}^{f_i} \end{cases} \quad (3.5)$$

where, g is the tool's tip point in Cartesian real space \mathbb{R}^3 , E_i is the rotation matrix in local coordinates of the robot, b_i is the tool's tip in local coordinates of the robot, nG is the number of polyhedra in the set G , f_i is the number of faces in G_i , and i is the i^{th} polyhedron in set G .

In this case, the tool G in equation (3.4) presents all the links of the robotic arm with the drilling or riveting gun is attached to it as an end-effector. However, the focus of the accessibility analysis in this thesis is about the end-effector tool accessibility. The RobotStudio™ embedded functions validate the collision part of the equation (3.5) for the robotic arm, but it does not validate the tool collision with the workpiece part of the equation (3.5). Therefore, the equation (3.5) could be simplified further when dealing with the end-effector tool accessibility:

$$G = \{g \in \mathbb{R}^3 | E g \leq b\} \quad \forall \quad \begin{cases} E \in \mathbb{R}^{f \times 3} \\ b \in \mathbb{R}^f \end{cases} \quad (3.6)$$

For the workpiece P and the obstacles O_j , they can be presented as a union of convex polyhedral Q , such that

$$Q = \bigcup_{j=1}^{nQ} Q_j \quad \forall \quad j = 1, \dots, nQ \quad (3.7)$$

with

$$Q_i = \{g \in \mathbb{R}^3 | C_j y \leq p_j\} \quad \forall \quad \begin{cases} j = 1, \dots, nQ \\ C_j \in \mathbb{R}^{q_j \times 3} \\ p_j \in \mathbb{R}^{q_j} \end{cases} \quad (3.8)$$

where nQ is the number of polyhedra in the set Q , q_j is the number of faces in Q_j , and j is the j^{th} polyhedron in set Q .

These two polyhedral sets are collision-free only if

$$G_i \cap Q_j = \emptyset, \quad \forall \quad \begin{cases} i = 1, \dots, nG \\ j = 1, \dots, nQ \end{cases} \quad (3.9)$$

This implies that G_i set will never experience collision with Q_j set unless there is no point in $g_{(i,j)} \in \mathbb{R}^3$ that can satisfy the following condition:

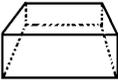
$$\begin{pmatrix} E_i \\ C_j \end{pmatrix} g_{(i,j)} \leq \begin{pmatrix} b_i \\ p_j \end{pmatrix} \quad (3.10)$$

3.2. Geometrical Classification

A geometry in 3D space can be defined in form of polyhedrons of faces, sides, and vertices. The geometrical face is an area of a given shape, while a side is a line or a curve that forms a parameter around the geometrical face, and a vertex is a point where two or more lines, edges, or curves meet. In order to conduct the accessibility analysis, the geometric shapes of the tools as

well as the workpieces have to be classified into the closest simplified geometrical representations of their shapes for simplicity. These simplified geometrical representations include cube, cylinder, cone, and sphere, are shown in Table 3-1.

Table 3-1: The Faces, Sides, and Vertices Count for Some Geometries

Simplified Shape	Geometry	Number of Faces/Surfaces (f_i or q_j)	Number of Sides/Edges	Number of Vertices
Cube / Rectangular Parallelepiped		6	12	8
Cylinder/ Elliptic Cylinder		3	2	0
Cone		2	1	1

The values presented for the number of faces in Table 3-1, can be used in equation (3.6) based on the shape of the tool G , and in equation (3.8) based on the shape of the obstructive objects Q (the obstacles O_j and the workpiece P).

When checking for accessibility, it is very important that the faces, the sides, and the vertices of both the tool G experience no intersection with the faces, the sides, and the vertices of the workpiece P and other obstacles O_j combined as shown previously in equations (3.9) and (3.10).

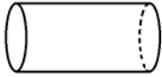
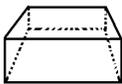
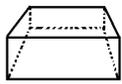
In order to solve the accessibility problem, the classification of the aircraft parts and tools of interest based on their simplified geometrical shapes is presented in the following subsections.

3.2.1. Tool Geometry

Each tool has its own geometrical shape and size that has a direct impact on the accessibility of that tool to reach a specific target location in a confined space. The smaller the tool, the higher the chance of its successful accessibility. This is also true for the tool shape when it is less complex and closer in shape to the basic geometrical shapes like cube, rectangular parallelepiped, or

cylinder, the higher the chance of its successful accessibility. For those reasons, it is better to classify the tools used into their simplest geometrical elementary representation as shown in Table 3-2.

Table 3-2: Tool Geometrical Classification

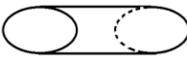
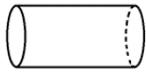
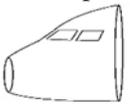
Tool	Volume Approximation	Equation	Geometrical representation
Drilling Gun	Cylinder	$V = \pi r^2 h$	
Sealing Gun	Cylinder	$V = \pi r^2 h$	
Riveting Gun	Rectangular Parallelepiped	$V = l \times h \times w$	
Bucking Bar	Rectangular Parallelepiped	$V = l \times h \times w$	

The simplified shapes presented above are to be used in the geometrical accessibility analysis in the following section.

3.2.2. Part Geometry

Similar to the tool geometry, each aircraft part has its own geometrical shape and size which affect tool accessibility given that there are surrounding obstacles like spars, stringers, ribs, and flanges. The larger the part, the higher the chance of the tool's accessibility. This is also true for the part's shape when it is less complex with less surface curvatures and closer in shape to the basic geometrical shapes like cone, cylinder, cube, or a rectangular parallelepiped, the higher the chance of its successful accessibility. For these reasons, it is better to classify the aircraft part of interest into their simplest geometrical elementary representation as shown in Table 3-3.

Table 3-3: Aircraft Part Geometrical Classification

Aircraft Segment	Volume Approximation	Equation	Geometrical representation	Curvature Type
 <p>Wing</p>	Rectangular Parallelepiped	$V = l \times h \times w$		0 curvature
 <p>Empennage</p>	Elliptic Cylinder	$V = \pi \times a \times b \times h$		1 curvature
 <p>Fuselage</p>	Cylinder	$V = \pi r^2 h$		1 curvature
 <p>Cockpit</p>	Cone	$V = \frac{1}{3} \pi r^2 h$		2 curvatures

The simplified shapes presented above are to be used in the geometrical accessibility analysis in the following section.

Another important aspect to validate is the crowdedness that each aircraft part experiences as known based on the estimated number of constraints and obstacles available in each part. In Table 3-4, a crowdedness check is presented to identify any possible accessibility limitations that is available when joining structure to skin and skin to skin.

Table 3-4: Crowdedness and Accessibility Check for each Aircraft Part

Aircraft Part	Structure to Skin Joining	Skin to Skin Joining
Wing	Limited accessibility	Accessible
Empennage	Not accessible	Accessible
Fuselage	Accessible	Accessible
Cockpit	Accessible	Accessible

It is important to note that the crowdedness and accessibility check presented above takes into consideration the shape of the tool attached to the robotic arm, as well as the robotic reach.

3.3. Global Accessibility Area (GAA)

The global accessibility area (GAA) is defined as the boundary area within the free configuration ξ (C-free) in the configuration space ζ (C-Space). This boundary is where the tool can move freely in any direction without colliding with any obstacle. The tool and obstacles are in global coordinates, when projected on a 2D plane vertical to the tool's motion. This approach is used when studying the accessibility of the tool through 2D projection view.

Let the tool G be the moving source object, the workpiece P be a static target with the target point F_i , and the obstacles O_j be static obstructions that obstruct the ability of the tool to reach target point F_i . The position of the tool, the workpiece, and the obstacles are all defined in global coordinate system. The C-free ξ is the area within the C-Space ζ that is not occupied by the obstacles O_j and the workpiece P , as shown in Figure 3-2.

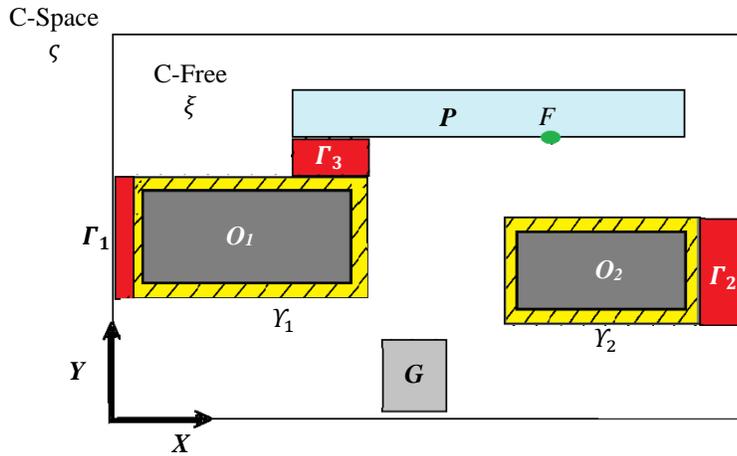


Figure 3-2: Accessibility Representation in 2D

The C-free ξ can be expressed as:

$$\xi = \zeta - \left(\sum_1^m O_j + P \right) \quad \forall \quad j = 1, \dots, m \quad (3.11)$$

The dashed zones *highlighted in yellow* Y in Figure 3-2 present the safety boundary areas around the obstacles. This safety boundary is important as it ensures that the tool will not get very close or collide the obstacles. The zones *highlighted in red* Γ are the areas that are not accessible for the tool, due to the tool size and shape. The yellow red zones are part of the C-Free ξ space.

The global accessibility area (GAA) by definition is the area that allows the tool to move freely without any interference with the obstacles in the C-Space. This GAA is the area excluding the yellow safety Y zones and the red inaccessible Γ zones from the C-Free space as shown in Figure 3-3.

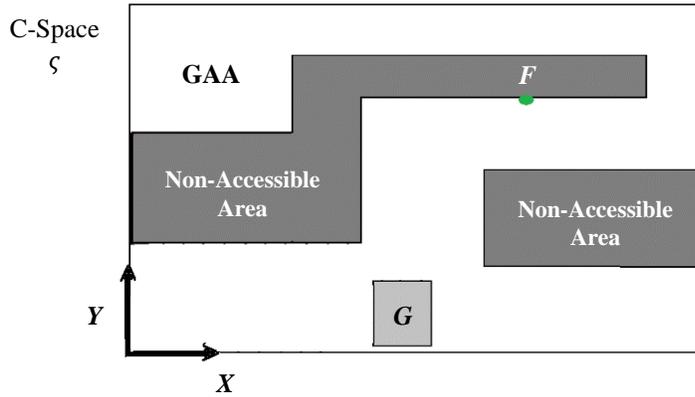


Figure 3-3: Global Accessibility Area Representation

Therefore, the GAA can be expressed as the following:

$$GAA = \xi - \left(\sum_1^m Y_j + \sum \Gamma_k \right) \quad \forall \quad \begin{cases} GAA \in \xi \\ GAA \in \mathbb{R}^2 \end{cases} \quad (3.12)$$

where, j is the j^{th} element, and k is the k^{th} element.

The accessibility of the tool within the GAA boundary can then be presented based on equation (3.3) as follows:

$$Acc(F_i, \xi) = \{s : G \cap (\xi - GAA) = \emptyset\} \quad \forall \quad \begin{cases} i = 1, \dots, n \\ G \in GAA \end{cases} \quad (3.13)$$

where, n is the number of targets F .

In order to solve for accessibility the GAA has to be depicted accurately on a 2D plane. The 3D objects' geometric properties have to be mapped onto a 2D plane. The GAA can then be tested for fixed and variable tool orientation to illustrate all possible solutions for the GAA. Thus, the resulting GAA can then be used to generate the collision-free path for the tool.

3.3.1. Geometrical Mapping

Geometrical mapping is the process of projecting 3D objects into 2D planes. This process is an important tool for accessibility analysis as it is used to construct the feasibility mapping of the objects in 3D space to a 2D plane to check for accessibility. To be able to construct the feasibility mapping, it is important to be able to transform the points and vectors from one coordinate system to another (Refer to Appendix B for coordinate's transformations).

There are two types of projection mapping: the perspective projection, and the orthographic projection. The perspective projection is the condition when an object is seen from a distance, where the object tend to reflect a decrease in the measurements at the focal points. On the other side, the orthographic projection ignores the depth element and reflect the measurement with a much better accuracy. The orthographic projection has all the projection lines are parallel to each other and perpendicular to the 2D projection plane. In engineering drawings, the orthographic projection is the one commonly used to represent objects in accurate-to-scale measurements. This type of projection is ideal when performing accessibility analysis, as it reflects the correct measurements of the objects projected onto the 2D plane.

The matrix notation of the orthographic projection can be presented as follows:

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim \begin{bmatrix} x_h \\ y_h \\ z_h \\ w \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.14)$$

where, $[x_h \ y_h \ z_h \ w]^T$ represents the homogeneous vertex coordinates. For homogenous condition $w = 1$, then

$$x = \frac{x_h}{w} = X \quad \text{and} \quad y = \frac{y_h}{w} = Y \quad \text{and} \quad z = Z = 0 \quad (3.15)$$

Considering the scale factor when presenting the matrix notation of the orthographic projection, let \bar{d} be the distance from the object, and f be the focal representation. The scale term can be denoted as f/\bar{d} . Now, updating the matrix notation, resulting in the following matrix:

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim \begin{bmatrix} x_h \\ y_h \\ z_h \\ w \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{d} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.16)$$

For the homogenous condition $w = 1$, then

$$x = \frac{x_h}{w} = X \left(\frac{f}{\bar{d}} \right) \quad \text{and} \quad y = \frac{y_h}{w} = Y \left(\frac{f}{\bar{d}} \right) \quad \text{and} \quad z = Z = 0 \quad (3.17)$$

3.3.2. GAA with Fixed Tool Orientation

In order to find the solution for tool accessibility in 2D space, the GAA accessible boundary has to be determined using the orthographic projection mapping approach. The orthographic projection can be accurately generated either by using the CAD geometry representation of the objects or by projecting the object features on a 2D plane perpendicular to the actual 3D objects. The goal of the orthographic projection is to simplify the required analysis for tool accessibility when dealing with 2D space. It is important to note that this approach is valid for surfaces that have zero and one curvatures only, such as flat plates, wing, empennage, and fuselage skin panels. Surfaces that have two or more curvatures cannot be accurately projected on a 2D plane. To further simplify the analysis, the tool orientation was fixed throughout the search process starting at this step, until the inaccessible targets are identified. The variable tool orientation is presented in the following subsection.

For cases with zero curvature, the generated 2D orthographic projection reflects the actual size of the objects on the flat panel. This size representation can have the same ratio of the actual object or with a given scale ratio representation. If the projection is scaled up or down, the scale factor shall be taken into consideration through the path planning process. Otherwise, the size representation will depict the available objects with high position accuracy.

For the cases with one curvature, like the aircraft fuselage section, there will be two orthographic projections due to the surface's curvature as shown in Figure 3-4 and Figure 3-5. By conducting the accessibility analysis or generating the path for the tool based on the GAA approach, both projections have to be considered at all steps. This ensures that the generated result will not interfere with the obstacles or collide with the workpiece itself.

The orthographic projections of a curved surface like a fuselage section that has one Z-beam, attached as an obstacle, can be depicted in the following figures. The Figure 3-4 presents the front projection of the Z-beam on the 2D plane, where the area highlighted in red is the inaccessible area when approaching the beam. The projection 2D plane is projected parallel to the Z-beam base, where the tool will be in perpendicular position to drill and rivet the workpiece. The Figure 3-5 presents the side projection of the Z-beam on the 2D plane, where the area highlighted in red is the inaccessible area when approaching the beam, and the areas highlighted in yellow are the safety zones that the tool should avoid to prevent collision with the workpiece. The front and side projections were selected as they provide all the required information to create the GAA required for the accessibility analysis.

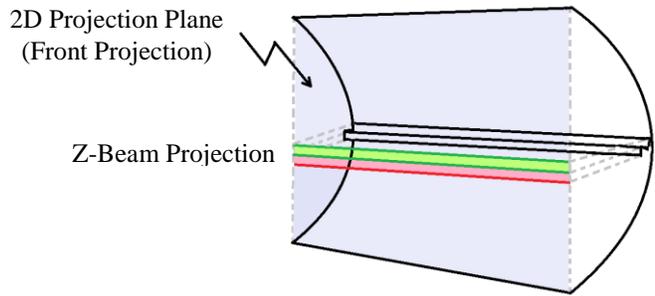


Figure 3-4: Front Projection of a Curved Surface

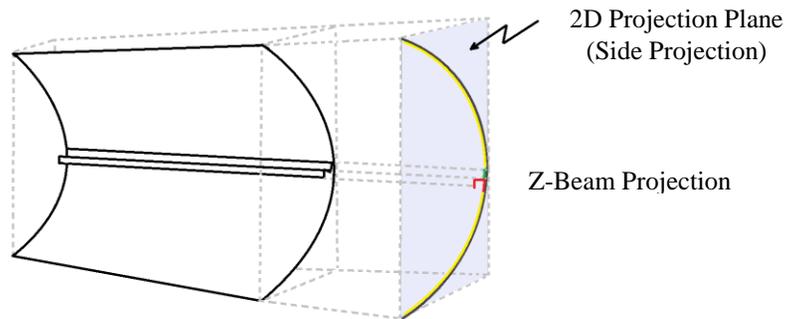


Figure 3-5: Side Projection of a Curved Surface

The Figure 3-6 presents a closer view to the accessible area around the Z-beam in the side projection.

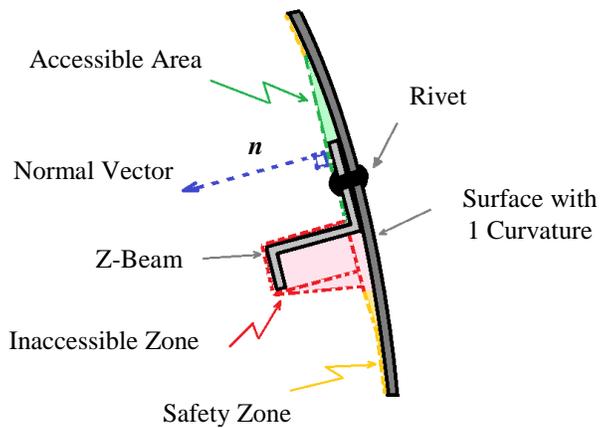


Figure 3-6: Close View of the Z-beam on the Side Projection of a Curved Surface

It is important to note that when dealing with a surface with one curvature, the tool should always be perpendicular to the surface of the workpiece and the surface of the Z-beam where the

rivet will be placed for drilling and riveting. This ensures that the projection plane will always reflect the actual size of the surface when planning for drilling and riveting even if the surface is curved. Furthermore, this will ensure that the tool's tip is able to reach the required point and is accurately depicted during the tool accessibility analysis. In addition, the projection plane orientation can be then used to correct the tool's angle of approach to ensure that the tool orientation is normal to the surface of the workpiece.

Another observation that was made is that in most of the aerospace assembly riveting cases, the spars, ribs, stringers and frame are always oriented either at 0° or 90° relative to the rivet pattern. This simplifies the requirement for obstacles orientation correction through the tool accessibility analysis.

Once the projection is determined, and the position of all the objects is calculated, the GAA boundaries can be determined. Let the tool have a square-shaped layout of the vertices' points $p_A, p_B, p_C,$ and p_D . Let the GAA boundary be outlined by p_1 to p_6 as shown in Figure 3-7.

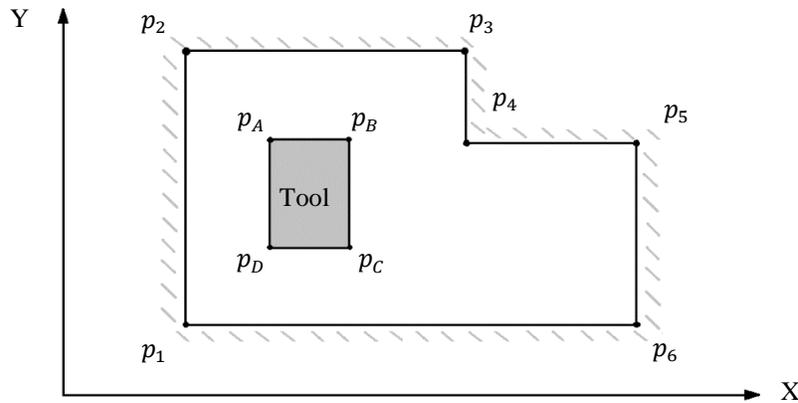


Figure 3-7: Tool in GAA with Boundary Limits

The GAA boundary can be presented as follows:

$$GAA = [p_1, p_2, \dots, p_m]^T \tag{3.18}$$

where, m is the number of vertices points that define the GAA boundary. As the tool move within the GAA boundary, there is no collision at all positions. If the tool touches or crosses the GAA boundary, then there will be collision.

To validate if the tool is in collision with the GAA boundary, the tool's linear representation with the GAA boundary shall be solved as a system of linear equations to determine if an intersection occurred. To find the line segment intersection from an algebraic point of view, the following approach can be used. Let $p_1 = (x_1, y_1)$ and $p_2 = (x_2, y_2)$ be the two points that define the segment of the first line ℓ_1 . Let $p_3 = (x_3, y_3)$ and $p_4 = (x_4, y_4)$ be the two points that define the segment of the second line ℓ_2 . These two lines can be represented in one of the three cases shown in Figure 3-8.

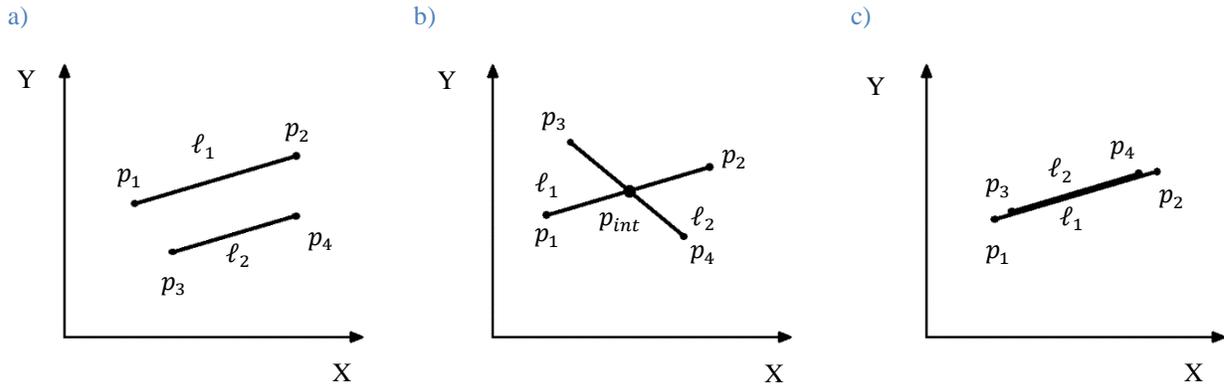


Figure 3-8: Linear Intersection for: a) No Solution, b) One Solution, c) Infinite Solutions

The equation of the first line ℓ_1 can be expressed as:

$$y_{\ell_1} = \frac{y_1 - y_2}{x_1 - x_2}x + \frac{x_1y_2 - x_2y_1}{x_1 - x_2} \quad (3.19)$$

Also, the equation of the second line ℓ_2 can be expressed as:

$$y_{\ell_2} = \frac{y_3 - y_4}{x_3 - x_4}x + \frac{x_3y_4 - x_4y_3}{x_3 - x_4} \quad (3.20)$$

The following two conditions become true only if both segments are overlaying each other:

$$\frac{y_1 - y_2}{x_1 - x_2} = \frac{y_3 - y_4}{x_3 - x_4} \quad (3.21)$$

and,

$$\frac{x_1 y_2 - x_2 y_1}{x_1 - x_2} = \frac{x_3 y_4 - x_4 y_3}{x_3 - x_4} \quad (3.22)$$

where, $x_1 \leq x_3 \leq x_2$ and $y_1 \leq y_3 \leq y_2$, or $x_1 \leq x_4 \leq x_2$ and $y_1 \leq y_4 \leq y_2$. In this case, there will be infinite solutions within the intersection segment as depicted in Figure 3-8.c. If equation (3.21) is only true, then the two lines are parallel to each other and will have no solutions as depicted in Figure 3-8.a.

For the case where the two segments intersect as depicted in Figure 3-8.b, the intersection point x_{int} can be found using the following equation:

$$x_{int} = \frac{x_1(x_3(y_2 - y_4) + x_4(y_3 - y_2)) + x_2(x_3(y_4 - y_1) + x_4(y_1 - y_3))}{(x_1 - x_2)(y_3 - y_4) + (x_4 - x_3)(y_1 - y_2)} \quad (3.23)$$

If the x_{int} has a solution, the y_{int} can be found by substituting the x_{int} in equations (3.19) and (3.20). In addition, the resulting x_{int} value must be between the $x_1 \leq x_{int} \leq x_2$ and $x_3 \leq x_{int} \leq x_4$, to consider that the segments are intersecting at a single point $p_{int} = (x_{int}, y_{int})$.

The tool is considered to have a collision only if the intersection has a single or infinite solutions at any of the tool's shape layouts. In this case, the target point is considered inaccessible at the given tool's orientation. The following subsection will present the solution for the tool's orientation.

3.3.3. GAA with Variable Tool Orientation

In order to present any object accurately in 2D space, the position as well as the orientation of that object shall be determined. Both the position of the object and the orientation in space shall

be with respect to the global frame of reference. When associating a coordinate frame in a given configuration to a given object, the orientation of that object can be expressed in terms of the global frame of reference using a rotation matrix. This rotation matrix presents each of the axis of the object locally in terms of the global frame reference.

In 2D space, the rotation matrix \mathbf{R} can be expressed as:

$$\mathbf{R} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (3.24)$$

where, θ is the angle of the object relative to the global frame reference where θ is in the clockwise direction. The position of the object can then be expressed as:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \mathbf{h} \quad (3.25)$$

where, \mathbf{h} is the translation vector, x and y are the local position coordinates, while x' and y' are the position in global frame reference.

In terms of 2D homogeneous transformation matrix \mathbf{T} , the object position and orientation can be expressed as:

$$\mathbf{T}_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & x_i \\ \sin \theta_i & \cos \theta_i & y_i \\ 0 & 0 & 1 \end{bmatrix} \quad (3.26)$$

where, i is the i^{th} object.

Let the object be the tool, where the tool in the section 3.3.2 had a fixed orientation. Now, let this tool have a variable orientation at all the target points F where accessibility was determined to be inaccessible based on the approach used in section 3.3.2. As the tool changes its orientation, the line intersection equations are revalidated to determine if there is collision at all angles. The orientation increments by 10° interval. If the tool completes a 90° rotation and still experience linear intersection, then this point is inaccessible at all angles.

3.4. Global Accessibility Volume (GAV)

The global accessibility volume (GAV) is defined as the boundary volume within the free configuration ξ (C-free) in the configuration space ζ (C-Space). In which, the tool is able to move freely in any direction without colliding with any obstacle where the tool and obstacles are in global coordinates. This approach is used when studying the accessibility of the tool in 3D space.

Let the tool G be the moving source object, the workpiece P be a static target with the target point F_t , and the obstacles O_j be static obstructions that prevents the tool from reaching target point F_t . The position of the tool, the workpiece and the obstacles are defined in global coordinate system. The C-free ξ is the area that is not occupied by the obstacles O_j and the workpiece P within the C-Space ζ as shown in Figure 3-9.

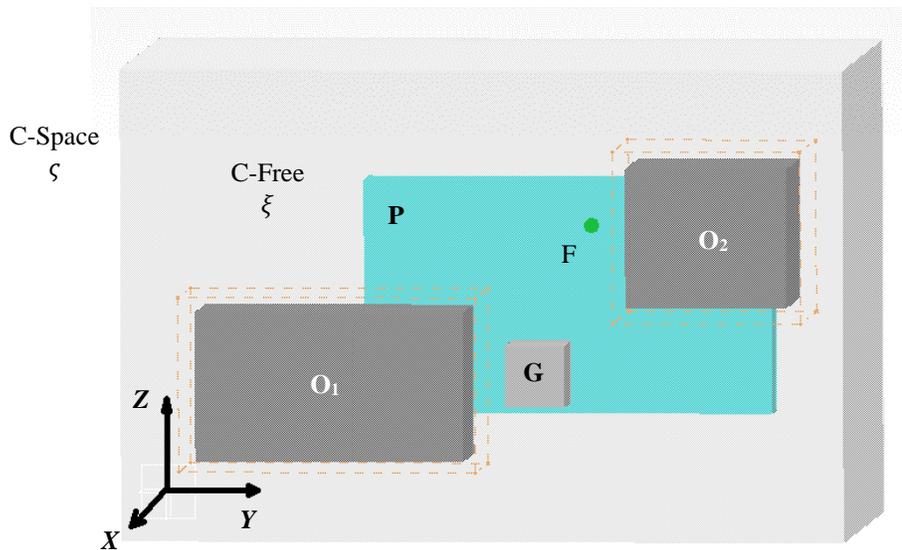


Figure 3-9: Accessibility Representation in 3D

A number of variables that have to be determined in order to determine the accessible boundary volume within the C-Space ζ . The first variable is to determine the position and orientation of all objects in the C-Space ζ using the global coordinate system. Obtaining this

variable is important to determine the boundaries of the accessible volume. The second variable is to determine the safety zones around the obstacles. These safety zones (the volume shown in dashed orange lines in Figure 3-9) include the minimum distance that is allowed for the tool to get closer to the obstacle without any interference. These safety boundaries are important to ensure that the tool does not collide with the obstacles. The last variable is to determine the size of the tool and the regions between objects and the C-Space ζ boundaries that will limit the tool accessibility. The global accessibility volume boundaries are then calculated when combining the results of the three variables. The Figure 3-10 presents the inaccessible zones from top view, side view, and left view perspective.

