7-12-2022

CONTROL AND PLANNING FOR MOBILE MANIPULATORS USED IN LARGE SCALE MANUFACTURING PROCESSES

Joshua T. Nguyen
Louisiana State University and Agricultural and Mechanical College

CONTROL AND PLANNING FOR MOBILE MANIPULATORS USED IN LARGE SCALE MANUFACTURING PROCESSES

A Thesis

Submitted to the Graduate Faculty of the
Louisiana State University
and Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Master of Science

in

The Department of Mechanical Engineering

by
Joshua Thy Nguyen
B.S., Louisiana State University and Agricultural and Mechanical College, 2020
August 2022

Acknowledgments

I would like to personally thank Corina Barbalata for advising and teaching me during my graduate program. I would also like to thank Marcio de Queiroz and Hunter Gilbert for being part of my advisory committee and for also providing me teachings over the years during both my graduate and undergraduate curriculum. I am grateful for all the past mentors, teachers, and advisors I have had throughout the years, specifically: Ignacio Carlucho, Tryfon Charalampopoulos, Elizabeth Melvin, Boz Bowles, Adrienne Steele, and Warren N. Waggenspack. I would also like to lastly thank all of my lab mates in PFT 2254, friends, and family members.

Table of Contents

Abstract

Sanding operations in industry is one of the few manufacturing tasks that has yet to achieve automation. Sanding tasks require skilled operators that have developed a sense of when a work piece is sufficiently sanded. In order to achieve automation in sanding with robotic systems, this developed sense, or intelligence, that human operators have needs to be understood and implemented in order to achieve, at the minimum, the same quality of work. The system will also need to have the equivalent reach of a human operator and not be constrained to a single, small workspace. This thesis developed solutions for a control scheme and a path planning algorithm to provide the next steps into sanding automation. The control scheme uses found insights on how vibration forces evolve over time during sanding operations to estimate the quality of the surface and adapt the velocity of a sander, akin to how human operators do, with a robotic manipulator. The path planning algorithm was developed to allow for the use of mobile manipulators to perform the sanding tasks and giving the manipulator an equivalent reach to that of an operator.

Chapter 1. Introduction

1.1. Motivation

Finishing operations, such as trimming, surface treatment, grinding, sanding, and painting, are necessary to ensure components possess required geometrical tolerances and properties [1]. Sanding and polishing represent two of the most common finishing processes, both of which are processes that remove small amounts of material on a components surface [2]. As presented in [3] the sanding process is common in furniture manufacturing, where it is used to remove residuals caused by wood processing to ensure homogeneously smooth parts. These operations can be costly and time-consuming [4]. Furthermore, for human operators, those tasks are tedious, leading to physical discomfort and repetitive strain injuries, risks that could be mitigated by an automated solution [5]. Robotic systems represent a solution that both decrease cost and improve the sanding process. The state-of-the-art in robotic sanding technology has followed two avenues: 1) autonomous sanding where the path and the velocity along the path are predefined or 2) teleoperated robotic sanding that leverages the knowledge of a human operator to monitor and intervene in the process described in the previous point and remove individual flaws [6]. Nevertheless, these approaches have limited applicability for wide deployment as they rely on human-expertise. To enable fully autonomous robotic systems to be deployed in sanding operations, the following barriers have to be addressed: 1) the motions for sanding cannot be planned prior to the sanding process, as residual stresses during manufacturing may cause dimensional variations in the final parts geometry by up to several centimeters [7], 2) both the nature and duration of the tasks to be performed vary from one part to another, 3) task completion is based on a human operators judgement and previous experience, which is difficult to automate, and 4) the robotic system must be able to maneuver

and sand objects within the same or higher reach of a human operator. To address these barriers, in this work we design a smart framework for achieving variable velocity motion of the robotic manipulator based on surface characteristics. The main contributions of this framework are: 1) observe a correlation between vibration information from a force/torque sensor and the surface quality, 2) design of an expert system for sanding operations without human-input, 3) enable sanding operations without the need of specialized sensors for measuring roughness, using only sensors that are readily available and already a part of a robotic system and 4) design a path planning system for the use of mobile manipulators to enable sanding operations over large areas. Hence, the proposed architecture will enable robotic manipulators with specialized sanding tools to perform smart sanding without human input, using information only from the sensors integrated in the robotic manipulator.

1.2. Chapter Overview

The following sections in this thesis are as follows. Chapter 2 discusses the background information of this thesis and covers past/related work and the contributions this thesis holds. Chapter 3 goes over the first part of this work, a control algorithm for automated sanding using a robot manipulator. Chapter 4 goes over the second part of this work, path planning for a mobile manipulator whose end-effector must follow a desired trajectory. Chapter 5 concludes this work and suggests ideas for future work given the findings.

Chapter 2. Background

2.1. Literature Review

Work on autonomous robotic sanding operations is limited and currently in early stages. A majority of the work has been focused on force calculations and tracking needed for sanding for a single robot manipulator and path planning for sanding tools along some given work surface. Few works have studied the use of robotic systems that encompass the use of multiple sensors for a more realized autonomous robotic sanding system. The use of a mobile manipulator for sanding is also lacking although there are works that use mobile manipulators for similar applications. This section is broken down into three sections: literature review on 1) robotic sanding systems, 2) control for autonomous sanding, and 3) similar works on path planning using mobile manipulators.

2.1.1. Review on Robotic Sanding Systems

In [8] and [9] an acoustic emission sensor is used to estimate the surface roughness, tool wear, and to adjust the polishing force and tool speed accordingly. The acoustic measured were the sounds produced during the sanding operation when the tool is actively sanding a surface. The author here first produces a database of experiments to build the relationship between the acoustics and the desired parameters then uses the database for online adjustments of sanding parameters to achieve a higher surface quality during sanding operations.