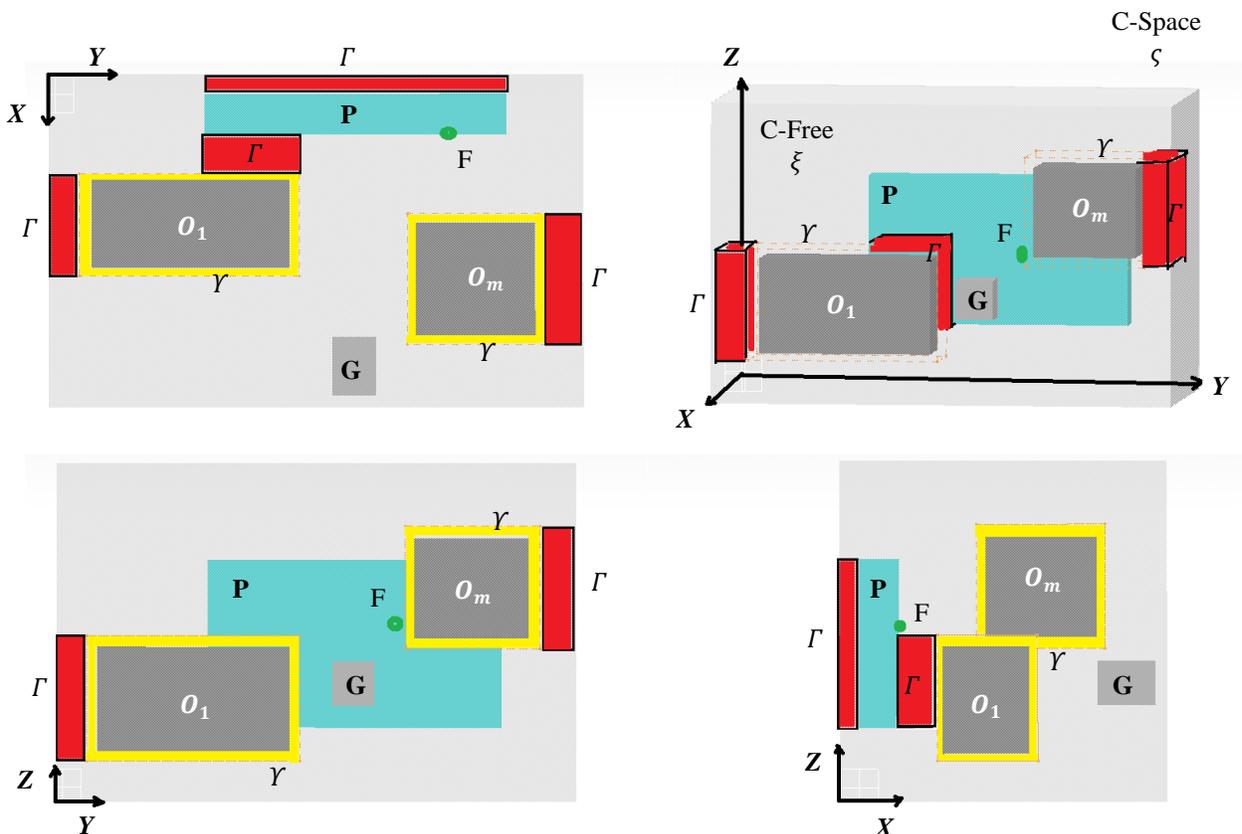


Figure 3-10: Inaccessible Zones Representation in 3D with Top, Side, and Left Views

The dashed zones highlighted in yellow Y in Figure 3-10 are the safety boundary area around the obstacles. The zones highlighted in red Γ represent the inaccessible space by the tool, due to the tool size and shape. The yellow dashed zone and the red zone are part of the C-Free ξ space.

The global accessibility volume (GAV) by definition is the space volume that allows the tool to move freely without any interference with surrounding obstacles in the C-Space. This GAV is the volume space that excludes the yellow dashed safety Y zones and the red inaccessible Γ zones from the C-Free space as shown in Figure 3-11.

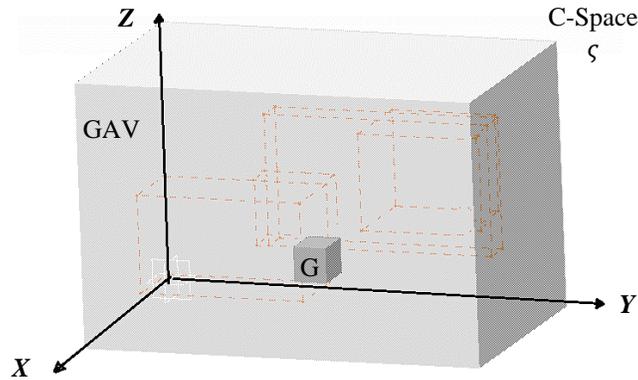


Figure 3-11: Global Accessibility Volume Representation

Therefore, the GAV can be expressed as the following:

$$GAV = \xi - \left(\sum_1^m Y_j + \sum \Gamma_k \right) \quad \forall \quad \begin{cases} GAV \in \zeta \\ GAV \in \mathbb{R}^3 \end{cases} \quad (3.27)$$

where, j is the j^{th} element, and k is the k^{th} element.

The accessibility of the tool within the GAV boundary can then be presented based on equation (3.3) as follows:

$$Acc(F_i, \zeta) = \{s : G \cap (\zeta - GAV) = \emptyset\} \quad \forall \quad \begin{cases} i = 1, \dots, n \\ G \in GAV \end{cases} \quad (3.28)$$

where, n is the number of targets F .

The tool has full accessibility within the GAV to reach the required location on the workpiece. This is accomplished only if there is no solution for the intersection of faces, the sides, or vertices of the tool with the workpiece or the obstacles.

In order to solve for accessibility using GAV, the 3D space has to be utilized to depict the GAV accurately. Thus, the accessible volume has to be extracted by deducting the objects geometries, the safety Y zones and the inaccessible I zones from the C-Free ξ volume. The GAV can then be tested for fixed and variable tool orientation to depict all possible solutions. The resulting GAV then can be used to generate the collision-free path for the tool.

3.4.1. Geometrical Volume Extraction

The geometrical volume extraction is a process used to determine the volume boundaries based on the object's vertices location, edges, and faces. The importance of the geometrical accessibility analysis is that it supports the equations for path calculation. Determining if any interferences exist between the constraints in the surrounding space and the workpiece part, ensures a collision-free path. Further, it will provide an ability to quantify the accessibility success when planning a desired path. This will support the goal of better path planning management and faster path execution time. Another benefit is that this will provide a quantitative value to support the aerospace parts design process to develop better designs that can be assembled using robotic automation.

In order to extract the accessible volume, the operational boundaries of the robot must be identified. This will provide the foundation to determine the outline of the configuration space (C-space ζ) as the accessible volume must be within the robot operational range. Then, the position and the orientation of the robot must be determined as well as the position and orientation of the tool, the workpiece and the obstacles. Once all these parameters are determined, the virtual

representation of these objects can then be presented in the CAD 3D environment up to scale. Next, similar to Figure 3-10 as presented in the definition of the GAV previously, a parallelepiped volume is added within the robot operational range that includes the tool, the workpiece and all the obstacles. Then, all the safety regions are added to envelop all the obstacles available within the parallelepiped volume. Following this step, all inaccessible regions are highlighted based on the tool size. These inaccessible regions are the regions between the walls of the parallelepiped volume and the objects inside the parallelepiped volume, and the regions that is between the objects, where the tool size cannot fit within it to reach the target points. Once all the regions are identified, the accessible volume is extracted based on equation (3.27) by subtracting all the regions that are occupied by the objects, the safety regions and the inaccessible region. The remaining GAV is then determined based on the vertices, the edges, and surfaces that present the accessible boundary of the GAV.

3.4.2. GAV with Fixed Tool Orientation

In order to find the solution for tool accessibility in 3D space, the GAV accessible boundary has to be determined using the volume extraction approach. The volume extraction can be determined using the CAD 3D geometry representation of the objects and identifying safety zones and the inaccessible zones bases on the tool size as mentioned in the previous subsection. The goal of the volume extraction is to determine all the vertices points, the edges and the surfaces that outline the GAV. This approach is valid for surfaces that have zero, one and two curvatures or more, such as flat plates, wing, empennage, fuselage skin panels and cockpit. To simplify the analysis, the tool orientation was set to be fixed through the search process at this step, until the inaccessible targets are identified. Then, in the following subsection, the variable tool orientation is presented.

When extracting the accessible volume around obstacles such as spars, stringers, and ribs, the shape of these obstacles can be simplified in a form of L-beam, Z-beam, C-beam, I-beam, J-beam, or a Hat-beam. The accessible regions can be determined based on the surface that the tool has to reach to perform drilling or riveting tasks as shown in Figure 3-12 and marked in green. The inaccessible regions are the regions that the tool cannot reach or they obstruct the path to the target point on the beam surface. In the drilling and riveting cases, the tool must always be perpendicular to the surface that the rivet will be placed in. This is also true even if the beam is on a curved surface, the tool orientation will be corrected for the surface curvature to be normal to the surface of the beam at the target point.

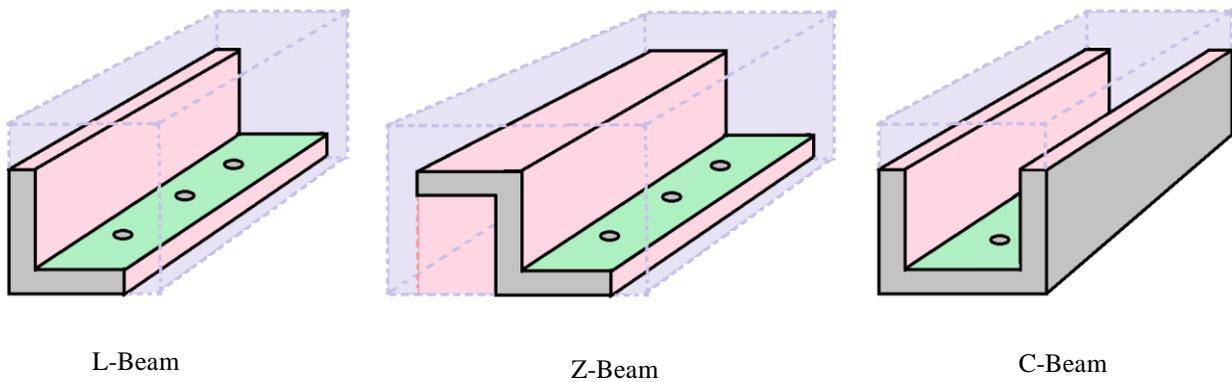


Figure 3-12: Global Accessibility Volume of the Accessible Region for Different Beam Shapes

To determine the GAV boundaries, let the tool have a cube shape layout of the vertices points $p_A, p_B, p_C, p_D, p_E, p_F, p_G,$ and p_H . Let the GAA boundary be outlined by p_1 to p_{24} as shown in Figure 3-13.

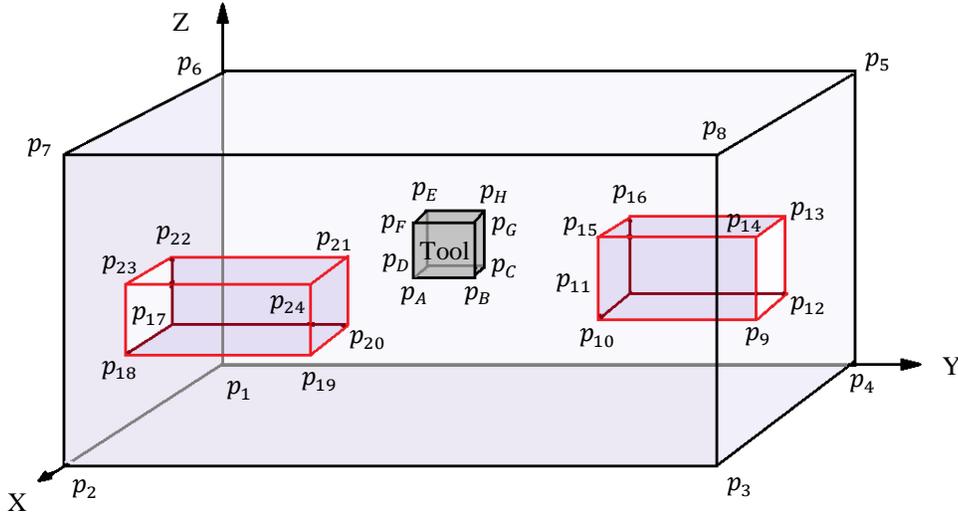


Figure 3-13: Tool in GAV with Boundary Limits

The GAV boundary can be presented as the following

$$GAV = [p_1, p_2, \dots, p_m]^T \quad (3.29)$$

where, m is the number of vertices points that define the GAV boundary. Refer to Appendix F for vertices representation for basic shapes. As the tool moves within the GAV boundary, there is no collision at all positions. If the tool touches or crosses the GAV boundary faces, edges or vertices, there will be a collision condition. To validate if the tool is in collision with the GAV boundary, the tool linear representation with the GAV boundary representation shall be solved as a system of linear equations to determine if an intersection occurred at the vertices, edges, or faces level.

To find the line segment intersection in 3D space from an algebraic point of view, the following approach can be used. Let $p_1 = (x_1, y_1, z_1)$ and $p_2 = (x_2, y_2, z_2)$ be the two points that define the segment of the first line ℓ_1 . Let $p_3 = (x_3, y_3, z_3)$ and $p_4 = (x_4, y_4, z_4)$ be the two points that define the segment of the first line ℓ_2 . The equation of the first line ℓ_1 can be expressed as:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} + b_1 \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{bmatrix} \quad \forall b_1 \in \mathbb{R} \quad (3.30)$$

Also, the equation of the second line ℓ_2 can be expressed as:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} + b_2 \begin{bmatrix} x_4 - x_3 \\ y_4 - y_3 \\ z_4 - z_3 \end{bmatrix} \quad \forall b_2 \in \mathbb{R} \quad (3.31)$$

The following equation can be then used to check for intersection:

$$\begin{aligned} x_1 + b_1(x_2 - x_1) &= x_3 + b_2(x_4 - x_3) \\ y_1 + b_1(y_2 - y_1) &= y_3 + b_2(y_4 - y_3) \\ z_1 + b_1(z_2 - z_1) &= z_3 + b_2(z_4 - z_3) \end{aligned} \quad (3.32)$$

If the two lines to intersect, this system of linear equation will only have a solution for (b_1, b_2) . To determine the coordinate of the point, substitute the value of b_1 or b_2 in equation (3.30) or (3.31).

The second situation is where faces of the tool and the GAV boundary intersect. Faces are made of a number of vertices points that are connected together by edges. The face could be treated as plane segment that has a boundary made of vertices and edges. But contrary to the planes, a face does not intersect if the edges or vertices does intersect first. The face could be expressed as:

$$f_j = [p_1, p_2, \dots, p_n]^T \quad \forall j = 1 \dots m \quad (3.33)$$

Any two faces in 3D space could experience one of these three cases: a case where the faces are parallel to each other (Figure 3-14.a), a case where the faces are intersecting each other (Figure 3-14.b), and a case where the faces are on each other (Figure 3-14.c).

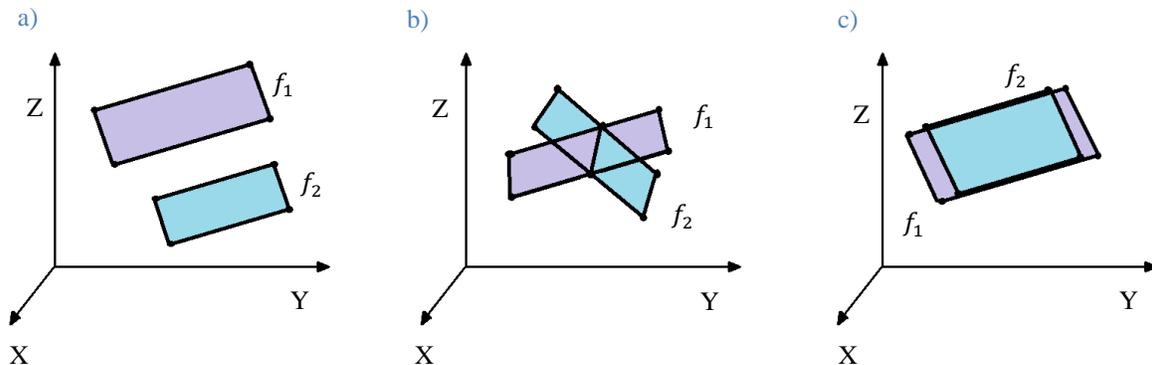


Figure 3-14: Faces Intersection for: a) No Solution, b) Linear Solution, c) Infinite Solutions

For the case where the faces are parallel to each other, the intersection has no solution. This indicates that there is no collision between the two faces.

$$f_1 \cap f_2 = \emptyset \quad (3.34)$$

For the case where the faces intersect each other, this intersection could have a point or a line as its intersection solution. This indicates that there is a collision between the two faces.

$$f_1 \cap f_2 \neq \emptyset \quad (3.35)$$

For the case where the faces are on top of each other, the segment of intersection would have infinitely many solutions similar to equation (3.35). This indicates that there is a collision between the two faces.

The tool is considered to have a collision only if the intersection of the edges, faces or vertices have a single or infinite solutions at any of the tool shape layout. In this case, the target point is considered inaccessible at the given tool orientation. The following subsection will present the solution for the tool orientation.

3.4.3. GAV with Variable Tool Orientation

Similar to the GAA, in order to present any object accurately in 3D space, the position as well as the orientation of that object shall be determined. Both the position of the object and the orientation in space shall be with respect to the global frame of reference. When associating a coordinate frame in a given configuration to a given object, the orientation of that object can be expressed in terms of the global frame of reference using a rotation matrix. This rotation matrix presents each of the axis of the object locally in terms of the global frame reference.

In 3D space, the rotation matrix \mathbf{R} can be determined using the three consecutive rotations called *Euler angles* (α, β, γ) , if the angles (α, β, γ) between the object local axis \mathbf{xyz} and the global frame reference \mathbf{XYZ} are known. Using *Euler angles*, the nested dimensions rotation matrix

that can be then used to calculate relative orientation of the object in terms of the global frame reference as shown in Figure 3-15.

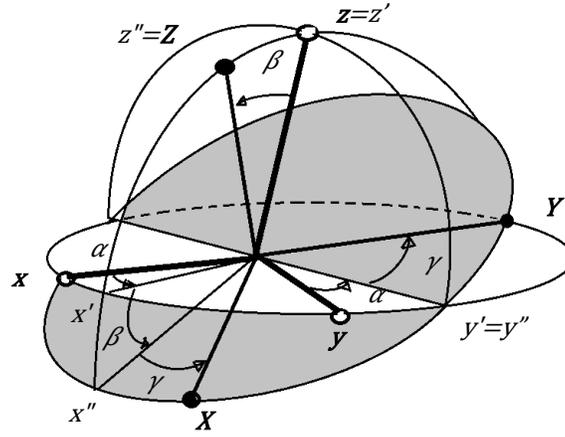


Figure 3-15: Euler Angles Representation

The following equations transform the set of *Euler angles* (α, β, γ) into a rotation matrix \mathbf{R} [44] & [45]:

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (3.36)$$

$$R_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \quad (3.37)$$

$$R_z(\gamma) = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.38)$$

$$\mathbf{R} = R_x(\alpha) \times R_y(\beta) \times R_z(\gamma) \quad (3.39)$$

This rotation matrix can be then used with the position of the object to determine its location in 3D space relative to the global coordinate system reference.

If the angles (α, β, γ) are not known, then the three points configuration approach can be used to determine the rotation matrix \mathbf{R} , as shown in Figure 3-16. This approach is used to find the

relative orientation of two independent coordinating systems relative to each other, like the workpiece P in relation to the robot tool G [46].

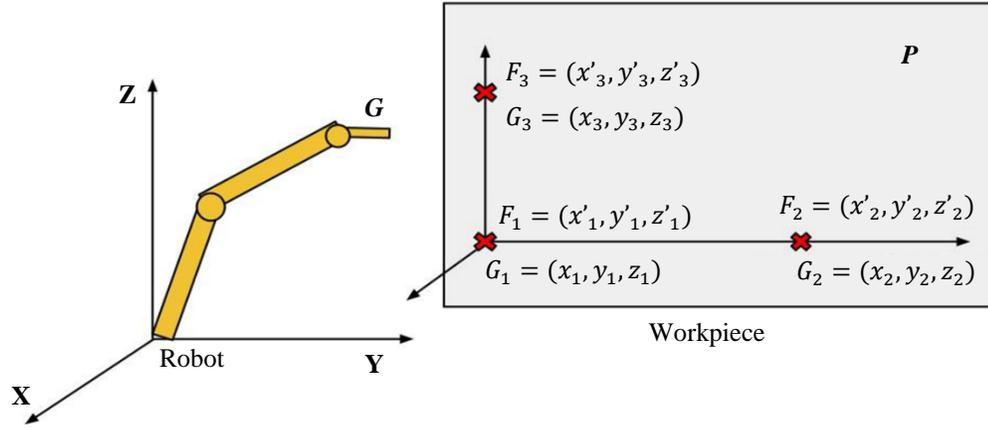


Figure 3-16: The Three Point Approach Diagram

In order to find the orientation relation between the robot frame and the workpiece frame, a set of three points on the workpiece P has to be determined. Two sets of values will be determined for each of the three point, where one set will be based on the robot coordinate system (G_1, G_2, G_3), and the other will be based on workpiece coordinate system (F_1, F_2, F_3). Then, two vectors will be calculated using each set of points obtained for the robot and the workpiece as the following:

$$\vec{V}_1 = G_2 - G_1 \quad (3.40)$$

$$\vec{V}'_1 = F_2 - F_1 \quad (3.41)$$

$$\vec{V}_2 = G_3 - G_1 \quad (3.42)$$

$$\vec{V}'_2 = F_3 - F_1 \quad (3.43)$$

These vectors were then transformed into a matrix format based on the following equations, where each vector forms one of the columns:

$$[M_R] = [\vec{V}_1 \quad \vec{V}_2 \quad \vec{V}_1 \times \vec{V}_2] \quad (3.44)$$

$$[M_W] = [\vec{V}'_1 \quad \vec{V}'_2 \quad \vec{V}'_1 \times \vec{V}'_2] \quad (3.45)$$

The resulting two matrixes were arranged as follows:

$$[M_R] = \mathbf{R} [M_W] \quad (3.46)$$

Equation (3.46) can be rearranged to determine the rotation matrix \mathbf{R} :

$$\mathbf{R} = [M_R][M_W]^{-1} \quad (3.47)$$

This rotation matrix represents the relation between the robot frame, which is the global coordinate system reference, and the local workpiece coordinate systems. The calculated rotation matrix \mathbf{R} can be then used with the position of the object to determine its configuration in 3D space relative to the global coordinate system reference.

Taking the rotation matrix \mathbf{R} in consideration, the position of the object can be expressed as:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \mathbf{R} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \mathbf{h} \quad (3.48)$$

where, \mathbf{h} is the translation vector, x , y and z are the local position coordinates, while x' , y' and z' are the position in global frame reference.

In terms of 3D homogeneous transformation matrix \mathbf{T} , the object position and orientation can be expressed as:

$$\mathbf{T}_i = \begin{bmatrix} \mathbf{R}_i & \mathbf{P}_i \\ 0 & 1 \end{bmatrix} \quad (3.49)$$

where, i is the i^{th} object, \mathbf{P} is the position point (x, y, z) .

Let the object be the tool, where the tool in the section 3.4.2 had a fixed orientation. Now, let this tool have a variable orientation at all the target points F where accessibility was determined to be inaccessible based on the approach used in section 3.4.2. As the tool changes its orientation around the axis perpendicular to the workpiece surface, the line intersection equations are

revalidated to determine if there is collision at all angles. The orientation is incremented by 10° interval. If the tool completes a 90° rotation and still experience linear intersection, then this point is inaccessible at all angles.

3.5. Accessibility Percentage

The accessibility percentage is a numerical representation of the ability of a given tool to reach most of the required targets in a given space successfully. Let F be the targets available on a given workpiece P . Let m be the total number of targets available on the workpiece P . Let n be the actual number of targets that are found to be inaccessible on the workpiece P . The accessibility percentage η_{acc} can be expressed as:

$$\eta_{acc} = \frac{m_F - n_F}{m_F} \times 100\% \quad (3.50)$$

If the workpiece has an accessibility percentage above 80% then the workpiece is considered accessible with minimal interference. If accessibility percentage is between 50% and 80%, then to complete the required task the tool have to be modified, and the accessibility analysis has to be conducted for another validation. If the accessibility percentage is below 50%, then the workpiece is in a crowded inaccessible area. That will require either a redesign of the tool used in the process, or a redesign of the workpiece.

CHAPTER 4: PATH PLANNING

Proper path planning is considered a major challenge for robotic riveting in confined spaces. This is due to the large number of limitations and constraints that the tool attached to a robotic arm faces. This chapter presents a detailed path planning analysis for robotic systems using a new pathfinder approach to optimize a collision-free path. The objective of the path planning is to determine a feasible pathfinder methodology based on the accessibility analysis, to generate the required path that the robot has to take. This optimal path should follow the governing standard equations required for drilling and riveting in the aerospace industry. In addition, this path should allow the tool to properly reach all the desired target points, and complete all the required tasks while avoiding all obstacles within the confined space.

In the first section, the standard equations required to generate the required pattern for the riveting process are presented and explained. This pattern will be used then as a template for the drilling and riveting tasks. The following section presents criteria required to select the optimal collision-free path. These criteria include the shortest path with the minimal distance traveled for better production efficiency, target approachability planning for a minimal jerk when considering the motion aspect, and variable tool orientation to handle surface curvature. The last section presents a new pathfinder methodology that is developed based on the GAA and GAV to determine all possible collision-free paths for the required drilling and riveting tasks.

4.1. Path Planning for Drilling and Riveting

In general, the path planning is a geometric problem that defines a geometric path between two points or more with no time factor, for the robot to perform a number of tasks while avoiding certain constraints. In this case, the required tasks are riveting and drilling, and the constraints are

the number of obstacles available around or on the workpiece that prevent the tool to reach directly the required target point. The process to determine the accessible boundaries was presented in the previous chapter. In this chapter, the process to determine the required riveting pattern and the steps to select the optimal path is presented.

It is important to follow the standard riveting patterns equations defined by the Federal Aviation Administration (FAA) when generating the required riveting pattern to generate the required path. These equations specify the required distance S from the riveted sheet or beam edge, the minimum and maximum pitch distance or distance between rivets B , the number of rows of rivets n , and the installation sequence, that will have a direct impact on the path planning process. Refer to Appendix A for the standard riveting equations and patterns.

4.2. Path Selection Process for Robotic Systems

The main goal of the robot path planning is to be able to define the optimal safe path for a robotic arm, to execute all the required tasks from start to finish while preventing any type of collision or near miss with any static obstacles in the workspace of the robot. This optimal path must be the one that allow the robot to execute the planned tasks in the fastest speed for the shortest production time, while preventing any harm to the mechanical robot. For this reason, a number of criterions were established to support the process of selecting the optimal path that will allow the robot to complete the required tasks for the shortest distance traveled, at a smooth approachability profile, and for the best tool orientation.

4.2.1. Shortest Travel Distance

There are multiple path approaches, which can be used by a robotic manipulator to perform drilling and riveting tasks. However, not all of them are efficient, fast, or use the least number of

steps, and the least direction change. The shortest distance traveled approach is considered one of the best approaches when it comes to robotic path planning. This approach is also known as the *Bellman-Ford* approach for the *shortest path problem*. In this approach, the target points, in this case is the riveting pattern, get ordered based on the distance between target and the distance to the home position of the tool. Then, the distances between each of these steps is added together for each target point starting from the pilot hole to the last hole in the riveting pattern. The path that has the least distance traveled is considered the optimal solution, which will ensure a faster execution time.

For instance, to rivet double row of four rivets each as shown in Figure 4-1, where the pilot hole is circled in green and the rivet holes are presented as red exes, there could be at least three possible approaches for the least number of steps: approach ‘a’, approach ‘b’, and approach ‘c’.

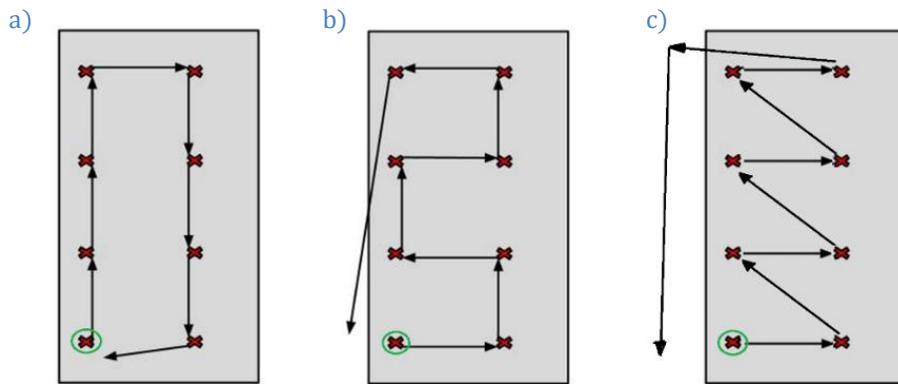


Figure 4-1: Shortest Path Problem: a) Linear Path, b) Chain Path, c) Zig-zag Path

However, the best solution is the one that use least number of movements and the shortest traveled distance s_{min} through these steps. This expression can be written as:

$$s_{min} = \sum_1^{n-1} d_{f_i} \quad (4.1)$$

where, s_{min} is the shortest path that can be obtained for all n targets, using the sum of the minimal distance d between all target points f_i . If s_{min} is valid for more than one path, the path with the least change in tool direction is considered the shortest. If s_{min} is still valid for more than one path, then the path with the shortest distance to reach the tool home position is considered the shortest.

Therefore, based on equation (4.1) and the criteria mentioned to isolate the shortest path, the optimal path for higher productivity that to be considered is the one of approach 'a'. This approach will save processing time, ensure a fast execution time, and reduce the robot work needed to perform the task as well as it will consume the least energy required by the robot. More, it will reduce the required steps to control the robot in order to reach the required targets, and as a result, this will reduce the computational power and the code needed.

4.2.2. Target Approachability Planning

Target approachability is the processes of determining the required number of intermediate target points that allow the tool to maneuver smoothly from one target point to reach the desired target point or to avoid an obstacle that could be located between the two targets. This criteria is a very important aspect when performing path planning as it facilitate the generation of the required intermediate target points that ensures a smooth motion of the robotic arm. The goal of these intermediate targets is to reduce the mechanical stress that the robot experience to move from one target to another one. It also can be used to plan the path to avoid any obstacle that could obstruct the motion of the robot, like the I-beam standing segment. More, these targets can be linked to a number of repetitive operational sequence like the drilling and riveting process routines.

It is very important to consider the way that the tool have to maneuver in order to move from one target to the following one. As there are a number of factors that the generated path have to account for, which include the obstacles, the safety zone around the obstacles as shown in

Figure 4-2, and the required routine for a process like drilling or riveting to perform a drilling or riveting tasks. A typical routine for drilling can be summarised as: approach the target point, drill trough, and pull back. The routine for riveting is to insert the rivet, extend the bucking bar, secure the rivet in the drilled hole, then rivet, and pull back. Thus, it is very important to determine these intermediate target points to generate the optimal path for the desired task.

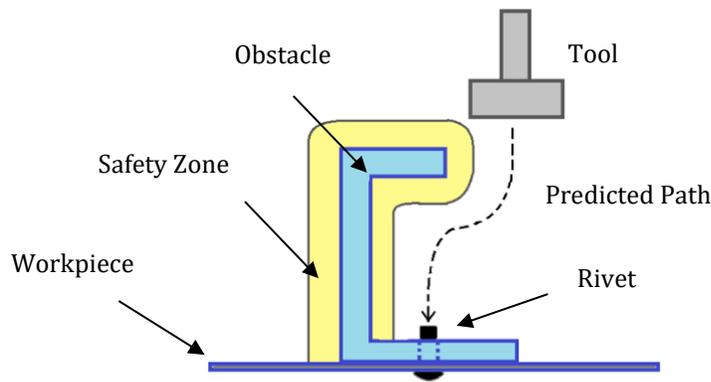


Figure 4-2: Predicted Path Maneuver for the Tool Around the Obstacle

For instance, to drill a double row of four holes each as shown in Figure 4-3, the drill gun has to reach the intermediate target point F_A before the drill runs. Then, progresses to reach the desired target point F_1 . Next, once the hole is drilled, the gun will pull back to the intermediate target point F_A , and more to the intermediate target F_B . These steps is then repeated at each target point until the task is completed.

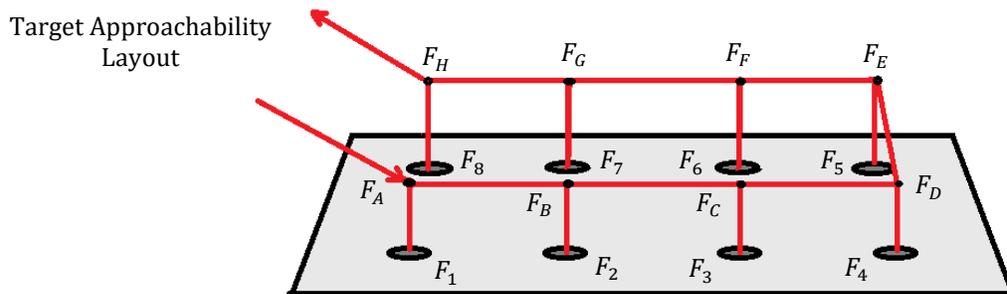


Figure 4-3: Basic Target Approachability Layout for Drilling and Riveting Pattern

These intermediate target points were classified into a numerical order based system to quantify the number of intermediate target points as shown in Figure 4-4. The zero order is the case where there are two intermediate points between each two targets. The first order is the condition where there are three intermediate target point between each two targets. The second order is the one where there are five intermediate points between each two targets. The third order is the one where there are seven intermediate points between each two targets, and so on.

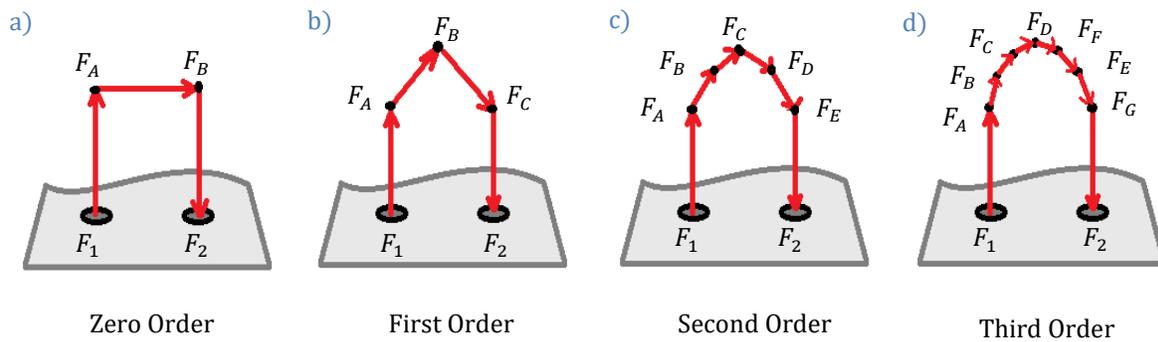


Figure 4-4: Target Approachability Planning Pattern Order

The number of intermediate target points required per order can be calculated using the following equation:

$$F(n) = \begin{cases} 2 & \forall n = 0 \\ 2n + 1 & \forall n \geq 1 \end{cases} \quad (4.2)$$

where, n is the order number.

In order to determine the position of these intermediate targets as shown in Figure 4-5, the distance between the targets B has to be found. More, the allowed maximum height H_{max} when clearing any obstacle between the target points must be determined. Then, the allowed clearance height H_{clear} that the tool such as the drill bit has to retract to allow the tool to be at a safe height above the target point must be found. By combining these measurements using trigonometry, the coordinates of the intermediate target points can be calculated.

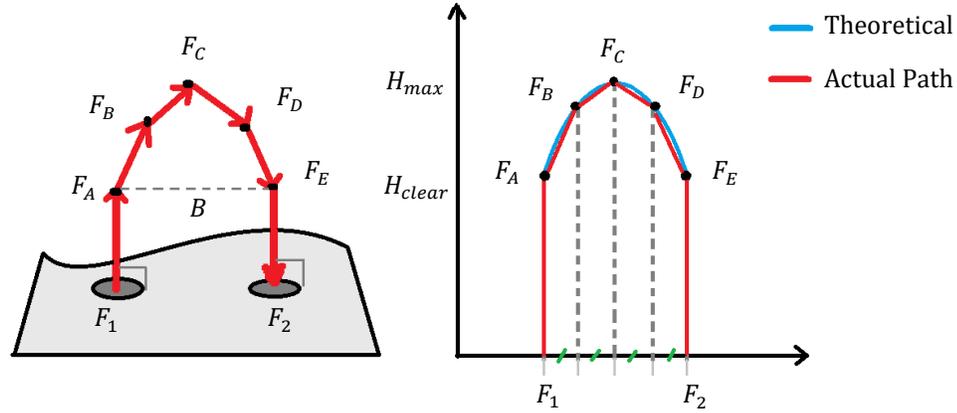


Figure 4-5: Target Approachability Planning Pattern for Higher Orders

The following equation is used to determine the distance between each intermediate target point:

$$F(n) = \begin{cases} B & \forall n = 0 \\ \frac{B}{2n} & \forall n \geq 1 \end{cases} \quad (4.3)$$

where, B is the distance between the rivet points, and n is the desirable approachability order. Once the distance between the intermediate points are determined, they can be used with the order number to generate the coordinates required for the intermediate target points to ensure a smooth transition motion between each target point and to avoid any obstruction. In addition, these target points can be used to perform the required steps for the drilling and riveting routines.

It is important to note that the higher the order the smoother the motion of the robot between these two targets. However, the higher the order the more time it takes to compute the optimal path. Thus, it is very important to balance the order required to reach the optimal path.

4.2.3. Variable Tool Orientation

It is important to consider the variation of the tool orientation in 3D space through the generated path when planning for robotic riveting on surfaces with curvatures, especially when the tool is the end-effector. If the workpiece surface is curved, the generated path shall consider all the required tool orientation correction to properly reach the desired target position. In order to do so, the workpiece surface curvature has to be determined then the tool orientation at each target point has to be updated. For a process like drilling or riveting, the tool must always be perpendicular to the target point and normal to the surface curvature at this target point.

There are two approaches that can be used in order to determine the curvature of the workpiece surface (Figure 4-6). The first approach is by using two points (which mark the arc start and finish boundaries) the radius of curvature, and the direction of the surface curvature. The second approach is using just three points that the arc of curvatures passes through them.

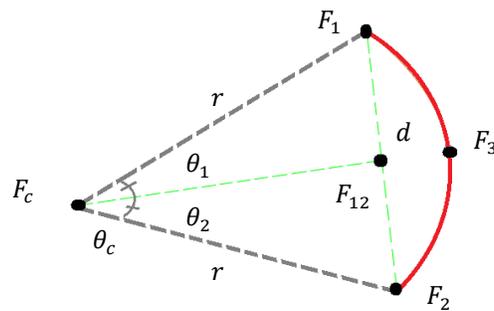


Figure 4-6: Curvature Presentation Diagram

A radius of curvature is defined as the radius of an imaginary circle that completes the missing portion of the arc segment. This radius r , with the unit vector representation of its direction $[r\hat{i} \ r\hat{j} \ r\hat{k}]$, can be used with two points F_1 and F_2 on the same plane of the arc to determine the arc equation. To calculate the center point location of this imaginary circle, the following equations are used:

- To calculate the midpoint between the two points F_1 and F_2

$$F_{12} = \frac{F_1 + F_2}{2} \quad (4.4)$$

- To calculate the distance between the two points F_1 and F_2

$$d_{12} = \sqrt{(x_{F_1} - x_{F_2})^2 + (y_{F_1} - y_{F_2})^2 + (z_{F_1} - z_{F_2})^2} \quad (4.5)$$

- To calculate the distance to the imaginary center point of the arc

$$d_c = \sqrt{r^2 - \left(\frac{d_{12}}{2}\right)^2} \quad (4.6)$$

- To calculate the arc angle

$$\theta_c = 2 \times \sin^{-1}\left(\frac{d_{12}}{2 \times r}\right) \quad (4.7)$$

- To calculate the center point coordinates of the imaginary circle

$$F_c = (d_c \times [r\hat{i} \quad r\hat{j} \quad r\hat{k}]) + F_{12} \quad (4.8)$$

- To calculate the coordinates of the midpoint between F_1 and F_2 on the arc

$$F_3 = F_c - (r \times [r\hat{i} \quad r\hat{j} \quad r\hat{k}]) \quad (4.9)$$

These three points F_1 , F_2 , and F_3 can be then used in the second approach to calculate the arc boundaries and equation.

The second approach uses three points on the arc to determine the equation of the curve. This approach can be used directly if the three points F_1 , F_2 , and F_3 are known and on the same plain of the arc. These points are ordered in such a way that F_1 , and F_2 are at the limits of the arc, and F_3 is at mid-way between F_1 , and F_2 on the arc. To calculate the center point location of this imaginary circle, the following equations are used:

- To calculate the vector representation of F_1 , F_2 and F_3

$$\overrightarrow{F_{13}} = F_1 - F_3 \quad (4.10)$$

$$\overrightarrow{F_{23}} = F_2 - F_3 \quad (4.11)$$

$$\overrightarrow{F_{12}} = F_1 - F_2 \quad (4.12)$$

$$\overrightarrow{w} = \overrightarrow{F_{23}} \times \overrightarrow{F_{13}} \quad (4.13)$$

- To calculate the center point coordinates of the imaginary circle

$$F_c = F_3 + \left(\frac{\|\overrightarrow{F_{23}}\|^2 \times (\overrightarrow{F_{13}} \cdot \overrightarrow{F_{12}}) \times \overrightarrow{F_{13}} - \|\overrightarrow{F_{13}}\|^2 \times (\overrightarrow{F_{23}} \cdot \overrightarrow{F_{12}}) \times \overrightarrow{F_{23}}}{2 \times \overrightarrow{w}} \right) \quad (4.14)$$

- To calculate the radius

$$r = \frac{1}{2} \times \sqrt{\frac{\|\overrightarrow{F_{13}}\|^2 \times \|\overrightarrow{F_{23}}\|^2 \times \|\overrightarrow{F_{12}}\|^2}{2 \times \|\overrightarrow{w}\|^2}} \quad (4.15)$$

- To calculate the vector limits

$$\overrightarrow{F_{1c}} = \frac{F_1 - F_c}{|F_1 - F_c|} \quad (4.16)$$

$$\overrightarrow{F_{2c}} = \frac{\overrightarrow{w} \times \overrightarrow{F_{1c}}}{|\overrightarrow{w} \times \overrightarrow{F_{1c}}|} \quad (4.17)$$

- To calculate the higher and lower angle limits

$$\theta_1 = \tan^{-1} \left((F_2 - F_c) \cdot \overrightarrow{F_{1c}} \right) \quad (4.18)$$

$$\theta_2 = \tan^{-1} \left((F_2 - F_c) \cdot \overrightarrow{F_{2c}} \right) \quad (4.19)$$

- The equation of the arc

$$F(\theta) = \left(r \times (\cos(\theta) \times \overrightarrow{F_{1c}} + \sin(\theta) \times \overrightarrow{F_{2c}}) \right) + F_c \quad \forall \theta_1 \leq \theta \leq \theta_2 \quad (4.20)$$

This arc equation will be used to determine the required orientation correction for the tool, and the correction required to the drilling and riveting pattern.

The transformation of the Cartesian based rotation matrix to quaternions is important when expressing the tool orientation as the ABB robot only accepts any orientation correction in the form of quaternions coordinates. The quaternions are a number system that comprise a four-dimensional vector space with a basis consisting of the real number 1 and three imaginary units i, j, k. This set of real and imaginary numbers is used by robotics to perform 3D rotations. In many robotic systems, the orientation can be described in the form of four quaternion parameters q_0 , q_1 , q_2 , and q_3 , where their sum must be normalised [47]:

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (4.21)$$

A quaternion is just a more appropriate way to describe the orientation of an object while avoiding the singularities experienced by the *Euler angles*. These quaternion parameters are calculated based on the elements of rotation matrix. If the elements of the rotation matrix are presented as the following:

$$[R] = \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{bmatrix} \quad (4.22)$$

The quaternion parameters can be found using the following equations:

- For the real parameter:

$$q_0 = \frac{\sqrt{x_1 + y_2 + z_3 + 1}}{2} \quad (4.23)$$

- For the imaginary parameters:

$$q_1 = \frac{\sqrt{x_1 - y_2 - z_3 + 1}}{2} \quad (4.24)$$

$$\text{Sign } q_1 = \text{sign}(y_3 - z_2) \quad (4.25)$$

$$q_2 = \frac{\sqrt{y_2 - x_1 - z_3 + 1}}{2} \quad (4.26)$$

$$\text{Sign } q_2 = \text{sign}(z_1 - x_3) \quad (4.27)$$

$$q_3 = \frac{\sqrt{z_3 - x_1 - y_2 + 1}}{2} \quad (4.28)$$

$$\text{Sign } q_3 = \text{sign}(x_2 - y_1) \quad (4.29)$$

Note that the sign has to be verified independently as presented in equations (4.25), (4.27), and (4.29), where the sign resulting from the calculation of equations (4.24), (4.26), and (4.28) does not provide the correct information about the parameter orientation. To verify if the quaternions coordinate are valid, the transformation from quaternions to Euler angles can be expressed as the following [47]:

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} \text{atan2}(2 \times (q_0 q_1 + q_2 q_3), 1 - 2 \times (q_1^2 + q_2^2)) \\ \arcsin(2 \times (q_0 q_2 - q_3 q_1)) \\ \text{atan2}(2 \times (q_0 q_3 + q_1 q_2), 1 - 2 \times (q_2^2 + q_3^2)) \end{bmatrix} \quad (4.30)$$

The Euler angles will match all the initial angles, only if the quaternions conversion was conducted correctly.

Knowing both the curvature calculations and the quaternions transformations facilitates the required calculations to accommodate the workpiece surface curvatures when generating the optimal path planning. These corrections can then be sent to the robot to enable the tool to reach all the desired target locations with the correct tool orientation.

4.3. Path Planning Algorithm for Collision-Free Path

It is important to represent the path planning process in an algorithm format as robotic drilling and riveting involves many subroutines tasks. These subroutines involve number of subtasks such as rivet insertion, lock in location, then the riveting action. More, a process like riveting has to be executed in such a way that prevents the workpiece from stretching or buckling. For these reasons, it is important to consider all possible implicit and explicit constraints related to accessibility validation, as it was mention in the previous chapter. The implicit constraints are derived from collision avoidance between the robot and obstacles, and explicit ones include the mechanical consideration of the robot, which is the limitations of joint motions. The mechanical limitations of the ABB IRB 4400 45kg industrial robot is presented in Appendix D, (Table D-1 and Table D-2) and Appendix E. It is recommended to place the workpiece within the work range boundaries specified by the manufacturer in such a way that allow the robot to reach all the desired target locations as required.

In order to facilitate the process of path planning, computer aid simulators such as RobotStudio™ provide the foundation to develop applications that could be used to generate all possible paths and determine the optimal path for the robotic manipulators. This ABB control package software also has features to visually notify if the tool is closer to the obstacle by changing the color of the object, which is useful to validate the developed software. In RobotStudio™ software, all the components of the work environment, including workpieces and possible obstacles, have to be loaded. Then, each component has to be positioned and set in the correct orientation in the virtual environment that matches the actual position an orientation of these object in reality. The position and orientation parameters of these objects in the virtual environment can

then be used as inputs for the developed software application to generate the required code scripts that command the simulation of the robotic motion as desired.

In this section, the path planning process algorithm is presented. This algorithm is one of the Riveting Planner[®] add-in software application foundations, that is used to generate all possible paths and select the optimal path for the desired tasks. The flowchart diagram in Figure 4-7 depicts the path planning algorithm that was developed to simplify the optimization process.

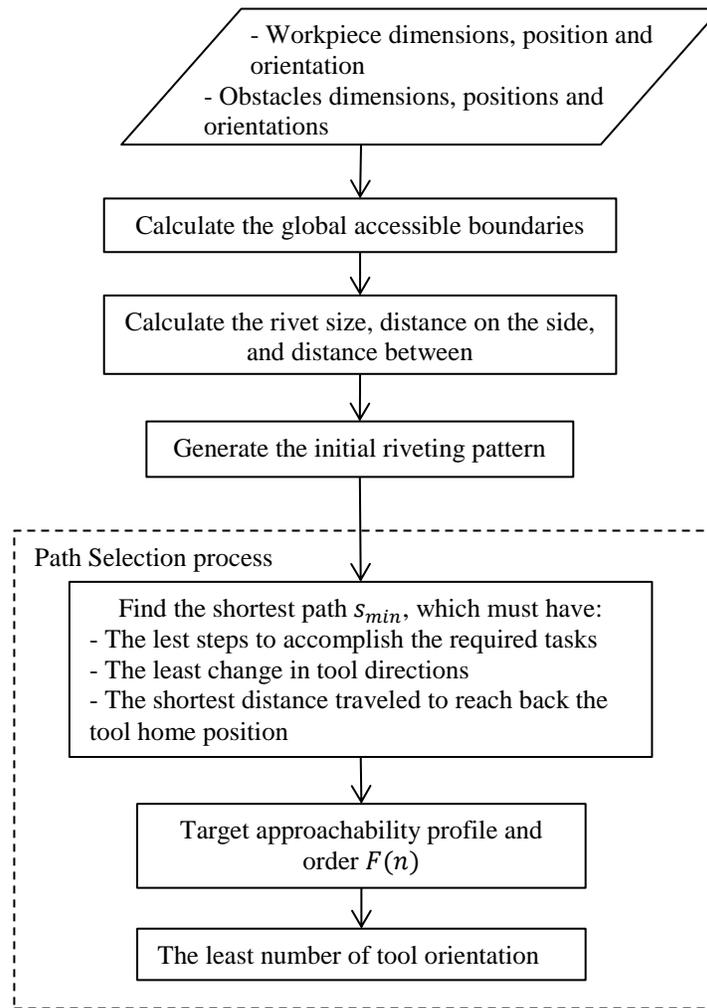


Figure 4-7: Flowchart Diagram of the Path Selection and Optimization Process

The path planning algorithm uses the parameters values of the dimension as well as the position and orientation of the workpiece and all obstacles in the configuration space. These

parameters values are then used to calculate the global accessibility boundary (GAV for the 3D case). Then, the dimension of the workpiece and obstacles are used to calculate the required number of rivets, the rivet size, the distance between rivets, distance on the side of the rivets and number of rows. Then, based on the GAV boundaries and the rivet calculations, the initial riveting pattern is generated. Once the pattern is complete, the path selection process starts. The first aspect is to find the shortest path. Then, filter the results down based on the least steps to accomplish the required tasks, the least change in tool directions, and the shortest distance traveled to reach back the tool home position. Once the shortest path is found, the process of adding the target approachability profile starts. This determines the number of intermediate target points required to complete a drilling or riveting routine. Once completed, the tool orientation validation is conducted to determine all the required tool orientation modification at each target point. This validation check can be presented as depicted in the flowchart of Figure 4-8 below.

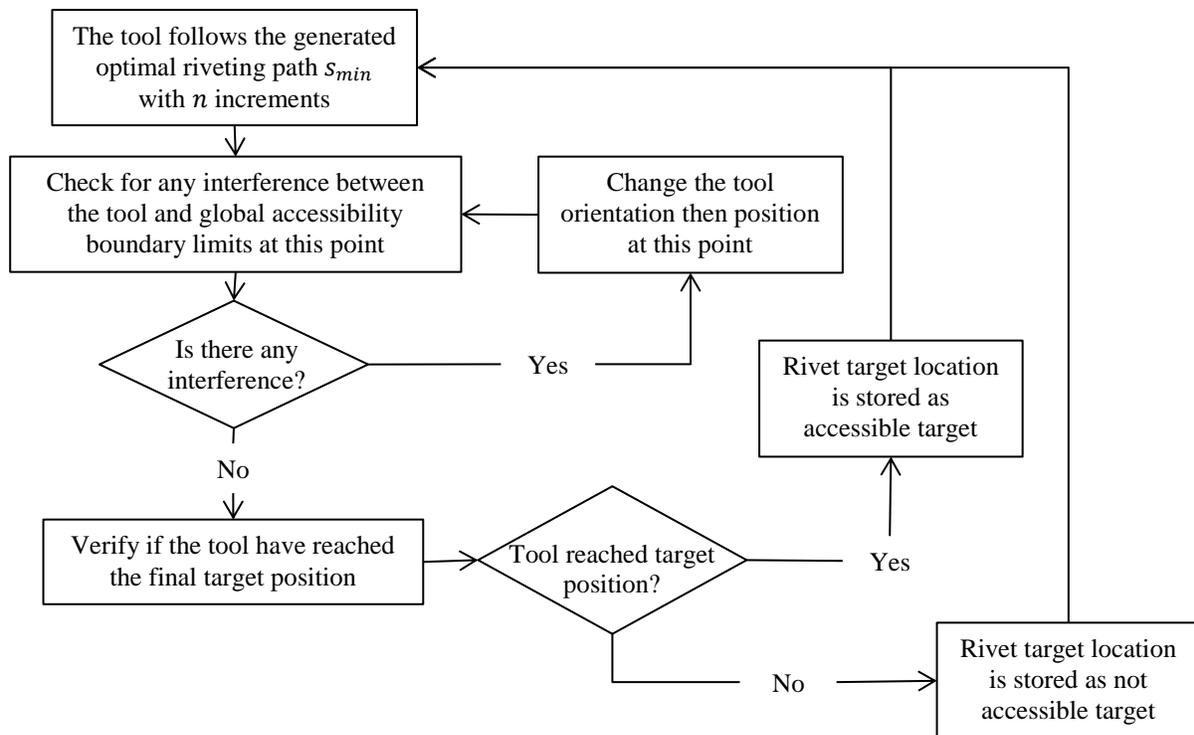


Figure 4-8: Flowchart Representation of the Accessibility Validation along the Optimal Path

In this validation check, the tool follow the optimal path and then check if there is any interference with the global accessibility boundary limits. If there is an interference then the tool orientation followed by the tool position will be modified. If there is no interference, the tool will verify if it reached its final target position. If it reached that target point, then this target portion is valid, and the target point as well as all the intermediate target points are stored. If it did not reach that target point, then this target point is no accessible and all the associated intermediate target points to reach this target is ignored, and the target point is stored as inaccessible target point. Once all target points are validated, the final optimal path is stored.

CHAPTER 5: MOTION ANALYSIS

Motion planning is the link that joins the generated path with the production rate for mass production. Understanding the motion planning process and how to calculate the cycle time of the entire task, is the key of fast assembly time at a minimum energy level. This chapter presents a detailed motion analysis based on the robot displacement, velocity and acceleration during the execution of the planned path. The objective of the motion analysis is to determine the velocity and acceleration required to complete the planned path without exceeding the kinematic constraints set by the robotic manufacturer. Further, it will determine the cycle time required to complete the planned path. This will then validate the pathfinder methodology and ensure that the optimal path selected is the best for mass production.

In the first section, review the motion curves for displacement, velocity, and acceleration. These will define the velocity and acceleration profiles required by the robot at each step of the optimal path, and validate that they are within the robotic mechanical control constraints. The second section will cover the robotic cycle time analysis. This is an important aspect when planning the productivity rate in mass production.

5.1. Motion Curves

Robots are programmable machines that can be used to accomplish a multitude of different tasks with high accuracy and great precision, which improve the production quality, works space safety, and manufacturing reliability. To operate complex systems like robotic manipulators, a set of code steps known as targets has to be generated to control the manipulator's path and motion to perform the required task. These target points carry not only the position of the required task and the required tool orientation at this target point, but also it carries the velocity and acceleration

required to reach that point or to pass through that target point. The goal of the manipulator motion planning is to break down a high-level task into low level of actuator commands that can ensure that the task is properly executed without causing any damage to the workpiece or the manipulator mechanical systems. Therefore, understanding the motion planning aspect of the tool accessibility, is very important as it allows a proper planning of the required drilling and riveting tasks, and it enables the possibility of quantify the process duration for mass production.

Motion planning is the study of the dynamic aspect of the planned path, in order to validate if the robot would be capable of required tasks within the minimal execution time without harming the mechanical and control system components of the robot. Combining the time information with the geometrical path while considering the kinematic and dynamic constraints of the robot, provide a better view of the actual task to accomplish. The manipulator motion can be expressed in a form of profiles for the robot displacement, velocity, acceleration and jerk. These generated profiles can be used to analyse the stress that the mechanical systems of the robot experience throughout the planned path. The ultimate goal of the motion planning is to generate smooth profiles that ensure a continuous acceleration. This continues acceleration will ensure that the robot experience the least jerk amount.

As previously mentioned in chapter 2, there are two types of motion profile representations that are used to conduct motion planning, the trapezoidal curve and the high order polynomial curve (S-curve). The simplest format is the trapezoidal curve representation; however, this approach generate high jerk profiles, which harm the robot mechanical systems. Therefore, the S-curve approach will be followed for the motion analysis. The S-curve approach takes the higher order polynomial curvature representation to ensure that the acceleration has at least third order polynomial representation. This ensures that the resulting motion will have minimal jerk profile.

In order to study the motion through the optimal path, the path has to be sliced down based on the target points. The distance between any two target points in sequence is used as the foundation of the motion analysis. This distance could be between the main target point and their intermediate targets, which regulate the motion profile between the main target points.

Let the path s between the two target points F_1 and F_2 be of the zero order of approachability as shown in Figure 5-1. This zero order will generate two intermediate targets F_A and F_B . Let B be the distance between the F_1 and F_2 , and H be the distance between F_1 and F_A .

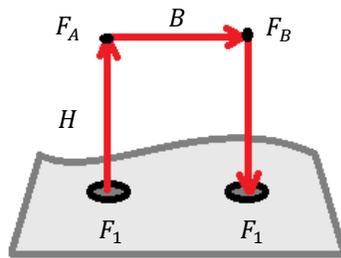


Figure 5-1: Generated Path Representation of Zero Order Approachability

The tool will move from F_1 until it reaches point F_A , and then will pause at point F_A . Next, the tool will move from F_A until it reaches point F_B , and then will pause at point F_B . Next, the tool will move from F_B until it reaches point F_2 , and then will pause at point F_2 . The motion profile for this path will be similar as the tool move between each target point.

There are four factors to consider when planning the motion of the robot between two target points. The first element is displacement. Displacement \mathcal{D} is the vector representation of the distance measured in a straight line and in a specified direction. The displacement between target point F_1 and the intermediate target point F_A , can be presented as a fifth polynomial curve as shown in Figure 5-2, where H is the distance between F_1 and F_A .

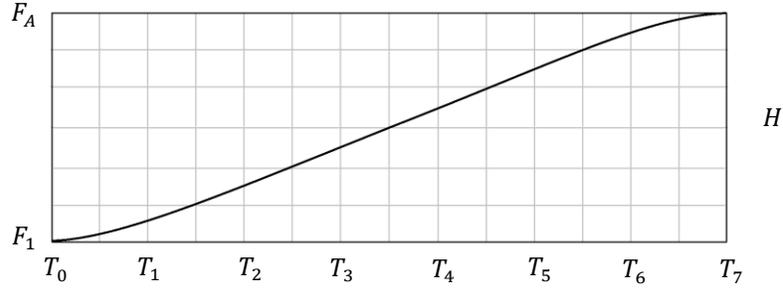


Figure 5-2: Graphical Representation of Displacement

In this graph, the displacement of the tool change gradually as it moves from F_1 at T_0 and moves gradually until it reaches F_A at T_7 . Note that the fifth order polynomial equation is required as there are six constraints, which are the initial and final displacement, initial and final velocity, and initial and final acceleration.

Let \mathcal{D} be a function of normalized time \mathcal{T} :

$$\mathcal{T} = \frac{t}{T} \quad (5.1)$$

where, t is the instantaneous time, and T is the total time.

The displacement can be then expressed as:

$$\mathcal{D}(\mathcal{T}) = p_5\mathcal{T}^5 + p_4\mathcal{T}^4 + p_3\mathcal{T}^3 + p_2\mathcal{T}^2 + p_1\mathcal{T} + p_0 \quad (5.2)$$

Now, consider the segment between T_0 and T_3 . Let the time between T_0 and T_3 be normalized, where $T_3 = T$, as the segment T_5 to T_7 is its mirror, and the segment T_3 to T_5 is steady motion. The boundary conditions for the displacement when $T_0 = 0$ is $\mathcal{D}(0) = s_0$, and when $T_3 = 1$ is $\mathcal{D}(1) = s_0 + d$; where, s_0 is the initial displacement and d is the magnitude of displacement at T_3 . However, when $\mathcal{T} = 0$ equation (5.2) becomes $\mathcal{D}(0) = p_0$, where the coefficient p_0 becomes:

$$p_0 = s_0 \quad (5.3)$$

and when $\mathcal{T} = 1$ equation (5.2) becomes:

$$\mathcal{D}(1) = p_5 + p_4 + p_3 + p_2 + p_1 + p_0 = s_0 + d \quad (5.4)$$

The second element to define the motion between two target points is the velocity. The velocity is the rate of change in displacement that the robot make to move from target point F_1 to F_A along the task path with respect to time. The velocity in its basic form can be expressed as:

$$v = \frac{ds}{dT} \quad (5.5)$$

where, s is the segment of the path between F_1 and F_A target points, and T is the time that the robot take to move from F_1 to F_A . The velocity between target point F_1 and the intermediate target point F_A can be presented as shown in Figure 5-3.

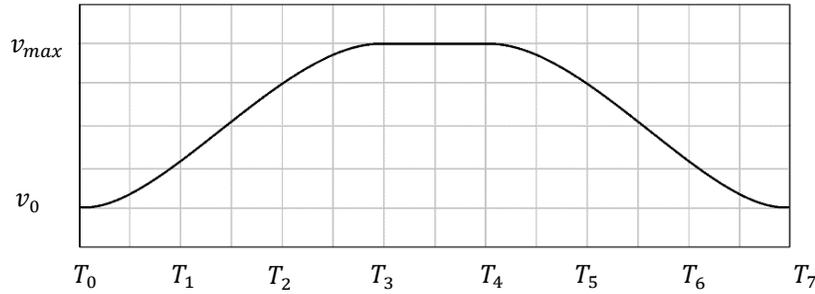


Figure 5-3: Graphical Representation of Velocity

In this graph, the velocity of the tool increase from complete stop at F_1 and T_0 until it reaches the maximum velocity at T_3 . Then, the tool moves with a steady velocity from T_3 to T_4 . Last, the speed of the tool gradually decrease from T_4 until it reaches a complete stop at T_7 when it reaches F_A .

The velocity can be then expressed as:

$$v(\mathcal{J}) = (5p_5\mathcal{J}^4 + 4p_4\mathcal{J}^3 + 3p_3\mathcal{J}^2 + 2p_2\mathcal{J} + p_1)/T \quad (5.6)$$

Similar to the displacement, consider the segment between T_0 and T_3 . The boundary conditions for the velocity when $T_0 = 0$ is $v(0) = v_0$, and when $T_3 = 1$ is $v(1) = v_f$; where, v_0 is the initial

velocity at T_0 , and v_f is the final velocity at T_3 . However, when $\mathcal{J} = 0$, equation (5.6) becomes $v(0) = p_1/T$, where the coefficient p_1 becomes:

$$p_1 = v_0 T \quad (5.7)$$

and when $\mathcal{J} = 1$ equation (5.6) becomes:

$$v(1) = 5p_5 + 4p_4 + 3p_3 + 2p_2 + p_1 = v_f T \quad (5.8)$$

The third element is acceleration, which is the rate of change in velocity that the robot make to move from target point F_1 to F_A along the task path with respect to time. The acceleration in its basic form can be expressed as:

$$a = \frac{dv}{dT} = \frac{d^2s}{dT^2} \quad (5.9)$$

where, s is the segment of the path between F_1 and F_A target points, and T is the time that the robot take to move from F_1 to F_A . The acceleration between target point F_1 and the intermediate target point F_A can be presented as shown in Figure 5-4.

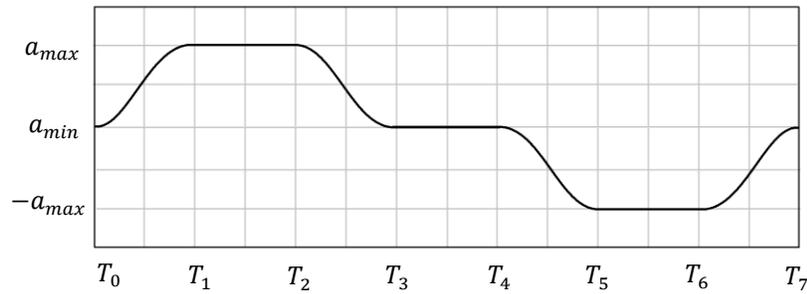


Figure 5-4: Graphical Representation of Acceleration

In this graph, the acceleration of the tool increase starting from F_1 at T_0 until it reaches the maximum acceleration at T_1 . Then, the tool moves with a constant acceleration from T_1 to T_2 . Next, the tool started to decelerate from T_2 to T_3 . Then, the tool moved between T_3 and T_4 with a

constant velocity that resulted into zero acceleration. Next, the tool started to decelerate between T_4 and T_5 . Then, the tool moves with a constant deceleration from T_5 to T_6 . Last, the tool gradually accelerates from T_6 until it reaches a complete stop at T_7 when it reaches F_A .

The acceleration can then be expressed as:

$$a(\mathcal{J}) = (20p_5\mathcal{J}^3 + 12p_4\mathcal{J}^2 + 6p_3\mathcal{J} + 2p_2)/T^2 \quad (5.10)$$

Similar to the displacement and velocity, consider the segment between T_0 and T_3 . The boundary conditions for the acceleration when $T_0 = 0$ is $a(0) = a_0$, and when $T_3 = 1$ is $a(1) = a_f$; where, a_0 is the initial acceleration at T_0 , and a_f is the final acceleration at T_3 . However, when $\mathcal{J} = 0$, equation (5.10) becomes $a(0) = p_2/T^2$, where the coefficient p_1 becomes:

$$p_2 = 0 \quad (5.11)$$

and when $\mathcal{J} = 1$ equation (5.10) becomes:

$$a(1) = 20p_5 + 12p_4 + 6p_3 + 2p_2 = 0 \quad (5.12)$$

Now, in order to find the coefficients for p_3 , p_4 , and p_5 , lets substitute the values for p_0 , p_1 , and p_2 of equations (5.3), (5.7), and (5.11) in equations (5.4), (5.8), and (5.12). The result will be:

$$\mathfrak{D}(1) \Rightarrow p_5 + p_4 + p_3 = d - v_0T \quad (5.13)$$

$$v(1) \Rightarrow 5p_5 + 4p_4 + 3p_3 = v_fT - v_0T \quad (5.14)$$

$$a(1) \Rightarrow 10p_5 + 6p_4 + 3p_3 = 0 \quad (5.15)$$

These equations could be rearranged to be in a matrix format:

$$\begin{bmatrix} 1 & 1 & 1 \\ 3 & 4 & 5 \\ 3 & 6 & 10 \end{bmatrix} \begin{bmatrix} p_3 \\ p_4 \\ p_5 \end{bmatrix} = \begin{bmatrix} d - v_0T \\ v_fT - v_0T \\ 0 \end{bmatrix} \quad (5.16)$$

Rearrange equation (5.14) into augmented matrix and solve using Gaussian elimination,

$$\left[\begin{array}{ccc|c} 1 & 1 & 1 & d - v_0T \\ 3 & 4 & 5 & v_fT - v_0T \\ 3 & 6 & 10 & 0 \end{array} \right] \Rightarrow \left[\begin{array}{ccc|c} 1 & 1 & 1 & d - v_0T \\ 0 & 1 & 2 & 2v_0T + v_fT - 3d \\ 0 & 3 & 7 & 3v_0T - 3d \end{array} \right] \Rightarrow \left[\begin{array}{ccc|c} 1 & 1 & 1 & d - v_0T \\ 0 & 1 & 2 & 2v_0T + v_fT - 3d \\ 0 & 0 & 1 & 6d - 3v_0T - 3v_fT \end{array} \right] \quad (5.17)$$

Now the remaining coefficients can be found:

$$p_5 = 6d - 3v_0T - 3v_fT \quad (5.18)$$

$$p_4 = 2v_0T + v_fT - 3d - 2(6d - 3v_0T - 3v_fT) = 8v_0T + 7v_fT - 15d \quad (5.19)$$

$$\begin{aligned} p_3 &= d - v_0T - (8v_0T + 7v_fT - 15d) - (6d - 3v_0T - 3v_fT) \\ &= 10d - 6v_0T - 4v_fT \end{aligned} \quad (5.20)$$

The equations for displacement (5.2), velocity (5.6), and acceleration (5.10), using the confidents p_0 , p_1 , p_2 , p_3 , p_4 , and p_5 , can be presented as:

$$\begin{aligned} \mathfrak{D}(\mathcal{J}) &= (6d - 3v_0T - 3v_fT)\mathcal{J}^5 + (8v_0T + 7v_fT - 15d)\mathcal{J}^4 \\ &\quad + (10d - 6v_0T - 4v_fT)\mathcal{J}^3 + v_0T\mathcal{J} + s_0 \end{aligned} \quad (5.21)$$

$$\begin{aligned} v(\mathcal{J}) &= 5\left(\frac{6d}{T} - 3v_0 - 3v_f\right)\mathcal{J}^4 + 4\left(8v_0 + 7v_f - \frac{15d}{T}\right)\mathcal{J}^3 \\ &\quad + 3\left(\frac{10d}{T} - 6v_0 - 4v_f\right)\mathcal{J}^2 + v_0 \end{aligned} \quad (5.22)$$

$$\begin{aligned} a(\mathcal{J}) &= \frac{20}{T}\left(\frac{6d}{T} - 3v_0 - 3v_f\right)\mathcal{J}^3 + \frac{12}{T}\left(8v_0 + 7v_f - \frac{15d}{T}\right)\mathcal{J}^2 \\ &\quad + \frac{3}{T}\left(\frac{10d}{T} - 6v_0 - 4v_f\right)\mathcal{J} \end{aligned} \quad (5.23)$$

The fourth element that is used to express motion is jerk, which is the rate of change of acceleration that the robot makes to move from target point F_1 to F_2 along the task path with respect to time. The jerk in its basic form can be expressed as:

$$J = \frac{da}{dT} = \frac{d^2v}{dT^2} = \frac{d^3s}{dT^3} \quad (5.24)$$

where, s is the segment of the path between F_1 and F_A target points, and T is the time that the robot takes to move from F_1 to F_A . The jerk between target point F_1 and the intermediate target point F_A can be presented as shown in Figure 5-5.

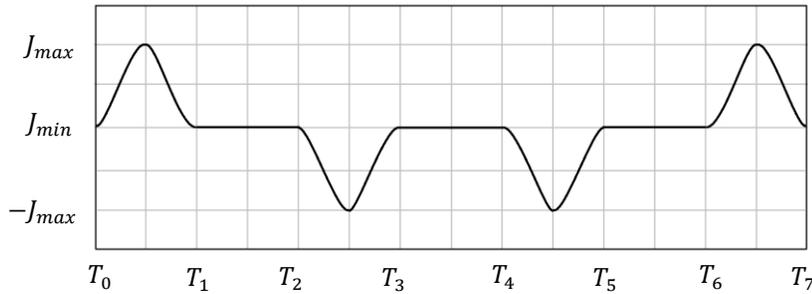


Figure 5-5: Graphical Representation of Jerk

In this graph, the jerk of the tool increases starting from F_1 at T_0 until it reaches the maximum jerk at $T_{0.5}$. Then, the jerk decreases to reach zero at T_1 . Next, the tool starts to experience negative jerk at T_2 and it reaches the maximum negative jerk at $T_{2.5}$. Then, the jerk decreases to reach zero at T_3 . Next, the tool starts to experience positive jerk at T_4 and it reaches the maximum positive jerk at $T_{4.5}$. Then, the jerk increases to reach zero at T_5 . Next, the tool starts to experience negative jerk at T_6 and it reaches the maximum negative jerk at $T_{6.5}$. Then, the jerk decreases back to reach zero at T_7 when it reaches F_A .

The jerk can then be expressed as:

$$J(T) = (60p_5T^2 + 24p_4T + 6p_3)/T^3 \quad (5.25)$$

Using the confidents p_3 , p_4 , and p_5 , the jerk can be presented as:

$$J(\mathcal{T}) = \frac{60}{T^2} \left(\frac{6d}{T} - 3v_0 - 3v_f \right) \mathcal{T}^2 + \frac{24}{T^2} \left(8v_0 + 7v_f - \frac{15d}{T} \right) \mathcal{T} + \frac{6}{T^2} \left(\frac{10d}{T} - 6v_0 - 4v_f \right) \quad (5.26)$$

Equations (5.21), (5.22), (5.23), and (5.26) are then used calculate the motion profiles based on the s-curve approach to ensure the minimal jerk effect throughout the motion of the robot.

To find the total time T of a given increment between the T_0 to T_3 segment using the mid-way through movement at $\mathcal{T} = 0.5$, it is found that the velocity can be presented as $\frac{(v_0+v_f)}{2}$.

Substituting this value back into equation (5.22) for $\mathcal{T} = 0.5$, will become:

$$v(0.5) = 5 \left(\frac{6d}{T} - 3v_0 - 3v_f \right) (0.5)^4 + 4 \left(8v_0 + 7v_f - \frac{15d}{T} \right) (0.5)^3 + 3 \left(\frac{10d}{T} - 6v_0 - 4v_f \right) (0.5)^2 + v_0 = \frac{v_0 + v_f}{2} \quad (5.27)$$

Simplifying equation (5.27):

$$v_0T + v_fT = \left(\frac{15}{4}d - \frac{15}{8}v_0T - \frac{15}{8}v_fT \right) + (8v_0T + 7v_fT - 15d) + (15d - 9v_0T - 6v_fT) + 2v_0 \quad (5.28)$$

Rearrange equation (5.28):

$$\frac{15}{4}d + \frac{15}{8}T(-v_0 - v_f) = 0 \quad (5.29)$$

Rearrange equation (5.29) for T :

$$T = \frac{2d}{v_0 + v_f} \quad (5.30)$$

For the cases where $v_0 = 0$ and $v_f = 0$, like in the T_0 to T_7 segment, equation (5.30) cannot be used as it will result into a constant velocity of zero. Instead, equations (5.21), (5.22), and (5.23) are used, with $v_0 = 0$ and $v_f = 0$. This will result the following equations:

$$\mathfrak{D}(\mathcal{T}) = d(6\mathcal{T}^5 - 15\mathcal{T}^4 + 10\mathcal{T}^3) \quad (5.31)$$

$$v(\mathcal{T}) = \frac{d}{T} (30\mathcal{T}^4 - 60\mathcal{T}^3 + 30\mathcal{T}^2) \quad (5.32)$$

$$a(\mathcal{T}) = \frac{d}{T^2} (120\mathcal{T}^3 - 180\mathcal{T}^2 + 60\mathcal{T}) \quad (5.33)$$

Similarly, using the mid-way through movement at $\mathcal{T} = 0.5$ in equation (5.32), to find the v_{max} , the resulting equation will be:

$$v_{max} = \frac{d}{T} \left(\frac{15}{8} \right) \quad (5.34)$$

Rearrange equation for T :

$$T = \frac{15 d}{8 v_{max}} \quad (5.35)$$

The ultimate goal of motion panning is to always ensure that the task has minimum execution time, the robot experiences minimum effort at the actuator joints, and the acceleration profile is as smooth as possible for minimum jerk. The robot motion has to be below the manufacturer specified maximum velocity $v_{R_{max}}$ and acceleration $a_{R_{max}}$ in order to realize these goals.

$$v_{max} \leq v_{R_{max}} \quad (5.36)$$

It was found that the manufacturer specifications for the ABB IRB 4400 45kg stated that the maximum velocity of robot when moving with all axis in motion on ISO test plane was $v_{R_{max}} = 2.2$ (m/s), and the maximum acceleration was $a_{R_{max}} = 14$ (m/s²) [48]. Thus to ensure a safe

operation of the robot the velocity and acceleration shall never exceed 2.2 (m/s) and 14 (m/s^2) respectively.

For cases with higher approachability order (above zero order), if the robot does not pause at each intermediate target point, the resulting motion profile for the s-curve will be much more complex and requires a higher level of computation. This is because the motion profile will change direction while moving with a steady velocity throughout the planned path. As the number of the intermediate targets increase, if the robot does not pause at each intermediate target, will result into a much smoother motion of the robot as it moves from F_1 to reach F_2 . The smoother the motion between targets, the less jerk can the robot experience, which causes less vibration and waste of energy.

5.2. Robotic Cycle Time Analysis

Cycle time is an important factor to consider when dealing with robotic automation for mass production. Cycle time is the average time required to complete a given cycle of tasks from start to finish. The ability to quantify the time that the robot takes to complete all required tasks per workpiece is essential when planning for a production line, to know the number of workpiece completed per hour. When compared to production rate, cycle time is inversely proportional with production rate, where the lower the cycle time the higher the production rate.

There are two levels of cycle time, the task cycle time and the total cycle time as shown in Figure 5-6. The task cycle time is the average time required to complete a given task like a riveting routine for n number of targets. While, the total cycle time is the average total time that the robot take to complete a cycle starting from the home position F_{home} , doing through the n number of task routines then back to the home position F_{home} .

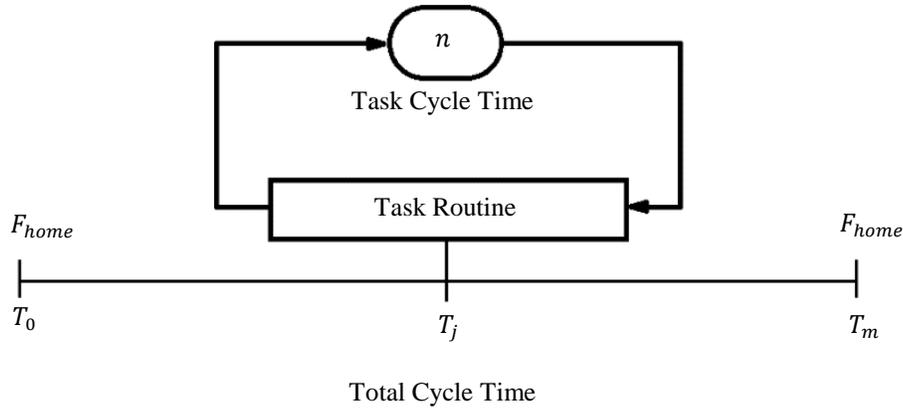


Figure 5-6: Total Cycle Time and Task Cycle Time Representation

This total cycle time T_{Total} can be expressed as:

$$T_{Total} = (T_1 - T_0) + n(T_{cycle}) + (T_m - T_{m-1}) \quad (5.37)$$

where, n is the number of repetition of the task cycle, and m is the total time at the home position.

Simulation software are the most effective tool when it comes to managing and validating the cycle time optimization as they can perform effectively the entire cycle and can provide an accurate representation of the time that the cycle takes. These software slice down the total cycle time into a number of segments as depicted in Figure 5-7. The first segment s_A is the time that the robot takes to move from the home position to reach the pilot point target on the workpiece. This pilot target point is the first target to achieve and it is the reference point to the robot. Then, the following segment s_B is the routine task segment, where the task cycle time is calculated and multiplied by the number of targets available. The last segment s_C is the time that the robot takes to move from the last target point back to the home position.

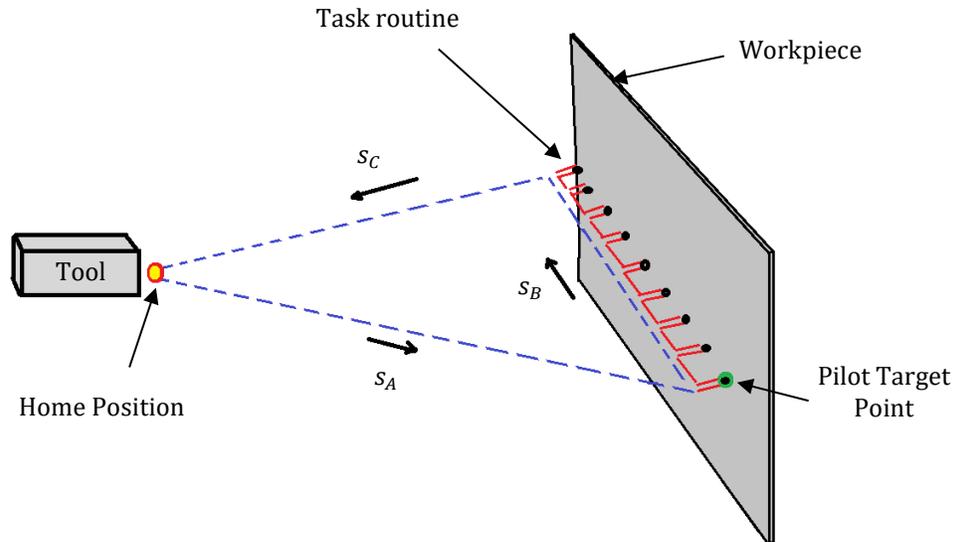


Figure 5-7: Path Process Segmentation for Motion Planning

If there is more than one task cycle, for instance riveting a number of spars, each task cycle will have two segments added. One segment to track the time that the robot take to move from the last target point of the last segment to reach the new pilot target point of the new segment. The other is the time that the robot takes through that new task cycle segment.

When considering robotic drilling and riveting, the task cycle time can be defined as time required to drill one hole or to place one rivet per target position. While the total cycle time is the time required to execute the entire path like the drilling path from home position, going through each cycle task, and returning back to the home position. The flowchart in Figure 5-8 depicts a typical drilling routine process starts by reaching the desired target point, moving forward until the drill bit touches the drilling location, and the stabilization harness ensure that the drill is locked in position. Then, the drill starts operating until the hole is complete, then the drill retracts back and move to next target point. Workpiece

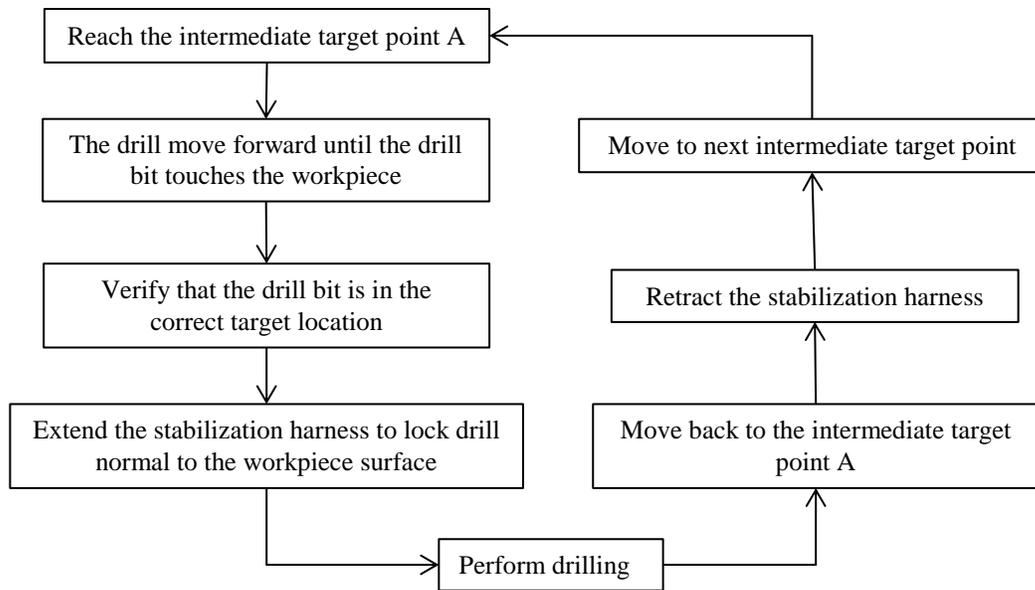


Figure 5-8: Flowchart Diagram of the Drilling Routine Cycle

The flowchart below in Figure 5-9 depicts a typical riveting routine process that starts by reaching the desired target location, preparing the rivet, inserting the rivet in hole, locking the rivet in place then perform the riveting action, then pulling back and move to the following target point.

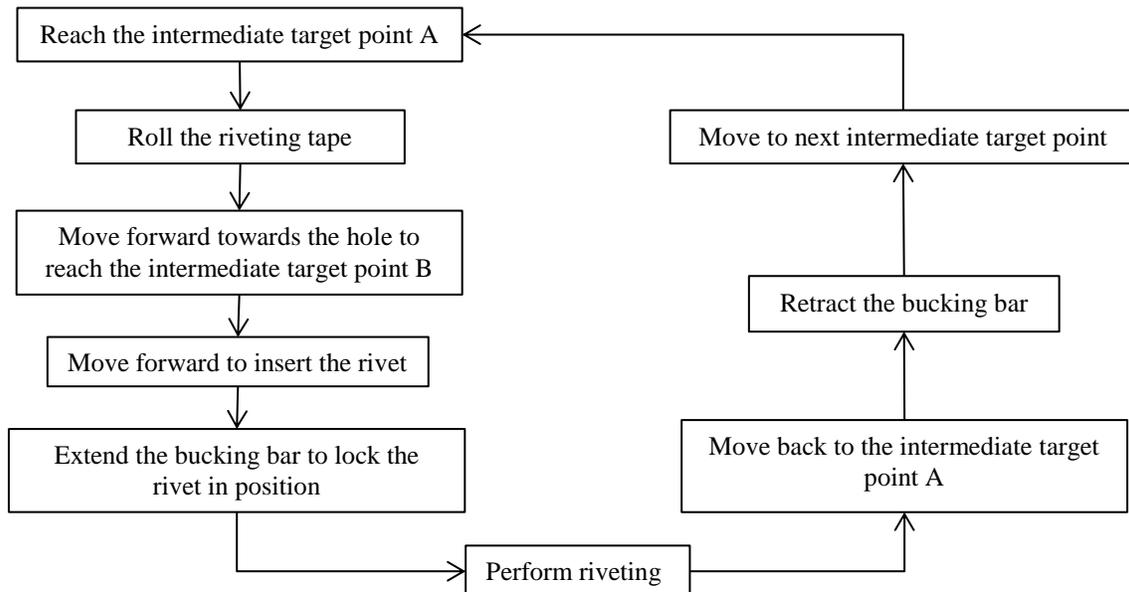


Figure 5-9: Flowchart Diagram of the Riveting Routine Cycle

The faster the motion through the intermediate steps of this riveting cycle the lower the total time will be which will result in a higher production rate. To calculate cycle time for a riveting task, the following equation can be used:

$$T_{Cycle} = (T_{2A} - T_{1A}) \quad (5.38)$$

where, T_{1A} is the time at the intermediate target F_{1A} of the F_1 target point, and T_{2A} is the time at the following intermediate target F_{2A} of the F_2 target point. The T_{1A} and T_{2A} can be calculate based on the robot velocity, acceleration and deceleration between these two target points. The velocity and acceleration shall never exceed the maximum velocity and acceleration set by the manufacturer of the robotic arm.

To calculate the possible riveting rate \mathcal{R}_{Cycle} per minute based on the riveting task cycle time, the following equation is used:

$$\mathcal{R}_{Cycle} = \frac{1 \text{ rivet}}{T_{Cycle}(\text{min})} \quad (5.39)$$

This riveting rate can then be used to estimate the required duration to complete n number of rivets per minute.

$$T_{Cycle} = \frac{n}{\mathcal{R}_{Cycle}} \quad (5.40)$$

where, n is the number of rivets. The cycle time T_{Cycle} value can then be used to calculate the total cycle time and estimate the average production rate \mathcal{R}_{Total} per unit time.

$$\mathcal{R}_{Total} = \frac{m}{T_{Total}} \quad (5.41)$$

where, m is the number of parts completed.

CHAPTER 6: SIMULATION

This chapter presents the simulation aspect of the methodology developed through this thesis as an application of the tool accessibility analysis with path and motion planning as well as the simulation results. The objective of the simulation is to validate the proposed methodology and ensure that the developed software tool is generating the correct pattern as required for the drilling and riveting tasks.

In the first section, the methodology developed through this thesis is presented in a form of a flowchart diagram that depicts the algorithm for the simulation process. Then, using the developed Matlab™ scripts, a number of test cases are presented as a validation of the algorithm flow for the tool accessibility simulation. In the following section, the implementation for aircraft assembly of the developed algorithm is presented as the developed Riveting Planner® add-in software is introduced and the operational side of the software is briefly explained. The last section involves the three test cases and their simulation validation to ensure that the generated path is valid and has no collision throughout the drilling and riveting task.

6.1. Algorithm Overview

The ultimate goal of this thesis was to develop a methodology that can be used to determine the accessibility of a tool like a drill gun, sealing gun or a riveting gun attached as an end-effector to a robotic arm into confined spaces without any collision. Through Chapters 3 to 5, the foundation of this methodology was presented using three main components, which are the tool accessibility, the path planning, and the motion planning.

The first component was the tool accessibility, where the accessibility was defined, the geometry of the tools as well as the aircraft parts were classified, and the approach of determining

the accessible space based on the Global Accessibility Area (GAA) and Global Accessibility Volume (GAV) were introduced. Then, the percentage accessibility is determined to evaluate the crowdedness of any given part. This component provides the area and volume boundaries required for the simulation application.

The second component was the path planning, where the path for drilling and riveting tasks based on the standard riveting and drilling equations were introduced. Then, the concept of approachability is presented to determine the ideal number of points required for the tool to move smoothly from one target point to the next. Then, the concept of the shortest path and path planning optimization were presented. Last, the foundation of collision-free path was presented. This component provides the path planning pattern optimization required for the simulation application.

The third component was the motion planning, where based on the generated path and the mechanical constraints of the robot, the displacement, velocity and acceleration of the robotic motion curves were reviewed and optimized. Then, the robotic cycle time was presented to evaluate the efficiency of the robot to complete all the required tasks in the minimal time possible. This component provides the motion planning optimization required for the simulation application.

The simulation application can be achieved when combining the three components together. The algorithm for the 2D and 3D space simulation based on the developed Matlab script is shown in Figure 6-1. The algorithm starts by the user inputs of the objects such as the tools, the workpiece and obstacles, with their position and orientation. Then, the script calculate the required riveting size and the total number of rivets. Next, the script calculate the initial riveting pattern.

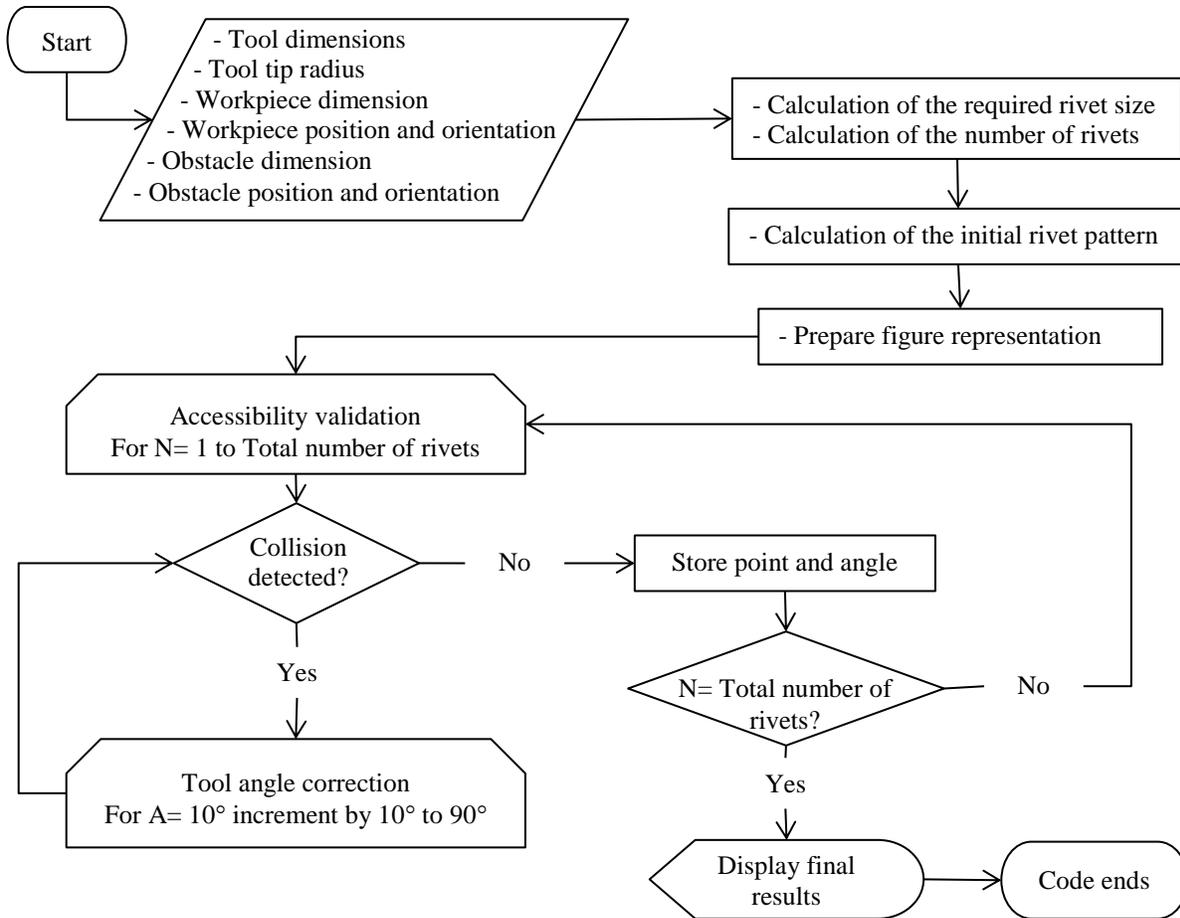


Figure 6-1: Algorithm Flowchart Representation of the Matlab Scripts for 2D and 3D

This initial pattern is then displayed as estimated rivet point. Then, the accessibility validation process gets initiated. In this process, at each rivet, the accessibility and collision interference is validated. If there is no collision detected, the rivet point is highlighted in green, which means that this target point is accessible. If there is any collision detected, the tool will start rotating in anti-clockwise direction with an increment of 10° until it reaches a total rotation of 90° . This rotation is around the axis of tool, which is perpendicular to the workpiece surface at all times, even if the surface has one or two curvatures. If the tool was able at any given angle to reach an orientation where there is no collision or interference, the riveting point is marked with an orange diamond as shown in Figure 6-2 for 2D, and Figure 6-3 for 3D cases.

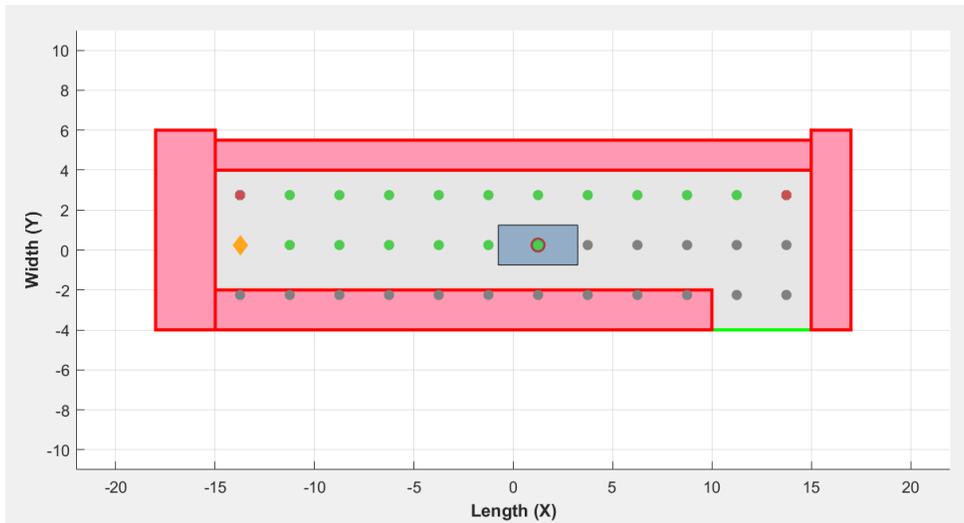


Figure 6-2: Matlab Riveting Process Check 2D

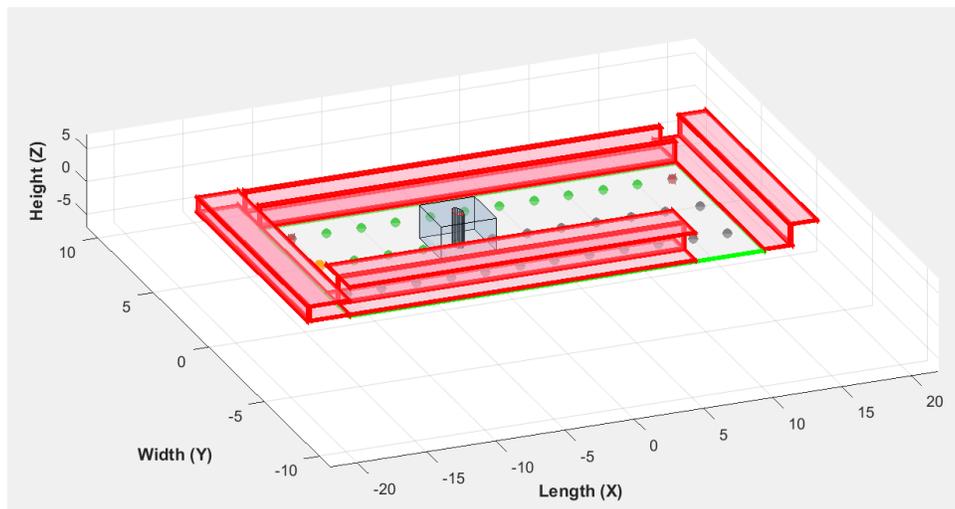


Figure 6-3: Matlab Riveting Process Check 3D

If the tool was not able to reach any successful orientation to access the required riveting point target, the riveting point is marked with a bolded red x as shown in Figure 6-4 for 2D, and Figure 6-5 for 3D cases.

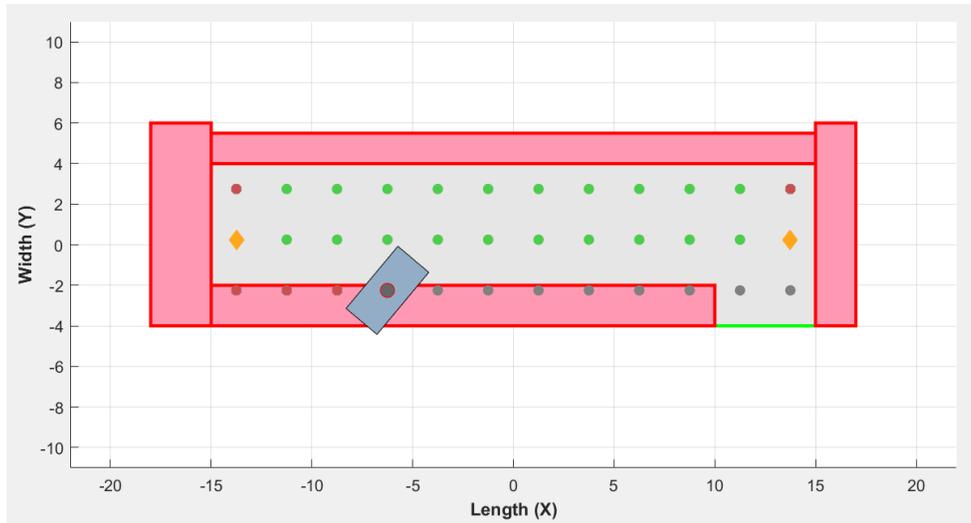


Figure 6-4: Matlab Accessibility Check 2D

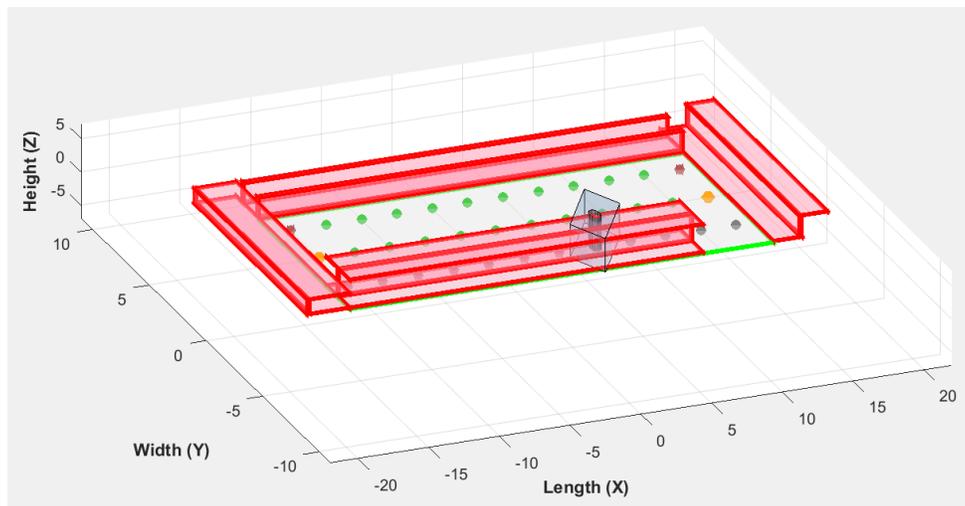


Figure 6-5: Matlab Accessibility Check 3D

In cases where the surface has one or two curvatures, when to tool change its orientation, the axis of the tool has to be always normal to the workpiece surface. This will ensure that the tool width does not collide with the workpiece's curvature as the tool rotate to access the required target point.

6.2. Simulation Test Cases

In this section, a number of test cases are presented to simulate the inputs and the outputs of the Matlab scripts for both the 2D and 3D space test cases.

6.2.1. 2D Test Cases

Presented in Table 6-1, are three test cases, where the workpiece is of a zero curvature surface and surrounded by a number of obstacles including stringers and spars in shape of U-beam, Z-beam, I-beam, and L-beam. It was observed that in most riveting cases in aerospace assembly, the spars, ribs, stringers and frame are always oriented either at 0° or 90° relative to the rivet pattern. Thus for simplicity, the test cases assumed that the obstacles and workpiece are all at the same orientation following the global frame of reference.

Test case 1 represents a part of a wing panel that is surrounded by a number of obstacles that consists of spars and ribs. Test case 2 is riveted L beam with two rows of rivets as it is the case with fuselage stringers. Test case 3 is a portion of a wing panel that is intercepted by a rib at its edge.

Table 6-1: Test Cases Sample of Inputs and Outputs for 2D

	Case 1: Overlaying Obstacle	Case 2: L Beam	Case 3: Spar
Inputs:			
Tool length	4	6	6
Tool width	2	2	2
Tool tip radius	0.35	0.35	0.35
Workpiece length	30	30	8
Workpiece width	8	6	30
Workpiece thickness	0.2	0.2	0.2
Workpiece X-axis position	-15	0	0
Workpiece Y-axis position	-4	0	0
Workpiece Theta orientation	0	0	0
Number of obstacles	4	1	1

	Case 1: Overlaying Obstacle				Case 2: L Beam	Case 3: Spar
Obstacle Number:	1	2	3	4	1	1
Obstacle length	25	2	30	3	30	8
Obstacle width	2	10	1.5	10	0.2	4
Obstacle X-axis position	-15	15	-15	-18	0	0
Obstacle Y-axis position	-4	-4	4	-4	0	0
Obstacle Theta orientation	0	0	0	0	0	0
Outputs:						
Rivets diameter	0.6				0.6	0.6
Rivets length	1.3				1.3	1.3
Number of rivets rows	3				2	3
Number of rivets	36				24	36
Distance on the side	1.25				1.25	1.25
Distance between	2.5				2.5	2.5
Status	Has some limitations				Accessible	Has some limitations
The number of rivets that can be accomplished	24				-	30
The number of rivets that requires angle correction	4				-	-
The number of rivets that are not accessible	12				-	6
The percentage of accessibility	66.67%				100%	83.33%

The results of the test cases are depicted in Figure 6-6 for case 1, Figure 6-7 for case 2, and Figure 6-8 for case 3. Note that the objects highlighted in red are obstacles and are not accessible by the tool. If the tool interfere with the obstacles that can indicate a collision can occur with the obstacles if the robot is forced to perform riveting at that inaccessible riveting point. The accessible rivet points are highlighted in green, indicating that the tool can access this target point without any interference or collision. If a target point is only accessible at a given tool orientation different from the default tool orientation, the target point is then marked with an orange diamond. If the target point is not accessible at all possible tool orientation, the target point is marked with a red x, indicating that this target is not accessible with the tool used for this task.

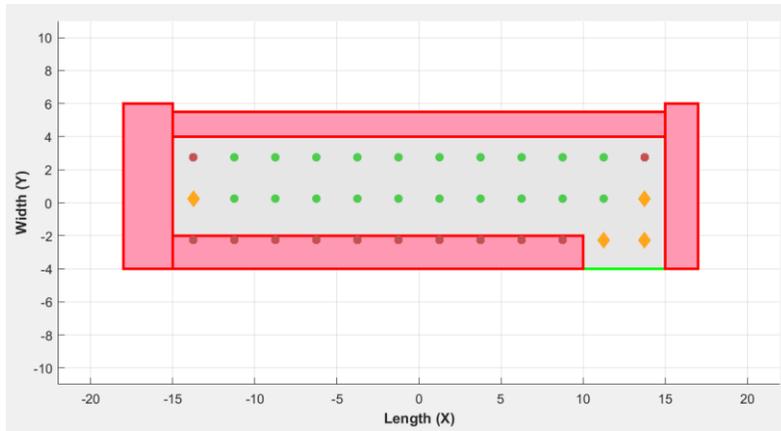


Figure 6-6: Matlab 2D Test Case 1

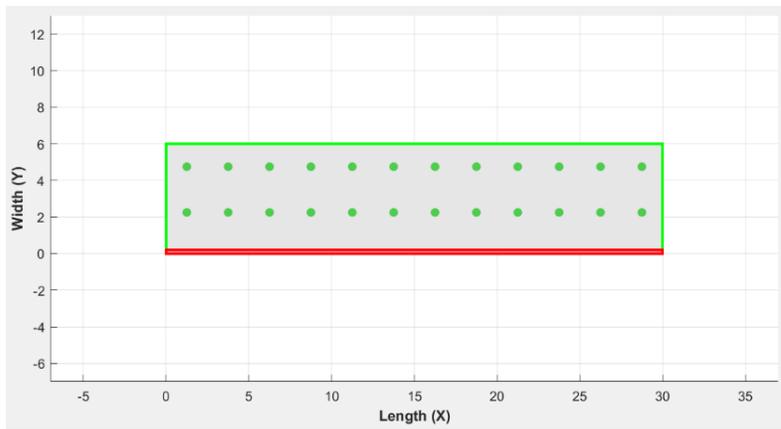


Figure 6-7: Matlab 2D Test Case 2

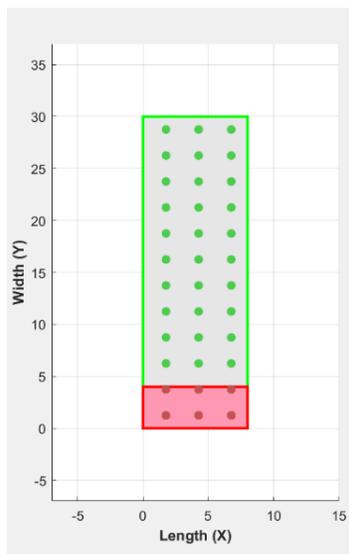


Figure 6-8: Matlab 2D Test Case 3

6.2.2. 3D Test Cases

Similar to the 2D space test cases, presented in Table 6-2, are three test cases, where the workpiece is of a zero curvature surface and surrounded by a number of obstacles including stringers and spars in shape of U-beam, Z-beam, I-beam, and L-beam. It was observed that in most riveting cases in aerospace assembly, the spars, ribs, stringers and frame are always oriented either at 0° or 90° relative to the rivet pattern. Thus for simplicity purposes, the test cases assumed that the obstacles and workpiece are all at the same orientation following the global frame of reference.

Test case 1 represents a part of a wing panel that is surrounded by a number of obstacles that consists of spars and ribs. Test case 2 is riveted L beam with two rows of rivets as it is the case with fuselage stringers. Test case 3 is a portion of a wing panel that is intercepted by a rib at its edge.

Table 6-2: Test Cases Sample of Inputs and Outputs for 3D

	Case 1: Overlaying Obstacle				Case 2: L Beam	Case 3: Spar
Inputs:						
Tool length	4				6	6
Tool width	2				2	2
Tool height	5				8	8
Tool tip radius	0.35				0.35	0.35
Workpiece length	30				30	8
Workpiece width	8				6	30
Workpiece thickness	0.2				0.2	0.2
Workpiece X-axis position	-15				0	0
Workpiece Y-axis position	-4				0	0
Workpiece Z-axis position	2				0	0
Workpiece orientation	[0,0,0]				[0,0,0]	[0,0,0]
Number of obstacles	4				1	1
Obstacle Number:	1	2	3	4	1	1
Obstacle length	25	2	30	3	30	8
Obstacle width	2	10	1.5	20	0.2	4
Obstacle height	4.4	3.4	3.2	2.4	5	5
Obstacle X-axis position	-15	15	-15	-18	0	0

	Case 1: Overlying Obstacle				Case 2: L Beam	Case 3: Spar
Obstacle Y-axis position	-4	-4	4	-4	0	0
Obstacle Z-axis position	2.2	2.2	2.2	2.2	0.2	0.2
Obstacle orientation	[0,0,0]	[0,0,0]	[0,0,0]	[0,0,0]	[0,0,0]	[0,0,0]
Outputs:						
Rivets diameter	0.6				0.6	0.6
Rivets length	1.3				1.3	1.3
Number of rivets rows	3				2	3
Number of rivets	36				24	36
Distance on the side	1.25				1.25	1.25
Distance between	2.5				2.5	2.5
Status	Has some limitations				Accessible	Has some limitations
The number of rivets that can be accomplished	26				-	30
The number of rivets that requires angle correction	1				-	-
The number of rivets that are not accessible	10				-	6
The percentage of accessibility	72.22%				100%	83.33%

The result of the test cases is depicted in Figure 6-9, and Figure 6-10 for case 1, Figure 6-11 for case 2 and Figure 6-12 for case 3 below. Note that the objects highlighted in red are obstacles and are not accessible by the tool.

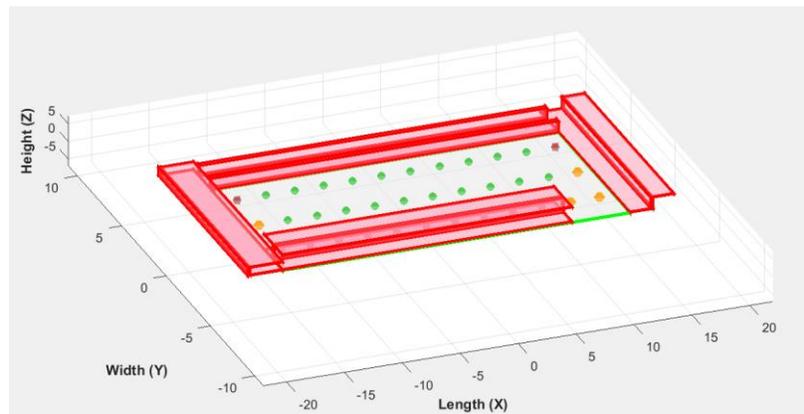


Figure 6-9: Matlab 3D Test Case 1 View at 40°

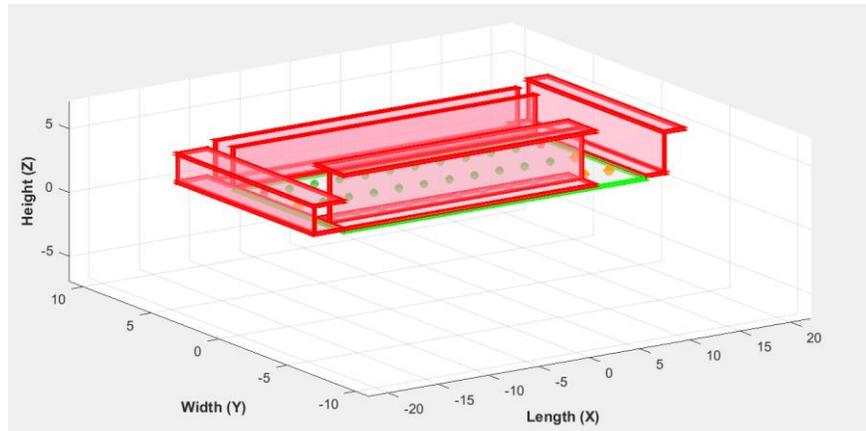


Figure 6-10: Matlab 3D Test Case 1 View at 20°

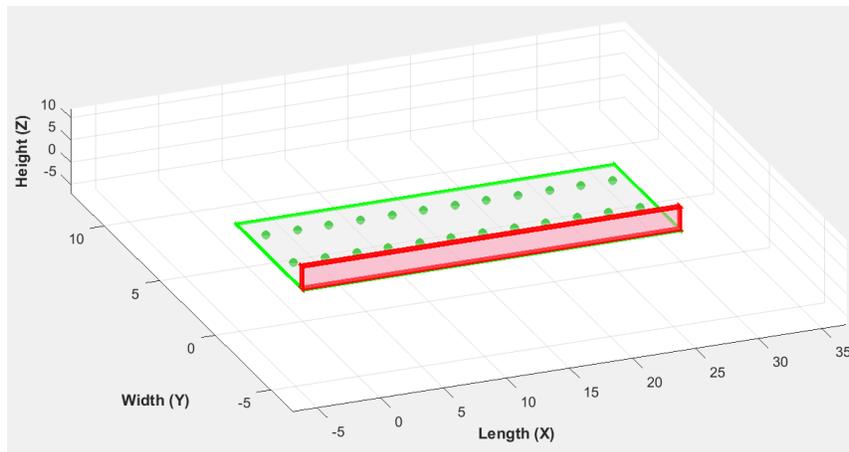


Figure 6-11: Matlab 3D Test Case 2

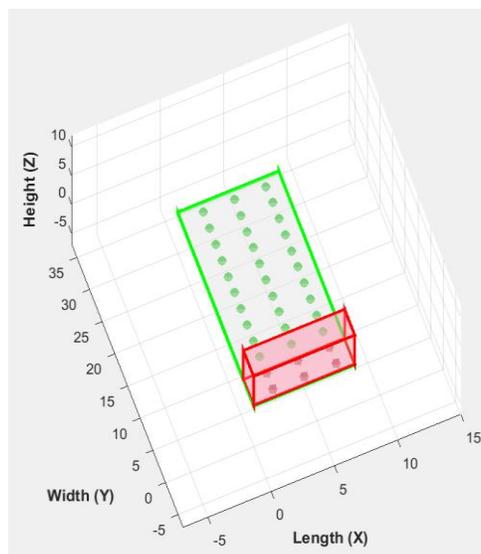


Figure 6-12: Matlab 3D Test Case 3

6.3. Algorithm Implementation for Aircraft Assembly

The Riveting Planner[®] (RP) a software application was developed using Visual Basic[™] (VB) to serve as an add-in for the RobotStudio[™] software to generate all required scripts to create the simulation of the drilling and riveting process. This software application was developed as an implementation of the methodology developed through this thesis for aircraft assembly. The algorithm used throughout Riveting Planner[®] software application can be presented in the flowchart of Figure 6-13 below. This algorithm is divided into three main blocks: Block A for user inputs and settings, Block B for riveting initial layout generation and accessibility validation, and Block C for the final riveting layout then results and execution.

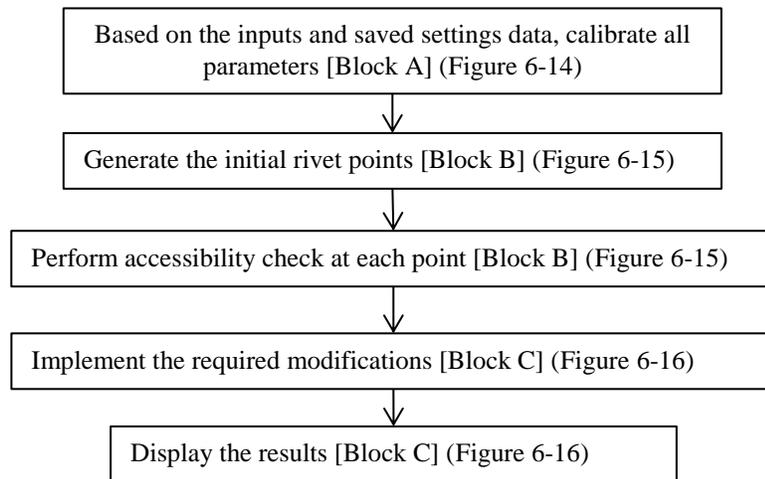


Figure 6-13: General Overview Flowchart Representation of the Add-In Code

The first segment is Block A (Figure 6-14), where most of the user inputs are entered and stored. This stage involves the user inputs for the workpiece and obstacles, as well as all the settings and calibrations required for all the systems involved in the drilling and riveting process. The user inputs includes the workpiece dimensions, workpiece curvature, obstacles dimensions, obstacles position, and obstacles orientation. This information are critical for the riveting

calculations and the initial riveting pattern planning. In addition, user inputs includes the settings and calibration required to set all the systems used in the drilling and riveting process to the global frame of reference. These target points that are generated in the global frame of reference, are then transformed using equivalent values into the local frame of each system. This is to ensure that the generated target points are valid in each system's local frame. Through the calibration process, the user enters the position and orientation of the ABB robot that carries the tool exchanger modules of the drilling gun, the sealing gun, and the riveting gun at the home position. Also, the position and orientation of the supportive CNC system that carries the bucking bar at the home position.

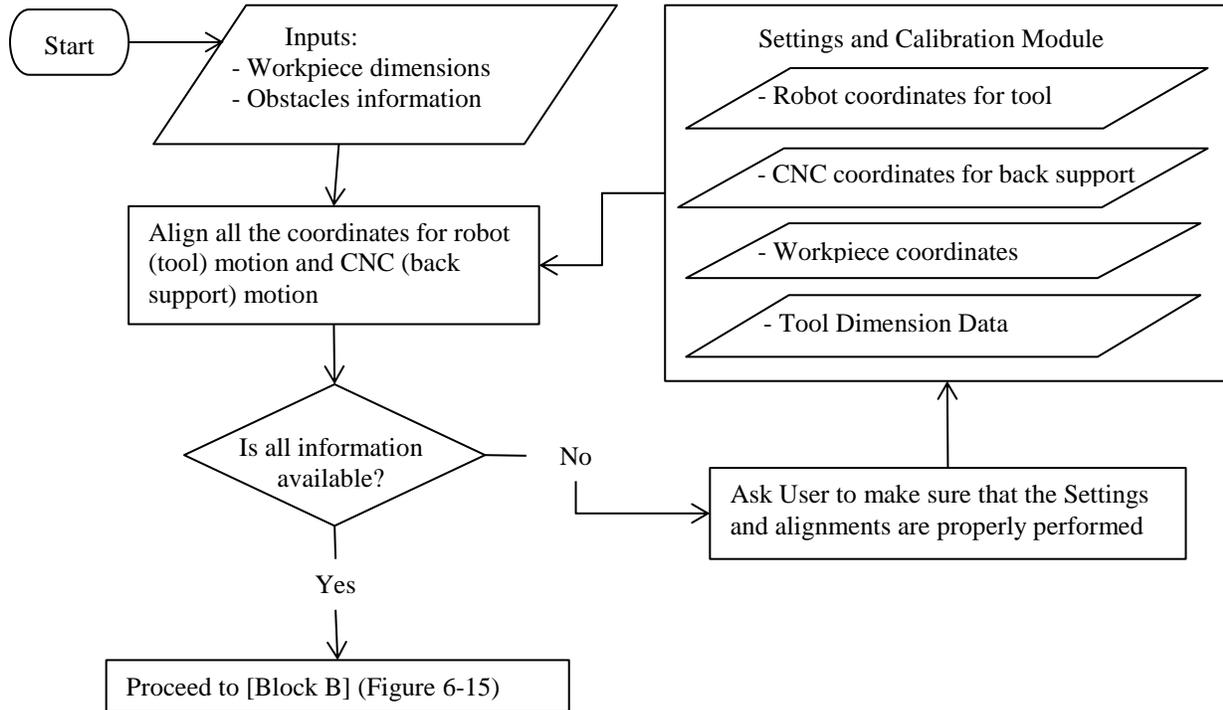


Figure 6-14: Logic Flowchart Representation of the Add-In Block A

Further, the user enters the position of the workpiece relative to the ABB Robot, and the pilot point on the workpiece. Last, the user enters all the dimensions of the tools in the tools database such as drilling gun, sealing gun, and riveting gun. All these settings and calibrations are used through the following segments in Block B and Block C.

The following segment is Block B (Figure 6-15), where all systems and components are presented in terms of the global frame of reference, global accessibility boundaries are determined, the riveting calculations are conducted, the riveting initial layout is calculated, and the tool accessibility validation is processed. Once all the tools, workpiece, and obstacles are presented in terms of global frame of reference, the process of riveting calculation based on the standard riveting equations can be processed. This part determine the type of rivets needed per riveting process, the number of rivets needed and the initial riveting pattern based on the standard riveting equations. Then, the task of accessibility analysis starts, where each target point is validated if the selected tool for the task will be able to access and reach the required target.

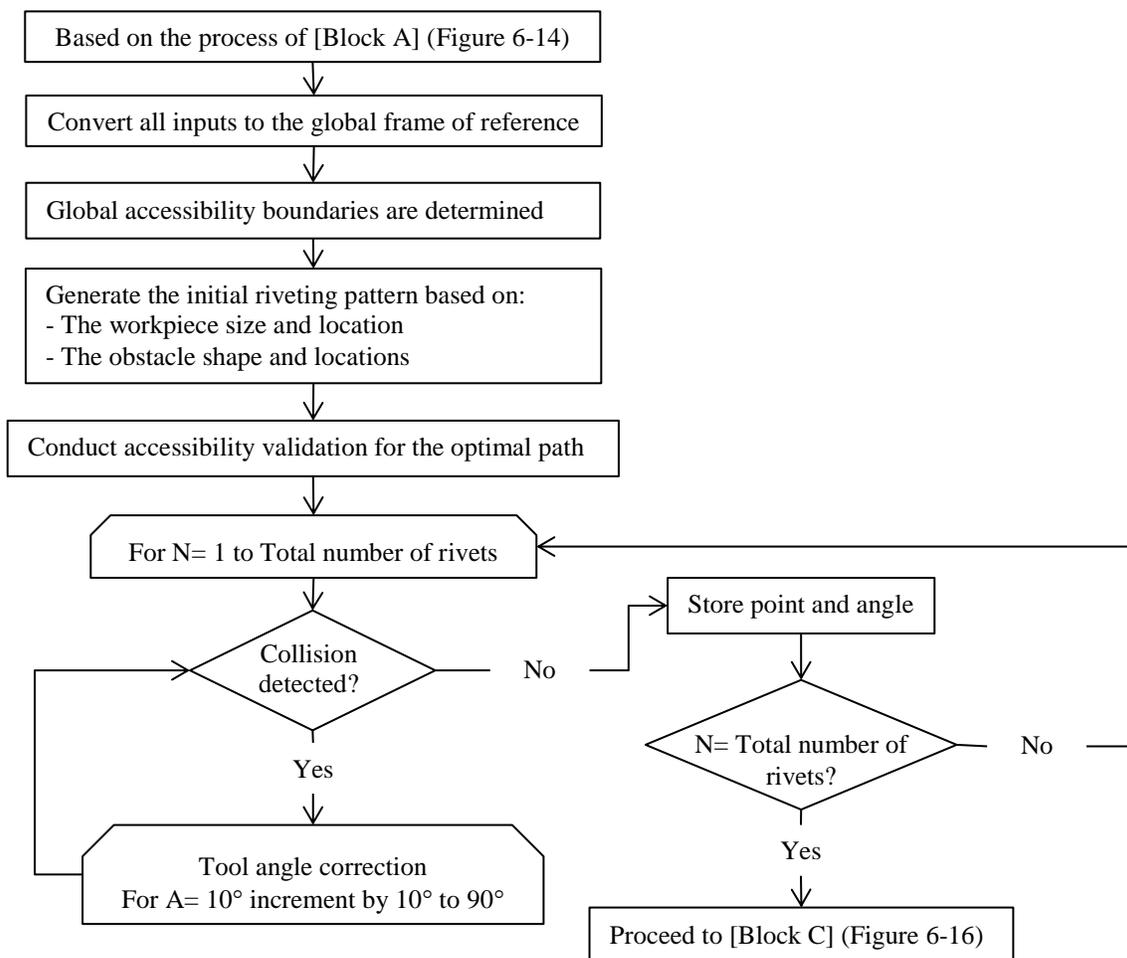


Figure 6-15: Logic Flowchart Representation of the Add-In Block B

Through the accessibility validation process, the tool size is checked if it will be able to access and reach the required target point at default tool orientation. If at a given point the tool was found to have interference with any obstacle, the tool is then rotated in an increment of 10° each until it reaches 90° . Once the tool reaches an angle at which there is no interference, the tool position and orientation are stored. If the tool reaches 90° and still experience interference, that target point is then categorized as inaccessible. Once all riveting target points are inspected for accessibility, the following segment can be initiated.

The last segment is Block C (Figure 6-16), where the final riveting layout is generated, then all parameters is back converted to the respective local frame of each system, and the result summary is displayed as well as the option of execution and simulation generation is enabled. Once all points are validated for tool accessibility, the initial riveting pattern is updated based on the points that are accessible with a given tool orientation, and the inaccessible points are removed. Once the riveting pattern is updated, the pattern is then used to generate all the required scripts for all the systems with their local coordinate systems. To display the results the user can select between three options, which includes the ABB RAPID based scripts with (.MOD) extension, the CNC G-code based scripts with (.NC) extension, and the actual coordinate systems for validation with (.txt) extension. The ABB RAPID scripts are used in the RobotStudio™ edit module that enables the actual programming of the robot and run the simulation of the desired process. The CNC G-code scripts are used by the supportive CNC system that carried the bucking bar. The coordinates' results display the plain riveting pattern.

Last, a results report is displayed to the user that includes all the statistical information, about the successful points, the points that required tool orientation change, the points that are not accessible, as well as all the other information such as the number of rivets required and rivet types.

This results report allow the user to assess the accessibility percentage and determine if some tool modification or design modification is required. Once the user is satisfied with the generated results, the user can save the generated scripts based on the appropriate formats, as well as proceeding with the simulation validation. If the simulation matches the user expectations, then the user can send the generated scripts through the RAPID module to the robot for actual execution. If the user is not satisfied with the results, the user can abort the generated results.

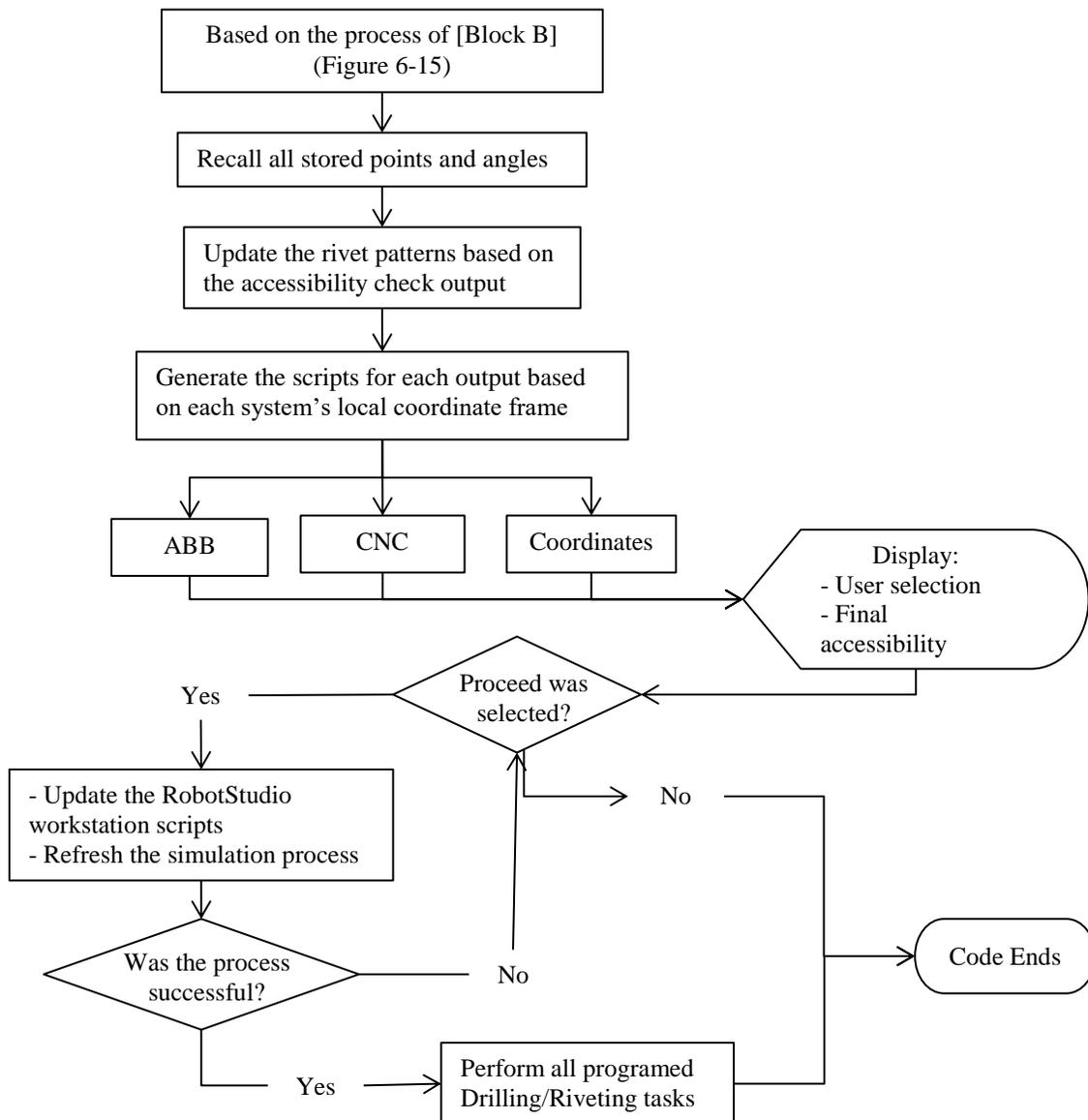


Figure 6-16: Logic Flowchart Representation of the Add-In Block C

Next is a quick overview that explain how the software operates. First, RobotStudio™ is a software created by ABB Company to facilitate the control of its robotic systems and have the ability to simulate the operation of the robot in virtual reality. In addition, this software allows a direct control of the robot when operated in the online mode. This enables the user to fully operate and control the robot using a remote computer system. More, this RobotStudio™ software has the ability to run user developed add-ins based on VB and C# scripts and user interface that can perform tasks and control the robot. For the scope of this thesis, the Riveting Planner® software application was developed to function as a tool to generate all the required scripts needed to control the ABB robot and its supporting CNC system to perform drilling and riveting tasks.

The first step before using the application is to create the workstation that mimic the actual workstation of the robot as shown in Figure 6-17 and Figure 6-18. This will ensure that the virtual reality representation of the workstation and the generated results matches what the actual environment of the robot during assembly encounters at all times.

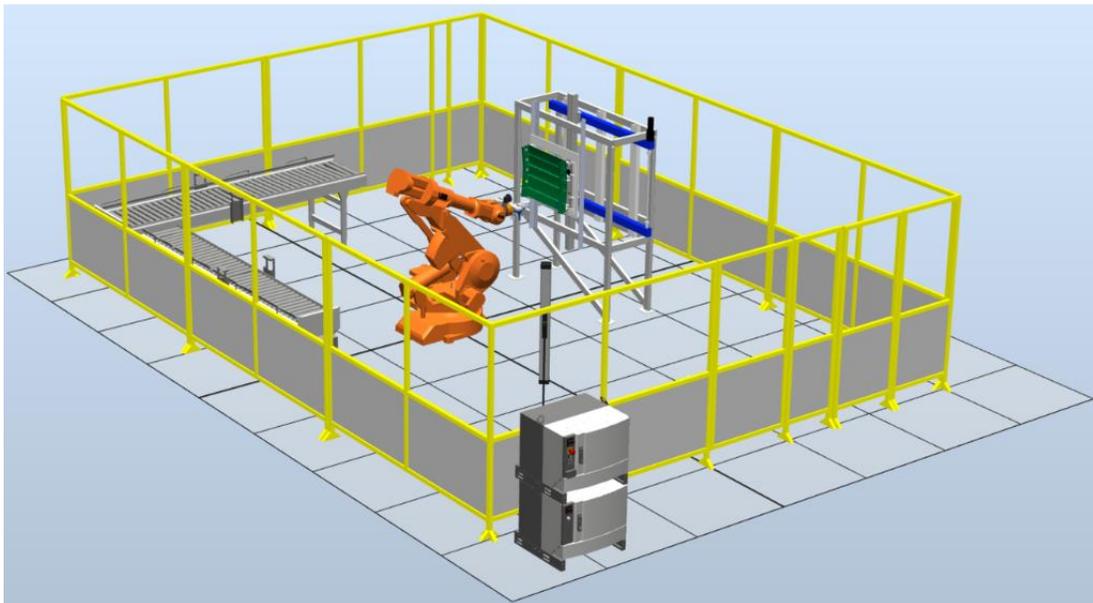


Figure 6-17: ABB Riveting Workstation

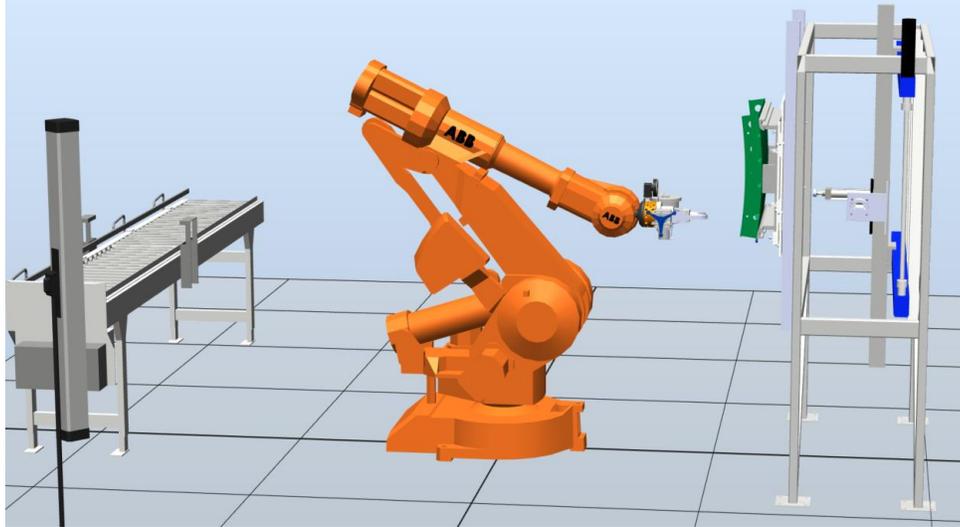


Figure 6-18: Riveting a Fuselage Section in Robotstudio

Once the workstation is created, the Riveting Planner[®] application package can be installed. Once the installation is complete, the user can easily launch the application from within the RobotStudio[™] add-ins tab. The main menu of the application as shown in Figure 6-19 will be displayed once the user launch the application. For the operation of the software application and all the windows that are within the software, please refer to Appendix C.

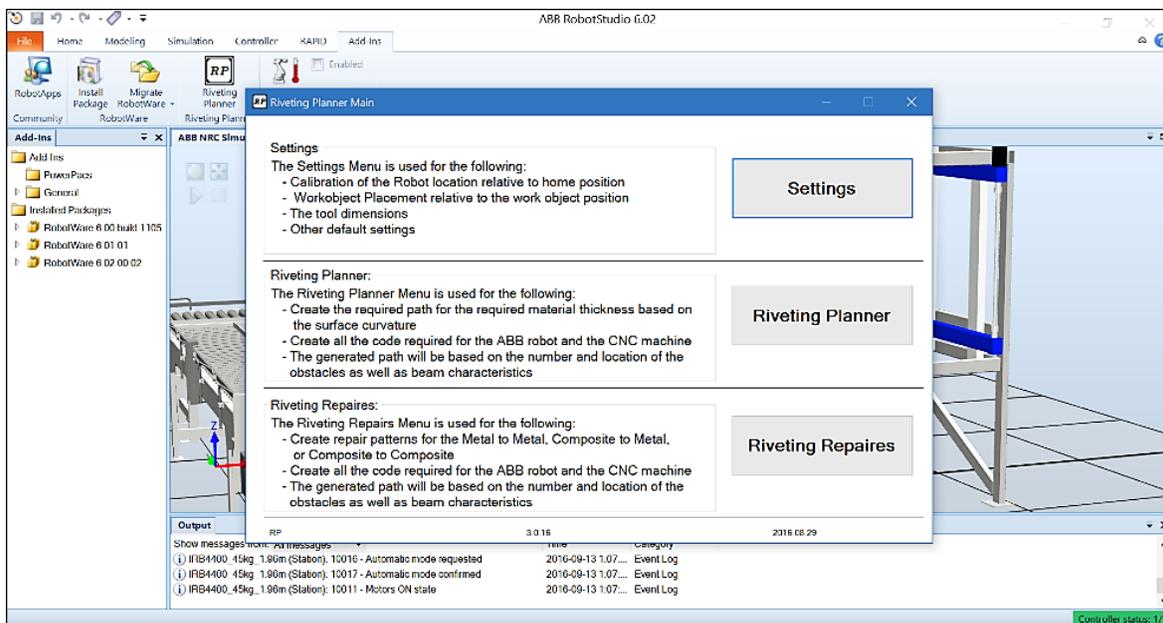


Figure 6-19: Riveting Planner[®] Add-In Tool in RobotStudio

Once the Riveting Planner[®] application generate all the required target points, the RobotStudio[™] can simulate the path that the ABB Robot and the bucking bar attached to the supportive CNC system has to follow to accomplish the required task. To further validate if the tool will experience any type of collision, the RobotStudio[™] has a feature that allow the user to visually see if the tool is within the specified safety tolerance distance or not. If the tool is within the safety tolerance distance, which means that the tool is close to the workpiece but not in collision, then the workpiece turns the color to yellow as in Figure 6-20 (a). If at any moment the tool collide with the workpiece the color of the workpiece turns to red as in Figure 6-20 (b), indicating a collision has occurred.

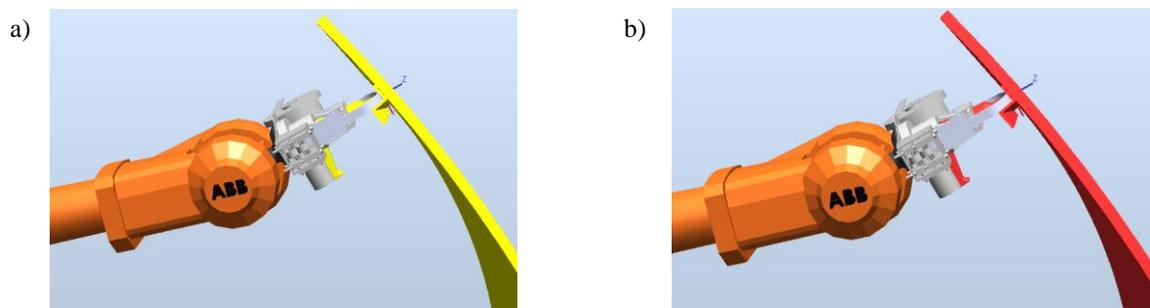


Figure 6-20: RobotStudio Collision Avoidance Validation

If the user run the simulation and all the points generated does not cause the workpiece graphical representation to turn to the red color, then the generated pattern for is successful. If the workpiece graphical representation to turn to the red color, then the generated pattern is not safe to implement. Reasons for the workpiece graphical representation to turn to the red color, is the position and orientation of the objects available in the workstation virtual environment are not accurately entered to the software application, or the calibration is not properly conducted. Thus, in situations where workpiece graphical turns to the red color, it is advisable to validate all user input data.

6.4. Test Cases of Aircraft Assembly Simulation

In this section, a number of test cases are presented based on the methodology implementation for aircraft assembly. The results generated are based on the Riveting Planner[®] software application for 3D space test cases. For simplicity, all the stringers have the same size and shape but have different orientations.

6.4.1. Test Case with Wing Section

The first test case was a top portion of a wing panel that has a large radius of curvature and a total number of four stringers to be riveted. In this case, a portion of the wing skin was extracted as shown in Figure 6-21 and Figure 6-22.

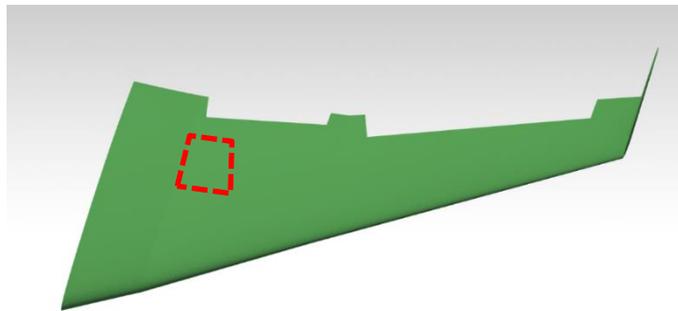


Figure 6-21: Test Case with Wing Section

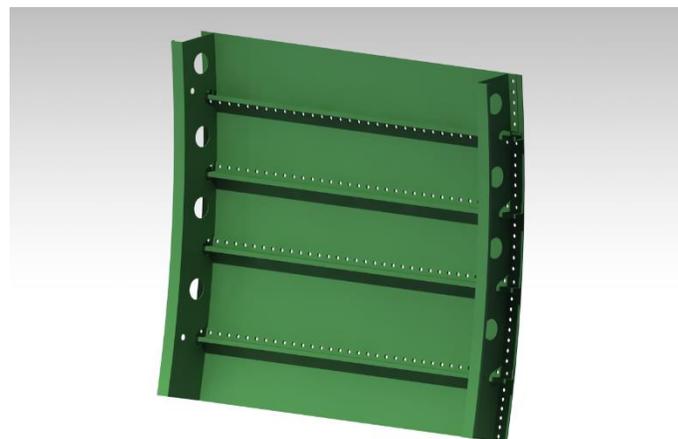


Figure 6-22: Test Case with Skin Panel Section of a Wing

The wing skin section in RobotStudio™ is depicted in Figure 6-23, where the skin section is attached to the mount frame.

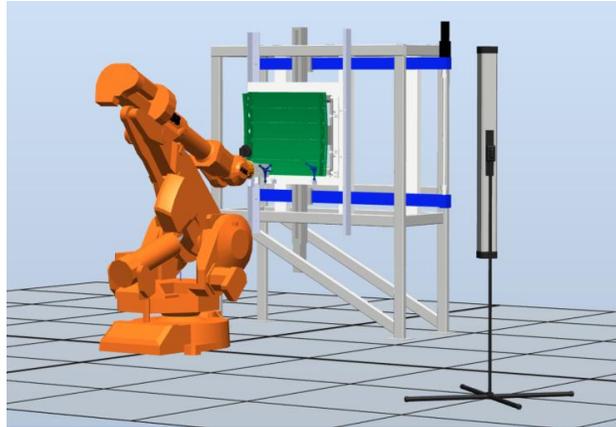


Figure 6-23: Test Case with Skin Panel Section of a Wing ABB Workstation View

The Table 6-3 capture the summary of the input and output results for the test case.

Table 6-3: Results for the Test Case with Skin Panel Section of a Wing

Parameter	Input/output
Workpiece length	567 [mm]
Workpiece width	650 [mm]
Workpiece thickness	1.7 [mm]
Number of stringers to rivet	4
Total number of obstacles	6
Surface type	Curved with $R = 4075$ [mm], $\theta = 8^\circ$
Results:	
Rivets diameter	5.5 [mm]
Number of rivets rows per stringer	1
Number of rivets in total	116
Distance on the side	11 [mm]
Distance between	22 [mm]
Status	Has some limitations
The number of rivets that can be accomplished	108
The number of rivets that requires angle correction	0
The number of rivets that are not accessible	8
The percentage of accessibility	93.1%
Simulation validation	Pass

The output results matches the analytical solution and as presented, the simulation confirms that the generated result is acceptable and as desired. The generated result confirms that the part is accessible for 108 rivets out of 116 total number of rivets, and the percentage accessibility was found to be 93.1%, which is considered accessible.

6.4.2. Test Case with Fuselage Section

The second test case was a segmented portion of a fuselage that has a small radius of curvature and a total number of three stringers to be riveted. In this case, a portion of the fuselage skin was extracted as shown in Figure 6-24.



Figure 6-24: Test Case with Fuselage Section

The fuselage skin section in RobotStudio™ is depicted in Figure 6-25.



Figure 6-25: Test Case with Fuselage Section ABB Workstation View

The Table 6-4 capture the summary of the input and output results for the test case.

Table 6-4: Results for the Test Case with Fuselage Section

Parameter	Input/output
Workpiece length	1130 [mm]
Workpiece width	2180 [mm]
Workpiece thickness	2 [mm]
Number of stringers to rivet	3
Total number of obstacles	10
Surface type	Curved with $R = 1550$ [mm], $\theta = 41^\circ$
Results:	
Rivets diameter	6.35 [mm]
Number of rivets rows per stringer	1
Number of rivets in total	264
Distance on the side	12.64 [mm]
Distance between	25.3 [mm]
Status	Has some limitations
The number of rivets that can be accomplished	258
The number of rivets that requires angle correction	0
The number of rivets that are not accessible	6
The percentage of accessibility	97.73%
Simulation validation	Pass

The output results matches the analytical solution and as presented, the simulation confirms that the generated result is acceptable and as desired. The generated result confirms that the part is accessible for a total of 258 rivets out of 264 total number of rivets, and the percentage accessibility was found to be 97.73% which is considered accessible.

6.4.3. Test Case with Cockpit Section

The third test case was a segment portion of a cockpit that has two curvatures, where one of them has a large radius of curvature and a total number of three stringers to be riveted. In this case, a portion of the cockpit skin was extracted as shown in Figure 6-26.

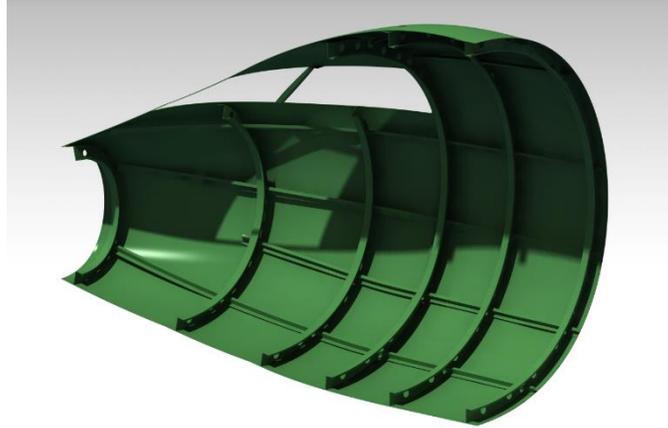


Figure 6-26: Test Case with Cockpit Section

The cockpit skin section in RobotStudio™ is depicted in Figure 6-27.

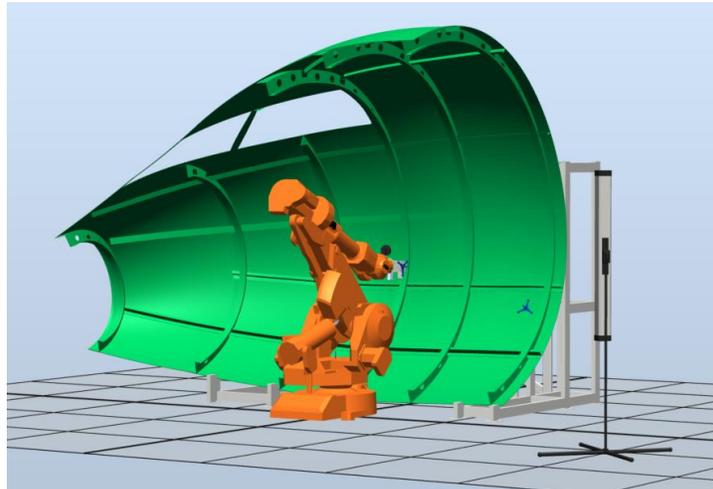


Figure 6-27: Test Case with Cockpit Section ABB Workstation View

The Table 6-5 capture the summary of the input and output results for the test case.

Table 6-5: Results for the Test Case with Cockpit Section

Parameter	Input/output
Workpiece length	1242.6 [mm]
Workpiece width	1370 [mm]
Workpiece thickness	2 [mm]
Number of stringers to rivet	3
Total number of obstacles	9
Surface type	Curved with $R_v = 1657$ [mm], $\theta_v = 46^\circ$, and $R_h = 8100$ [mm], $\theta_h = 6.8^\circ$

Parameter	Input/output
Results:	
Rivets diameter	6.35 [mm]
Number of rivets rows per stringer	1
Number of rivets in total	162
Distance on the side	12.7 [mm]
Distance between	25.6 [mm]
Status	Has some limitations
The number of rivets that can be accomplished	153
The number of rivets that requires angle correction	0
The number of rivets that are not accessible	9
The percentage of accessibility	94.44%
Simulation validation	Pass

The output results matches the analytical solution and as presented, the simulation confirms that the generated result is acceptable and as desired. The generated result confirms that the part is accessible for a total of 153 rivets out of 162 total number of rivets, and the percentage accessibility was found to be 94.44% which is considered accessible.

CHAPTER 7: CONCLUSION AND FUTURE WORK

7.1. Conclusion

Robotic applications in aerospace manufacturing and assembly are limited today as most of the aircraft parts are relatively small or have complex shapes and are located in crowded spaces. The complexity and crowdedness pose great challenges when planning for robotic automated tasks like drilling or riveting. These challenges include tool accessibility, path planning, and motion planning. For the tool accessibility, a number of methods and approaches were investigated and developed in this thesis.

First, the tool accessibility was analyzed based on the global accessibility area (GAA) approach, as an application for 2D accessibility analysis. This approach was proven successful when studying accessibility for zero and one curvature surfaces. Then, the tool accessibility was analyzed based on the global accessibility volume (GAV) approach, as an application for 3D accessibility analysis. This approach was proven to be successful when studying accessibility for zero, one and two curvatures surfaces. For the path panning, a number of path planning technique where reviewed and a process methodology was developed based on the results of the tool accessibility analysis. This methodology is then used to develop and optimize a proper path planning based on an approach that seeks the shortest path and fastest target approachability. For the motion planning, a number of motion planning technique where reviewed. The motion optimization was based on the shortest time possible with minimal jerk to complete the required task.

As a practical application of this thesis, a software application was developed. This software combines tool accessibility with path and motion planning to generate all the required target points

with speed and acceleration information for the RobotStudio™. This generated RAPID script is then synced with the virtual station in RobotStudio™, to generate a simulation that mimic the drilling and riveting process. This process ensure that the robot is able to accomplish all the drilling and riveting points in the locations where determined to be accessible. This software application was proven to be successful when tested with a number of aircraft parts.

7.2. Contribution

The contribution of this thesis can be summarized as the following:

- A systematic tool accessibility analysis methodology is developed for aircraft manufacturing and assembly. This methodology takes in consideration the path and motion planning for the generation of collision-free path for robotic drilling and riveting. It also takes into account the geometrical classification, crowdedness and accessibility level of major aircraft parts including wing, fuselage, cockpit and empennage. This methodology also identifies all the target points where accessibility is limited and provides adequate feedback to the design team to modify and improve the design of a given part.
- The global accessibility area (GAA) and global accessibility volume (GAV) approaches are developed and used to determine the accessible boundaries in order to conduct the tool accessibility analysis. These approaches took in consideration the possible curvatures of the part.
- The approachability order was developed to mathematically express the number of points required for the robot to move to next target without experiencing high jerk effect and to avoid any obstacles that exist between the riveting rows.

- A software application was successfully developed to support the practical application part of the thesis. This software implements the GAA and GAV approaches to analyse accessibility, plan and optimize the required path based on the shortest distance approach and motion profile, then generates a fully functional simulation of the drilling and riveting tasks. This software generates all the required scripts needed taking in consideration the calibration of the tool, the robot and the supporting CNC system.

7.3. Future Work

Tool accessibility with path and motion panning for robotic automation in aerospace manufacturing and assembly is a very important topic that requires a lot of research. Five areas for future work are suggested based on the research conducted through this thesis:

First, to expand on the contribution of this thesis, the tool accessibility analysis should be performed on actual aircraft parts of complex shapes that have more than two curvatures per part, and actual tool geometry. This will ensure a much more accurate accessibility representation results when dealing with actual assembly tasks. This process could utilize the actual CAD drawings of the parts, without the need to simplify the parts complexity to its basic elementary geometry.

Second, the entire robot kinematics, link geometry and sizes should be used to perform accessibility analysis when dealing with complex aircraft parts. This will ensure that the entire robotic structure is collision-free at all link motions when operating in a confined space and not only the end-effector tool as was the focus of this thesis. This should also include the error correction generated when conducting tool calibration in industrial applications.

Third, expand on the path planning techniques presented in this thesis, including the cell decomposition method mentioned in section 2.2.1.2 as another approach to use for accessibility analysis or utilize a real time 3D laser scanner technique as mentioned in section 2.2.3.2. This will ensure a more accurate representation of the accessible boundaries and ensure that the generated path has minimum energy. This will also improve the motion planning aspect of the generated path, for better energy efficiency.

Fourth, utilize the developed software application to test the generated scripts and target positions on real aircraft parts as validation cases. In other words, to conduct a number of tests on physical aircraft parts, and validate the robot execution of the generated path in real life and not only in RobotStudio™ virtual reality environment. These tests shall improve the accuracy of the generated scripts and implement all the required calibrations and modifications needed based on the actual test results.

Fifth, develop a redesign methodology for manufacturing to implement the suggestions provided by the accessibility analysis methodology to address the targets that prove to be inaccessible. This redesign methodology shall include the modification of the rivet position and tool shape and design. As a result, the future generated aircraft part designs will be more compatible with robotic automation assemblies, which will revolutionize the aircraft assembly as the foundation of the next century of aircraft manufacturing.

APPENDICES

Appendix A - Standard Riveting Calculations

Rivets are permanent mechanical fasteners that are widely used in industries to join two or more panels together especially in aircraft assembly. They are known to have lower cost than that of threaded fasteners, because they can be produced in large quantities on high-speed heading machines with little scrap loss. In addition, the time taken by labors to set the rivets in the parts through the assembly process is much less compare to threaded fasteners. Besides, this type of fasteners is well suited for high-speed automatic assembly operations, as it can be clinched in place without the need for advanced installation processes.

Like every type of fasteners, rivets have well known advantages and disadvantages. As for the advantages, rivets can be made out of any material that can be cold worked like metallic alloys like aluminum alloys which is lighter than steel. Besides, they can join any similar or dissimilar materials, metallic or non-metallic, in various thicknesses. They may have a variety of finishes such as plating or paint and can be applied to parts that are painted or have received other finishes. More the rivets like aluminum alloy solid-head rivet increase in size and strength when installed, which is better than a steel bolt, as the steel bolts tend to decrease in diameter after installation when torqued. For its disadvantages, rivets have a lower tensile and fatigue strengths compare to bolts or screws, where high tensile loads may pull out the clinch, or severe vibrations may loosen the fastening. In addition, riveted parts cannot be disassembled for maintenance or replacement without destroying the rivet. More, rivets not made with precision when produced in volume especially when compared to the screw-machine parts. Further, riveted joints are known to be

neither watertight nor airtight; however, such issues can be overcome using a sealing compound [24].

There are many types of rivets that are used in the aerospace industry, which include Solid-head, Semi-tubular, Flush-head, Blind, Friction-lock and others, where each rivet type has its advantages and its limitations. Solid-head rivets (Figure A-1) are the most famous rivets and widely used in the aerospace industry due to their lowest weight and cost compare to other types, good clamp-up properties, high rigidity, good static and fatigue in shear joint. However, Solid-head rivets has some limitations which include limited static shear, low tension shear and fatigue, they are associated with high noise level during installation, besides they cannot be easily removed [24].

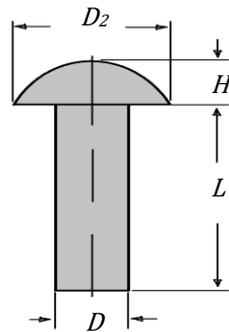


Figure A-1: Rounded Solid-head Rivet

A Solid-head rivet consists of a cylindrical shaft with smooth profile that has a head on one end and a buck-tail on the opposite end which get bucked during the riveting process [2]. This process uses a principle similar in function to the open-die forging principle, where a rivet gets compressed in a drilled hole that is slight larger than the rivet to attach two panels together. In riveting, a rivet gun is used to punch the rivet, and a bucking bar is used to die the buck-tail of the rivet. The rivet gun uses a series of impulsive punches to expand buck-tail of the rivet to fill the hole and clamped with the sheet metal [49].

A.1. Rivet Patterns and Spacing Calculations

Rivet spacing is a mathematical process that is used to calculate the distance between the centers of two adjacent rivets while taking in consideration all sort of bending and shear stresses that can be applied on the metal sheet and the rivet respectively [49]. This distance can be used to generate the required riveting pattern layout to allow the robot to drill and rivet the rivets accurately. Generally, multiple layout patterns that can be used to rivet two or more object together based on the design requirements. The simplest one is the riveting two sheet of metal together using one rows of rivets. Other approaches can involve two, three, four or more number of rows of rivets and different types (Chain or Zigzag) as shown in Figure A-2 and Figure A-3.

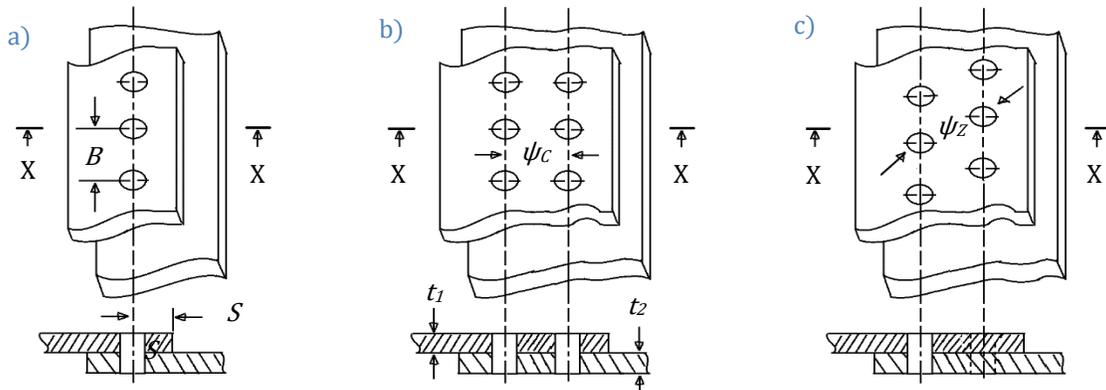


Figure A-2: Layout Patterns for: a) Single Riveted Lap Joint, b) Double Chain Riveted Lap Joint, c) Double Zig-zag Riveted Lap Joint [54]

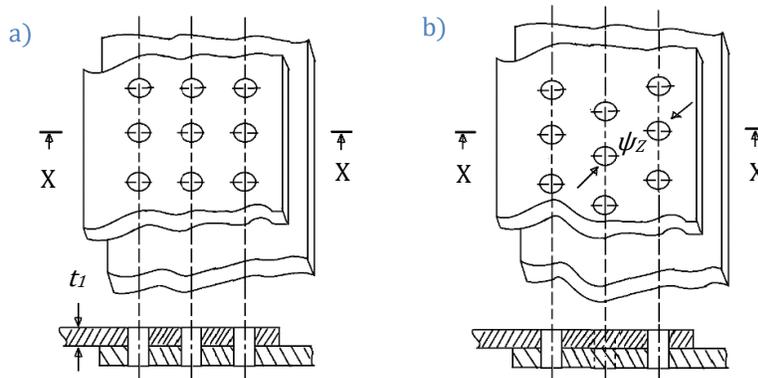


Figure A-3: Layout Patterns for: a) Triple Chain Riveted Lap Joint, b) Triple Zig-zag Riveted Lap Joint [54]

To perform a rivet spacing calculation for riveting two sheets with a single row, the first step in this process is to measure the thickness of the sheets t_1 and t_2 , then the length of the sheet W at the side which will get riveted as shown in Figure A-4.

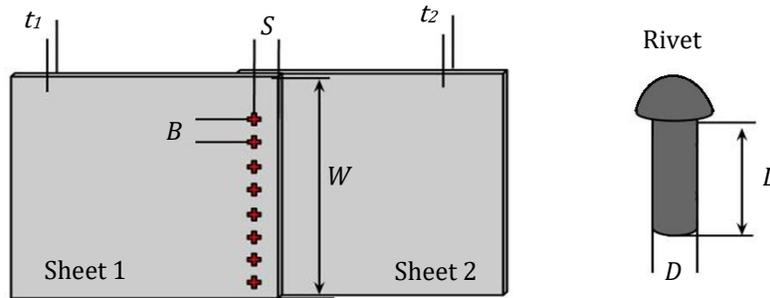


Figure A-4: Diagram of the Calculated Variables

To determine the recommended rivet diameter D and length L , the following formulas are used:

$$D = 3 \times t \quad (\text{A.1})$$

where, t is the thickness of the thickest sheet (Figure A-5). The rivet diameter D has to be in standard scale, thus the calculated rivet diameter has to be rounded up to the 32nd of an inch. It is important to note that when the hole is drilled, the drill bit tolerance range will result in a slightly larger diameter.

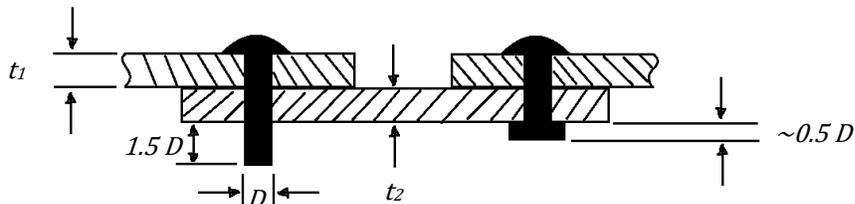


Figure A-5: Riveting Diagram [49]

The length of the rivet to be used can be calculated using the following formula:

$$L = t_1 + t_2 + 1.5D \quad (\text{A.2})$$

where, L is the recommended rivet length, t_1 and t_2 are the thickness of sheet 1 and sheet 2. The resulting answer can be rounded up to the 16th of an inch. If the thickness of sheet 1 and sheet 2 are the same, then equation (A.2) can be simplified to be:

$$L = 2 \times t + 1.5D \quad (\text{A.3})$$

Rivets are made according to standards based on their usage and application. These rivets are coded using letters and numbers, which indicate the standard type, the alloy used by the rivet, the rivet diameter and the rivet length. Standard types are National Aircraft, Military, and Air Force-Navy which uses standard letter of “NAS”, “MS20”, and “AN” respectively. An example of rivet designation is MS20 426 AD 5 - 8 where MS20 refers to Military Standard, 426 is the head shape, AD indicates the alloy like 2117-T4, the number 5 is the rivet diameter in $5/32^{\text{nd}}$ of an inch, and 8 is the length of the rivet in $8/16^{\text{th}}$ of an inch. Therefore, the rivet diameter and length calculated in equation (A.1) and (A.3) had to be rounded up to the 32^{nd} of an inch and 16^{th} of an inch respectively, to be standardized.

To determine the n number of rivets required for the riveting task:

$$n = \frac{W}{4D} \quad (\text{A.4})$$

If the resulting number has decimal places, the answer has to be rounded up to the next whole number.

The distance on the side S , and the distance between rivets B can be calculated using the following equations:

$$S = \frac{W}{2 \times n} \quad (\text{A.5})$$

$$B = \frac{W}{n} \quad (\text{A.6})$$

These calculations will determine the required distance for the rivet from the edges of the sheet and between each rivet. For multiple rows scenario, a distance B is used as a distance between each row.

A.2. Complex Riveting Patterns

Rivets are not only used to attach flat panels or parts with simple shapes, rivets are also used to attach various complex shapes together. Complex shapes require some special analysis that takes in consideration the structure integrity and performance of the part under various stresses and loads. Complex shapes include patches, curved panels, stringers, spars, and flanges.

Patches are sheets of metals that are used to repair damages created to panels like fuselage or wing skin panels without the need of replacing the damaged panels as shown in Figure A-6. These patches have the ability to carry the stress load applied on the damage area and reduce any chance of crack propagation. The layout and the spacing of the rivets has to be carefully calculated to allow the patch to perform its intended purpose [24]. Highlighted below is an example of a patch that is used to repair a crack cause to a fuselage panel. Rivet spacing calculation for patches uses a different approach compared to normal rivet spacing. For patches, the patch shape has to be octagonal as shown in figure below, where this shape reduces any excessive material that is not needed for the patching purposes [16]. To stop the crack from further propagating, two stop drill holes has to be placed across the lines of maximum stress, then the crack has to be cleaned.

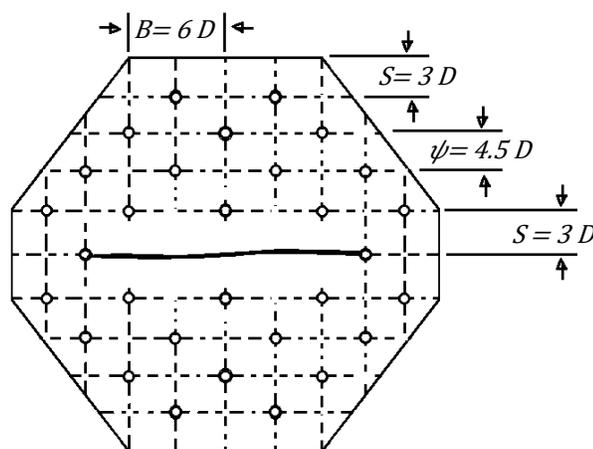


Figure A-6: Typical Layout for a Patch Repair for Damaged Fuselage Skin [16]

Then, the following equation is used to calculate the number of rivets required for the repair:

$$n = \frac{l \times t \times \sigma}{\tau} \quad (\text{A.7})$$

where, n is the of rivets required for each repair, l the length of the damaged area, t is the damaged panel thickness, σ is the nominal tensile strength of material (including factor of safety), and τ is the shear strength of rivet. Normally, such patching process will require form 3 to 4 rows of rivets on each side parallel to the crack depending on the crack size and shape. The distance S from the edge of the patch, the distance B between each rivet, and the distance ψ between rivet rows can be calculated using the following equations:

$$S = 3 \times D \quad (\text{A.8})$$

$$B = 6 \times D \quad (\text{A.9})$$

$$\psi = 4.5 \times D \quad (\text{A.10})$$

To layout the rivet pattern, it is recommended to leave a distance of S as well from the crack before laying out the first row of rivets from both sides parallel to the line of the crack. Based on the structure analysis, 75% of the rivets that are used to attach the patch are able to carry all forces and stresses applied on this section of damaged skin, while 25% of the rivets are redundant for safety purposes [16].

Stringers, spars, and flanges are also considered complex shapes as they have a limited space that can be critical on each side of the rivet. Stringers for instance, can be in L, U, or Z shapes, where the base of each shape has a limited width to allow the required distance for rivet attachment. It is recommended for metallic stringers to leave a distance of 0.03 [in] from the center of the stringer's bend and a minimum normal edge distance of $2 D$, with a zigzag pattern as shown in

Figure A-7. This layout ensures that the stress is not concentrated in a very small section close to the stringer edge, and it provides better attachment stability.

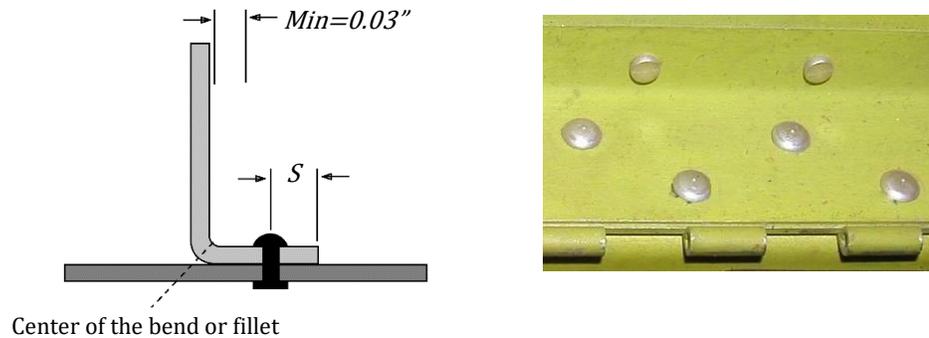


Figure A-7: Sample of Riveted Stringer with a Supportive Schematic Representation [50]

For composite structures, it is recommended to use a distance of 0.05 [in.] from the center of the stringer's bend and a minimum normal edge distance of $2.5 D$ [50]. Other much complex shapes requires an advanced structure analysis to properly layout and plan the proper rivet layout pattern. Generally, the product designer conducts such analysis in the designing phase.

Appendix B - Coordinate's Transformations

Consider P be a point in 3D space as shown in Figure B-1 below that is presented in the form of a vector with respect to a coordinate frame.

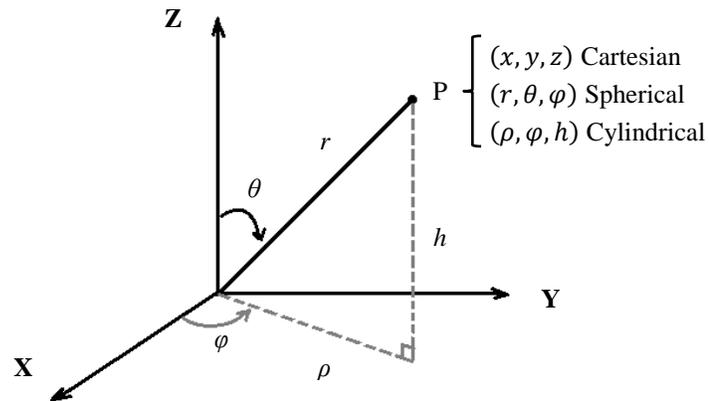


Figure B-1: Coordinates Representation

This vector \vec{P} can be presented in a number of coordinate systems. For the case of the Cartesian coordinates, the vector \vec{P} can be presented in terms of $[x, y, z]$ coordinates, where $x = [1,0,0]^T$ in the x axis direction, $y = [0,1,0]^T$ in the y axis direction, and $z = [0,0,1]^T$ in the z axis direction. For the case of the Spherical coordinates, the vector \vec{P} can be presented in terms of $[r, \theta, \varphi]$ coordinates, where $r = [1,0,0]^T$ is the radius of the sphere, $\theta = [0,1,0]^T$ is the colatitude angle, and $\varphi = [0,0,1]^T$ is the longitude angle. For the case of the Cylindrical coordinates, the vector \vec{P} can be presented in terms of $[\rho, \varphi, h]$ coordinates, where $\rho = [1,0,0]^T$ is the cylinder radius, $\varphi = [0,1,0]^T$ is the azimuth angle, and $h = [0,0,1]^T$ is the height location of the point P .

It is important to know the transformation of the coordinate system from Spherical to Cartesian, Spherical to Cylindrical, and Cylindrical to Cartesian, when mapping 3D geometrical objects onto a 2D plane.

The following set of equations are used to convert from spherical to Cartesian coordinates:

$$x = r \sin \theta \cos \varphi \quad (\text{B.1})$$

$$y = r \sin \theta \sin \varphi \quad (\text{B.2})$$

$$z = r \cos \theta \quad (\text{B.3})$$

where, r is the radius of the sphere, θ is the colatitude angle, and φ is the longitude angle. This transformation is important when mapping a sphere to a 2D plane.

The following set of equations are used to convert from spherical to cylindrical coordinates:

$$\rho = r \sin \theta \quad (\text{B.4})$$

$$\varphi = \varphi \quad (\text{B.5})$$

$$h = r \cos \theta \quad (\text{B.6})$$

where, r is the radius of the sphere, θ is the colatitude angle, φ is the azimuth angle, ρ is the cylinder radius, and h is the height location of the point P . This transformation is important when mapping a sphere into a projection cylindrical plane.

The following set of equations are used to convert from cylindrical to Cartesian coordinates:

$$x = \rho \cos \varphi \quad (\text{B.7})$$

$$y = \rho \sin \varphi \quad (\text{B.8})$$

$$z = h \quad (\text{B.9})$$

where, ρ is the cylinder radius, φ is the azimuth angle, and h is the height location of the point P . This transformation is important when mapping a cylindrical plane into a projection 2D plane.

Appendix C - Software Application Representation

The figures below depict the user interface of the software application developed as application of this thesis. This software is a part of the *Automated Percussive Riveting System* patent, and updated to version 3.0.16 as of August 29th, 2016 [10].

The user must have the RobotStudio™ software installed before installing the Riveting Planner® software. After the user install the executable installation package of the Riveting Planner®, the icon of the application will appear under the Add-Ins tab in RobotStudio™.

To launch the application, the user have to click the Riveting Planner® application icon as shown in Figure C-1, and the following main window will appear.

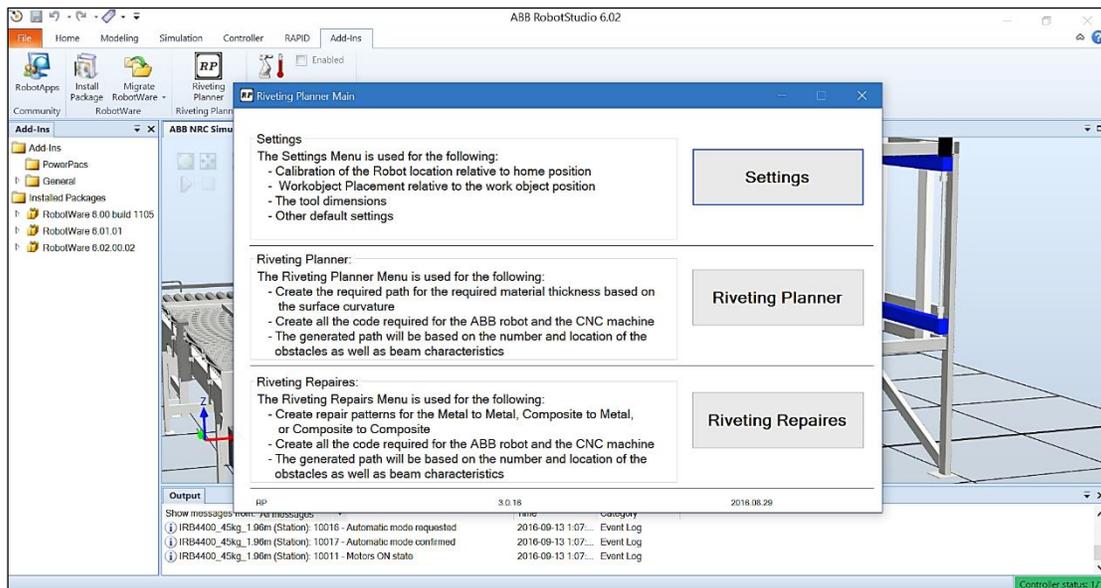


Figure C-1: Main window of the Riveting Planner® Software Application Add-in in RobotStudio™

This main window have the following three options: settings, Riveting Planner, and Riveting Repairs. The user have to set all the required settings before the user can start using the application. This settings window includes all the required calibration variables to ensure that the generated scripts are accurate and in the global frame reference.

The settings window includes the ABB Calibration tab, CNC Calibration tab, Tool dimension tab, and Defaults. All the settings parameters are stored once the user input them.

The ABB Calibration tab (Figure C-2) has all the required input to properly depict the position and orientation of the robot relative to the global frame of reference. This tab captures the position and orientation of the tool's tip, workpiece pilot point from the robot perspective. It also capture the various speed needed to accomplish some of the tasks.

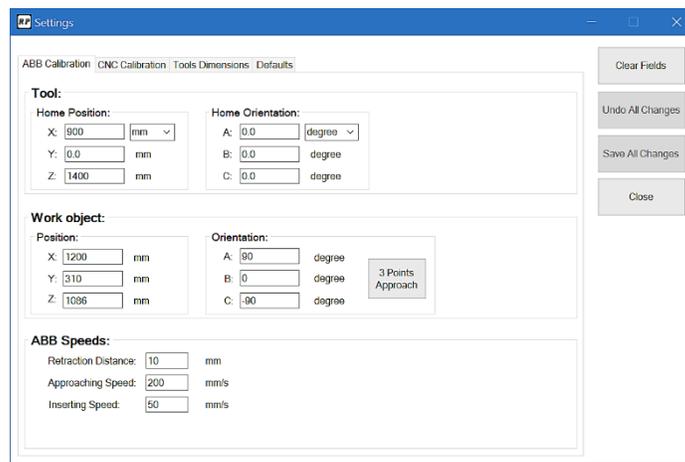


Figure C-2: Settings and Calibration Window

The three points approach will launch the window below (Figure C-3) that detect the position and orientation of the workpiece relative to the robotic global frame reference as covered in this thesis.

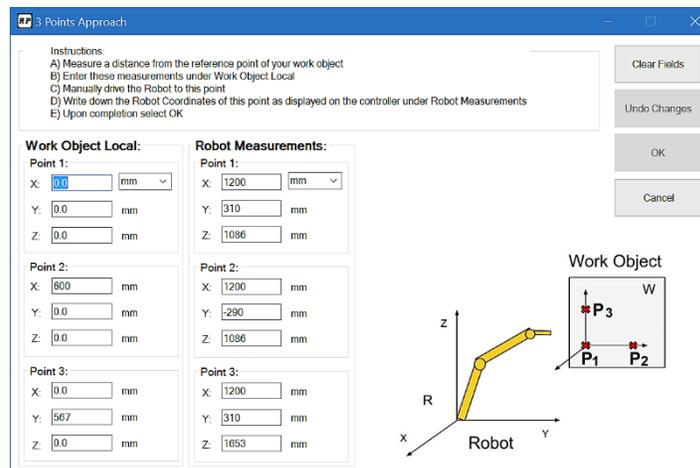


Figure C-3: Three points Approach Calibration Window

Next tab is the CNC Calibration settings (Figure C-4), this tab detect the required corrections to locate the home point position and orientation as well as the CNC required motion speed.

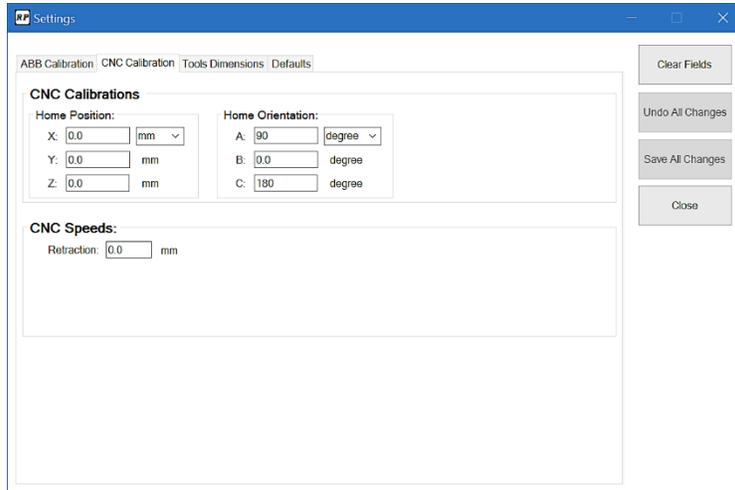


Figure C-4: CNC Calibration Window

The following tab is the Tools Dimensions tab (Figure C-5). This tab depict the size of the drilling gun, the sealing gun, and the riveting gun. These dimensions can be added, edited, or removed.

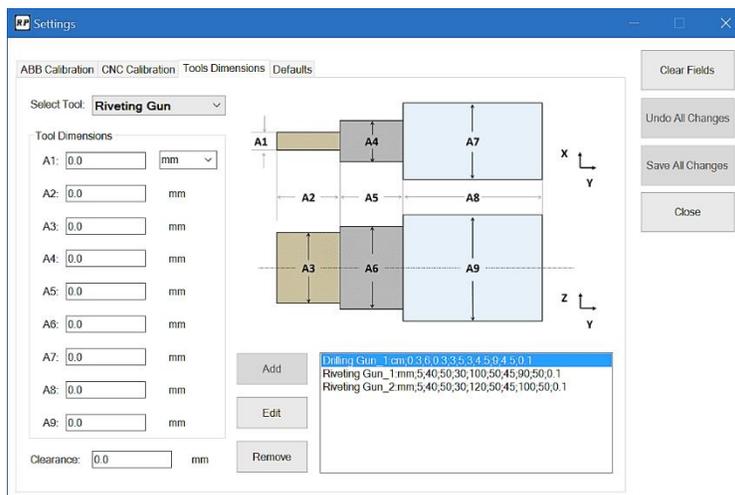


Figure C-5: Tools Dimensions Window

Once the user is satisfied that all settings are set properly, the user can click save all changes, and all the entered settings will be stored presently until modified by the user at any time.

The next button in the main window is the Riveting Planner button, which launches the Riveting Planner window (Figure C-6). This window has the tool used to perform the task, the required workpiece dimensions, and all the obstacles available on the surface of the workpiece as well as the results window.

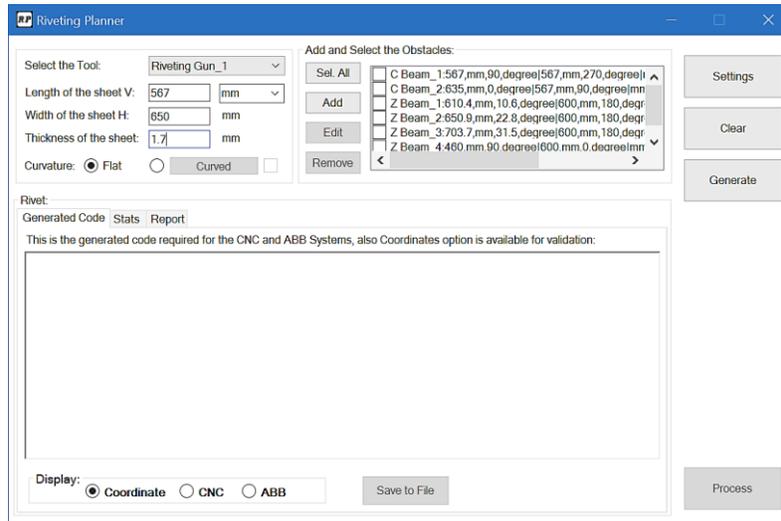


Figure C-6: Riveting Planner Window

If the workpiece is not flat, when selecting the curved selection, the following curvature input window will show up (Figure C-7).

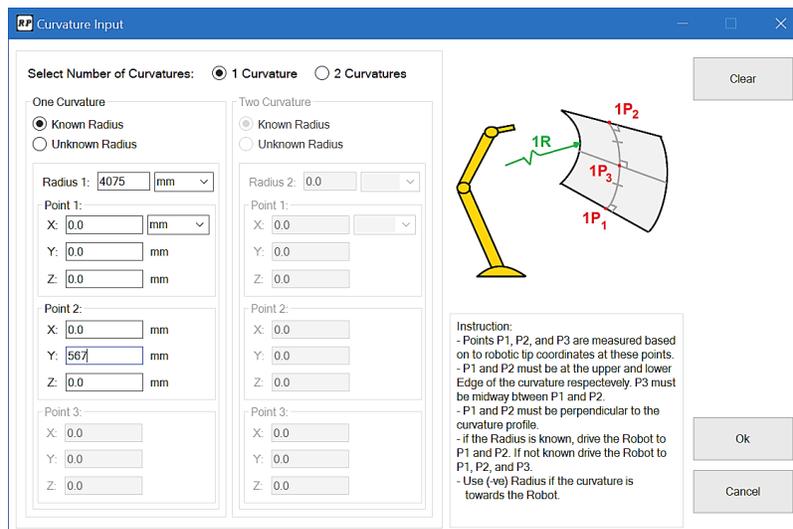


Figure C-7: Curvature Input Window for One Curvature Selection

If the surface of the workpiece has two curvatures, the curvature input window will enable the two curvatures input (Figure C-8). There are two options for the input, if the radius of curvature is known then the radius along with two points coordinates are added, else the robot has to reach 3 points on the workpiece on a vertical and horizontal arc of curvatures.

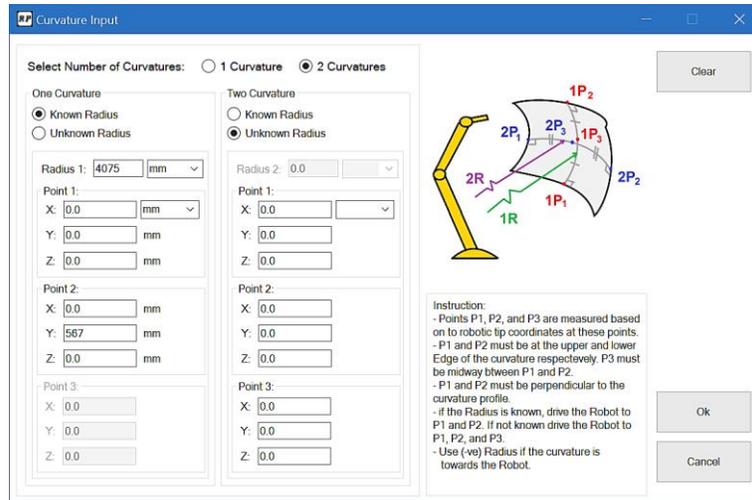


Figure C-8: Curvature Input Window for Two Curvature Selection

Next, when adding or editing an obstacle, the following obstacles window will open (Figure C-9). This window collect the dimensions of all the obstacles available on the surface of the workpiece such as I-beam, Z-beam, and other types of obstacles that affect the accessibility.

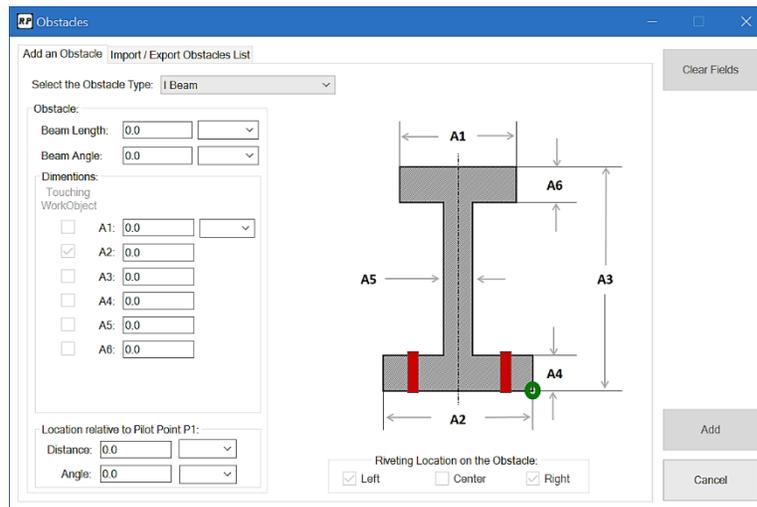


Figure C-9: Obstacles Window

Once all the inputs are entered, the user can select the generate button to generate the results of the riveting positions, for the ABB, CNC and normal Cartesian coordinate system in global frame of reference as shown in Figure C-10.

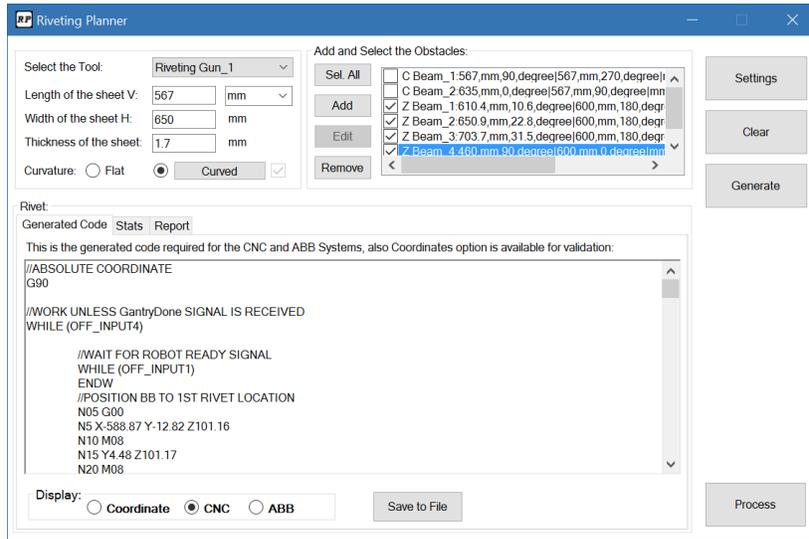


Figure C-10: Riveting Planner Window with Results

The last button on the Riveting Planner[®] main window, is the riveting repairs window (Figure C-11). This option is used when conducting patch repairs to generate the required riveting pattern using a process similar to the riveting planner option.

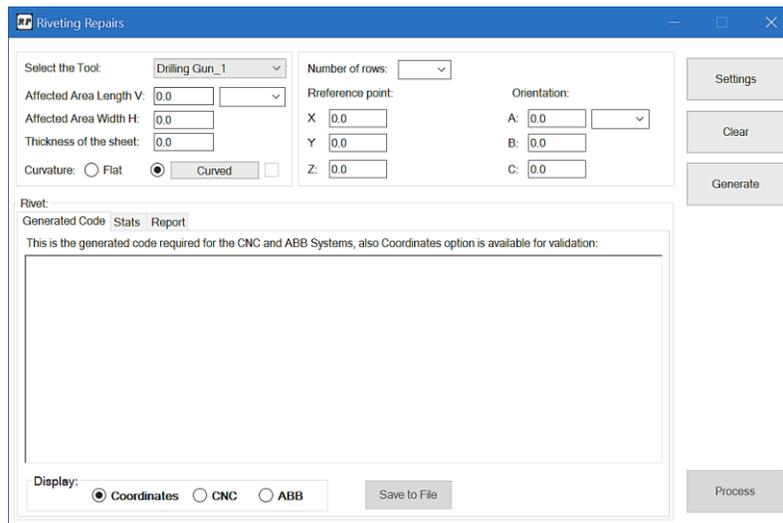


Figure C-11: Riveting Repairs Window

Appendix D - Robot Range Information

The Table D-1 and Table D-2 below depict the extreme positions and orientations of each link set by the manufacturer of the ABB Robot to prevent mechanical damage to the robotic joints, links and actuators.

Table D-1: The Extreme Positions of the Robot Arms IRB 4400 45 kg [48]

Pose no.	Position X (mm)	Position Z (mm)	Angle Axis 2 (degrees)	Angle Axis 3 (degrees)
0	1080	1720	0	0
1	887	2140	0	-30
2	708	836	0	65
3	1894	221	95	-60
4	570	-126	95	40
5	51	1554	-70	40
6	227	1210	-70	65

Table D-2: Types and Ranges of the Robot Motion in Every Axis [48]

Location of motion	Type of motion	Range of motion
Axis 1	Rotation	+165° to -165°
Axis 2	Arm	+95° to -70°
Axis 3	Arm	+65° to -60°
Axis 4	Wrist	+200° to -200°
Axis 5	Bend	+120° to -120°
Axis 6	Turn	+400° to -400°

Appendix E - Engineering Drawings of the Robot

The Figure E-1 and Figure E-2 below depict the operational range set by the manufacturer of the ABB Robot to prevent mechanical damage to the robotic joints, links and actuators

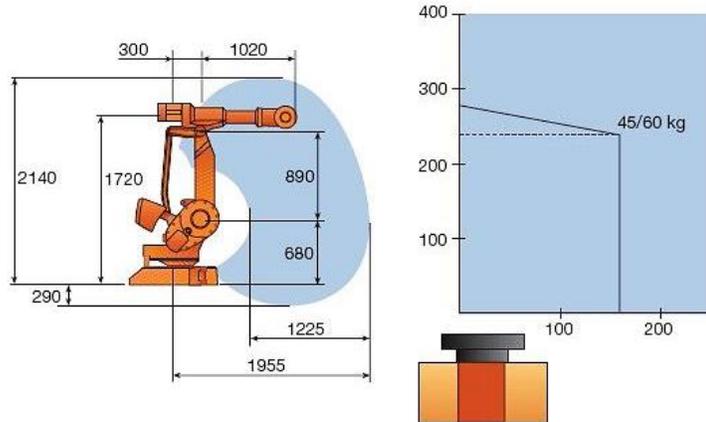
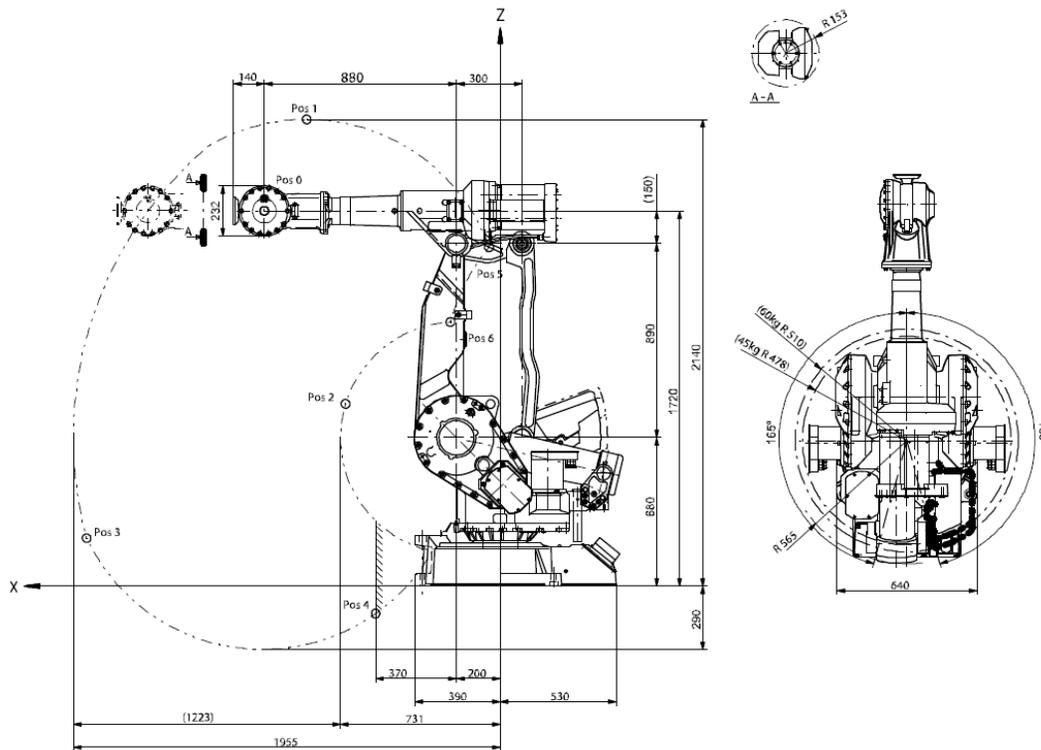


Figure E-1: Working Range and Load Diagram for the Robot Arms IRB 4400 45 kg [48]



xx0300000241

Figure E-2: Illustration of Extreme Positions of the Robot Arms IRB 4400 45 kg [48]

Appendix F - Mathematical Operations

A standard cube representation can be depicted in unit vector representation as:

$$Vertices_{cube} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}^T \quad \text{(F.1)}$$

REFERENCES

- [1] F. Xi, "History of the Machine," in *Computational Dynamics*, Toronto, Ryerson University, 2009, pp. 2-3.
- [2] M. Groover, *Fundamentals of Modern Manufacturing*, 4th ed., Wiley, 2007.
- [3] F. Xi, L. Yu and X.-W. Tu, "Framework on Robotic Percussive Riveting for Aircraft Assembly Automation," *Springer*, p. 112, 6 April 2013.
- [4] B. Monsarrat, E. Lavoie, G. Cote, M. De Montigny, C. Corbeil, D. Grenier and X. Tu, "High Performance Robotized Assembly System for Challenger 300 Business Jet Nose Fuse Panels," *Proceedings of the SAE AeroTech Congress and Exhibition*, pp. 12-17, Sept. 2007.
- [5] S. Peng, "Characterization and Ergonomic Design Modifications for Pneumatic Percussive Rivet Tools," *International Journal of Industrial Ergonomics*, vol. 13, no. 3, pp. 171-187, 1994.
- [6] Y. Li, F. Xi and K. Behdinan, "Dynamic Modeling and Simulation of Percussive Impact Riveting for Robotic Automation," *Journal of Computational and Nonlinear Dynamics*, vol. 5, no. 021011-1, 2010.
- [7] "Robot selector," ABB Group, 2013. [Online]. Available: <http://new.abb.com/products/robotics/robot-selector>. [Accessed 01 06 2013].
- [8] D. Dakdouk, "Design of Support Systems for Percussive Robotic Riveting," Ryerson University, Toronto, April, 2012.
- [9] Y. M. Zhao, Y. Lin, F. Xi and S. Guo, "Calibration-Based Iterative Learning Control for Path Tracking of Industrial Robots," *IEEE Transactions on Industrial Electronics*, vol. Vol. 62, no. No. 5, p. 2921, MAY 2015.
- [10] F. Xi, Y. Lin, D. Dakdouk, M. Helal and B. East, "Automated Percussive Riveting System". Canada Patent WO 2013152440 A1, 15 April 2013.

- [11] Q. Peng and X. Kang, "Tool Feasibility Analysis for Fixture Assembly Planning," *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 130, no. 4, p. 9, 2008.
- [12] Q. Peng and C. Chung, "Tool selection-embedded optimal assembly planning in a dynamic manufacturing environment," *Computer-Aided Design*, vol. 41, no. 7, pp. 501-512, 2009.
- [13] Q. Peng and C. Chung, "A novel approach to the geometric feasibility analysis for fast assembly tool reasoning," *International Journal of Advanced Manufacturing Technology*, vol. 31, pp. 125-134, 2006.
- [14] N. Jayaweera and P. Webb, "Automated Assembly of Fuselage Skin Panels," vol. 27, no. 4, pp. 343-355, 2007.
- [15] D. E. Ltd., "Riveting & Drilling - Selection Guide," DOBCO, Mississauga, 2012.
- [16] A. M. P. Inc., "Aircraft Rivets and Special Fasteners," in *Aircraft Sheet Metal Construction and Repair, Training Course*, Himalayan Books, 1983, pp. section 3: Pages 13-36.
- [17] T. Lozano-Péres, "Spatial Planning: A Configuration Space Approach," *IEEE Transaction on Computers*, Vols. c-32, no. 2, pp. 108-120, 1983.
- [18] T. C. Woo and L.-L. Chen, "Computational Geometry on the Sphere With Application to Automated Machining," *Journal of Mechanical Design*, pp. 114:288-95, 1992.
- [19] A. J. Spyridi and A. A. Requicha, "Accessibility analysis for the automatic inspection of mechanical parts by coordinate measuring machines," *IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1284-9, 1990.
- [20] T. C. Woo and L. L. Chou, "Path Detection for Mould and Die Design," *Computer Aided Design*, vol. 25, pp. 763-767, 1993.

- [21] S. Dhaliwal, S. Gupta, J. Huang and A. Priyadarshi, "Algorithms for Computing Global Accessibility Cones," *Journal of Computing and Information Science in Engineering*, vol. 3, pp. 200-209, 2003.
- [22] K. Suthunyanakit, E. Bohez and K. Annanon, "A new global accessibility algorithm for a polyhedral model with convex polygonal facets," *Computer-Aided Design*, vol. 41, pp. 1020-1033, 2009.
- [23] Q. Peng and C. Chung, "Analysis of Part Accessibility in Product Disassembly," *Journal of Computer-Aided Design and Applications*, vol. 4, no. 5, pp. 695-704, 2007.
- [24] M. C. Niu, *Airframe Structural Design, Practical Design Information and Data on Aircraft Structures*, 2nd Ed. (4th published) ed., Hong Kong: Hong Kong Conmilit Press Ltd., 2011.
- [25] E. Petriu, "Manipulator Arms," in *Robotics - Introduction*, Ottawa, University of Ottawa, 2001, p. 6.
- [26] G. Carbone and F. Gomes-Barvo, *Motion and Operation Planning of Robotic Systems: Background and Practical Approaches*, Frosinone, Italy: Springer, 2015.
- [27] P.-Y. Zhang, T.-S. Lü and L.-B. Song, "Soccer robot path planning based on the artificial potential field approach with simulated annealing," *Robotica*, vol. 22, pp. 563-566, 2004.
- [28] H. Jang, H. Moradi, S. Lee and J. Han, "A Visibility-based Accessibility Analysis of the Grasp Points for Real-time Manipulation," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. pp. 3111 - 3116, 2005.
- [29] ABB Group Ltd., "RobotStudio Development," 2008. [Online]. Available: <http://developercenter.robotstudio.com/Index.aspx?DevCenter=RobotStudio&OpenDocument&Url=html%2f08a465af-bdcc-4f65-a647-764c14005d00.htm>. [Accessed 24 Feb 2012].
- [30] R. Paul, *Robot Manipulators: Mathematics, Programming, and Control*, Massachusetts: The Massachusetts Institute of Technology, 1981.

- [31] "Chapter 4: Aircraft Metal Structural Repair," in *Aviation Maintenance Technician Handbook - Airframe*.
- [32] K. D. Nguyen;, T.-C. Ng and I.-M. Chen, "On Algorithms for Planning S-curve Motion Profiles," *IN-TEH, International Journal of Advanced Robotic Systems*, vol. 5, no. 1, pp. 99-106, 2008.
- [33] "Chapter 5. Custom Profiling," in *6K Series Programmer's Guide*, Rev B ed., Parke Mmotion, 2003, pp. 119-163.
- [34] J. Mortimer, "Jaguar 'Roadmap' Rethinks Self-Piercing Technology," *Industrial Robot: An International Journal*, vol. 32, no. 3, pp. 209-213, 2005.
- [35] MTorres, "Specials Solutions: Torres Riveting," MTorres Diseños Industriales, 04 03 2010. [Online]. Available: <http://www.mtorres.es/en/aeronautica/metalicos/special-solutions>. [Accessed 01 06 2013].
- [36] MTorres, "Flexible Drilling Head," MTorres Diseños Industriales, 2015. [Online]. Available: <http://www.mtorres.es/en/aeronautics/products/metallic/flexible-drilling-head>. [Accessed 04 02 2016].
- [37] "Robot Solutions," Gemcor Dirvmatic Fastening Systems, 2016. [Online]. Available: The KUKA Multi-Functional End Effector (MFEE)TM Boing. [Accessed 01 2016].
- [38] E. Stansbury, B. Bigoney and R. Allen, "E7000 High-Speed CNC Fuselage Riveting Cell," SAE, 2012.
- [39] "Aerospace Automation," KUKA, 2016. [Online]. Available: <http://www.kukaaero.com/capabilities/aerospace-automation>. [Accessed 01 2016].
- [40] "Bombardier to Combine Efficiency and Quality in Manufacturing of C Series Aircraft," Bombardier Aerospace, 11 5 2011. [Online]. Available: <http://www.bombardier.com/en/media/newsList/details.636-bombardier-to-combine-efficiency-and-quality-in-manufacturing-of-cseries-aircraft.bombardiercom.html>. [Accessed 01 2016].

- [41] T. Gray, D. Orf and G. Adams, "Mobile Automated Robotic Drilling, Inspection, and Fastening," *SAE International*, 09/17/2013.
- [42] A. Group, "The Airbus Shopfloor Challenge at ICRA 2016," Airbus Group, 01 Mar 2016. [Online]. Available: <http://www.airbusgroup.com/int/en/people-careers/Working-for-Airbus-Group/Airbus-Shopfloor-Challenge-2016.html#>. [Accessed 18 May 2016].
- [43] R. Henrion, D. Homberg and C. Landry, "Path Planning And Collision Avoidance For Robots," *Numerical Algebra, Control and Optimization*, vol. 2, no. 3, pp. 437-463, Sept 2012.
- [44] H. Goldstein, in *Classical Mechanics*, 2nd ed., Reading, MA, Addison-Wesley, 1980, pp. 146-147 and 608.
- [45] G. Arfken, in *Mathematical Methods for Physicists*, 3rd ed., Orlando, FL, Academic Press, 1985, pp. 199-200.
- [46] F. Xi, D. Nancoo and G. and Knopf, "Total Least Squares Methods for Active View Registration of 3-D Line Laser Scanning Data," *ASME Journal of Dynamic Systems, Measurement, and Control*, pp. 50-56, March 2005.
- [47] ABB Group Ltd., "Orient - Orientation," in *RAPID reference manual - part 2, Functions and data types A-Z*, 2011, pp. 407-409.
- [48] "ABB Product Guide: IRB 4400," ABB Group, 2013. [Online]. Available: <http://www.abb.ca/product/seitp327/d891cb5c242fe889c1256efc002871f3.aspx>. [Accessed 12 06 2013].
- [49] F. Xi, "Materials and Manufacturing (AER50)," Ryerson University, Toronto, 2010.
- [50] J. Schwaner, "Hardware: Rivet Edge Distance," Mechanic Support, 2010. [Online]. Available: http://www.mechanicsupport.com/Rivet_Edge_Distance.html. [Accessed 08 08 2013].
- [51] H. Ghaemi, "Introduction to Computational Dynamics," Ryerson University, Toronto, 2013.

- [52] "Rotations and Euler angles," easyspin.org, 2010. [Online]. Available: <http://www.easyspin.org/documentation/eulerangles.html>. [Accessed 12 Apr 2012].
- [53] "The Revolution Riveter," Superior Rivet-Machines, 2012. [Online]. Available: <http://www.rivet-machines.com/Products/Impact-Riveting.aspx>. [Accessed 01 06 2013].
- [54] R. Khurmi and J. Gupta, A Textbook of Machine Design, 14th ed ed., New Delhi: S Chand & Co Ltd, 2005.
- [55] A. Hourtash and M. Tarokh, "Manipulator Path Planning by Decomposition: Algorithm and analysis," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 100-115, 2001.
- [56] C. Qin and D. Henrich, "Path Planning for Industrial Robot arms: A Parallel Randomized Approach," in *In Proc. of the International Symposium on Intelligent Robotic Systems (SIRS96)*, Lissabon, 1996.
- [57] Harsle, "Z3060 Radial Drilling Machine," Nanjing Harsle Machine Tool Co., Ltd., 2015. [Online]. Available: http://www.harsle.com/e_products/show/?163-Z3063-Radial-drilling-machine-163.html. [Accessed 04 02 2016].