The work in [4] analyzed the feasibility use of an orbital sander attached to the end of a robotic manipulator for automated sanding and polishing of metal surfaces for aircraft manufacturing and found that the automated process was 25% more efficient than the typical manual process. [10] developed grinding strategies for a robotic manipulator equipped with an orbital sander. The system began with set parameters of a desired geometry and applied

normal force of the sander onto the work surface. The system monitored and ensured fitting of multiple parameters during sanding operations. Experiments showed that the system was able to perform more efficiently when compared to the manual process.

[5] and [11] study the combination of a griding/polishing wheel and a robotic manipulator. Both papers focus on manipulator path planning for grinding/polishing of objects with complex surfaces.

2.1.2. Review on Controls for Sanding

Initial studies in deploying collaborative robotic manipulators in finishing operations are presented in [12]. The goal of these works is to investigate both open- loop and closed-loop force control structures and their impact on the quality of finish. It was concluded that accurate force regulation improves the quality of the sanding, but an increase in the contact force does not ensure a better finish.

[13] uses a combination of a robot manipulator's joint torque input and momentum, read through joint encoders, with a Kalman filter to estimate the forces at the end effector without the need of a force-torque sensor. The author saw an improvement in the force estimation when compared to the momentum only estimation approach including model inaccuracies, indicating that the addition of a Kalman filter helps with the estimation. The author concluded that an improvement is made with a Kalman filter as it allows for model inaccuracies and compensates for them. Although this paper does not specifically focus on robot manipulator sanding, force estimation at the end-effector where the sander is equipped is seen as essential in robotic sanding applications [14]–[16].

[17] uses joint torque sensors to estimate the force at the end-effector. Unlike in [13], this work performs additional signal analysis only on the joint torque signal. Due to vibratory nature of sanding operations, the author implements an adaptive notch filter whose frequency is dependent on a frequency analysis of the joint-torque signal. The filtered joint-torque signal is then used to estimate the end-effector force which is then implemented into a hybrid force-position controller. Experiments showed that the addition of the adaptive filter to counteract the vibration due to sanding allows for improved force tracking during sanding operations.

[18] presents a novel hybrid position-force controller that uses sliding mode theory/approaches to ensure for robustness and low computational costs to perform sanding operations with a 7 degree-of-freedom manipulator. The author shows that the controller is able to maintain a desired trajectory and force during polishing operations. [1] and [14] implement a fuzzy-logic controller to perform force tracking at the end effector for sanding operations. This has shown an improvement to force tracking compared to a typical PID controller.

[19] analyzes the force signal for robot assisted polishing for surface estimation. The author performed experiments where force measurements during polishing operations were made and analyzed. The author in this work concluded that there is an evolution in the force signal during polishing while the surface becomes smoother. The author then suggested that a control algorithm using this information could be further developed for use in automated polishing processes.

2.1.3. Review on Mobile Manipulator Path Planning

The concept of attaching a robot manipulator to a mobile base is called simply, a mobile manipulator [20]. Mobile manipulators present itself today with many challenges. Since mobile manipulators consists of both a robot manipulator and a mobile base, it inherits the common challenges of those two robotic systems. Not only that, but mobile manipulators must also deal with the coupling of the two systems in which both systems must be coordinated to work desirably.

For sanding operations in industry, the path that a sander must cover is pre-determined. This is due to the fact that the sander must cover an entire area and there are works that go over coverage path planning [5][21][22][23]. In order for a mobile manipulator system to perform sanding operations, the mobile manipulator's end effector must follow this predetermined path and the joint paths of the mobile manipulator needs to be planned such that the end effector can achieve this. To ensure that the system's path is collision free, whole-system coordination is required, which can be difficult as mobile manipulators tend to be highly redundant-systems with infinite reach. [24] identifies this problem as Motion Planning along End-effector Paths (MPEP). [20] provides a systematic review on path planning for mobile manipulators and includes a section on mobile manipulator path planning where the end-effector must follow a given trajectory.

Path planning for mobile manipulators is typically broken into two categories, a decoupled approach in which path planning for the mobile base and the arm are done separately, and a coupled approach in which they are performed together [20], [24]. To clarify, the former approach works by moving the base into a desired position while keeping the

manipulator in a default state then moving the manipulator once the base kept at a position. In order to achieve a solution to MPEP, a coupled approach must be taken, in which the whole system's configuration path must be planned.

After deciding the initial approach of a coupled or decoupled approach to the path planning problem, generating the joint path for the mobile manipulator is also broken up into multiple approaches. The two main categories of approaches to generating joint paths are optimization-based path planning and sampling-based path planning, although other methods do exist [20], [25]. Each approach has their pros and cons.

Optimization-based methods are capable of finding locally optimal paths and incorporate a multitude of parameters, such as smoothness or distance from obstacles. They also tend to be computational expensive and may not find converge to a solution. Sampling-based methods are computational fast and can guarantee solutions, depending on the algorithm used, but the path tend to be jerky and discontinuous.

[26] performs path planning for a mobile manipulator with a 7 degree-of-freedom arm to perform 3-D printing. In this work, the end effector must maintain a specific trajectory, similar to the task of sanding, in which the end-effector must also maintain a specific trajectory. This author uses an RRT* (Rapidly-exploring Random Tree Star) algorithm to generate the path of the mobile manipulator's joints.

2.2. Thesis Contribution

2.2.1. Vibration Controller & Path Planner

The contribution of this thesis are: (1) a novel control algorithm that can be used for automated sanding for a robot manipulator and (2) a path planning algorithm for mobile

manipulators whose end-effector pose is constrained to a specified trajectory. The control algorithm developed provides an industry ready solution for intelligent sanding operation using robots. An analysis on the force generated at the end-effector during sanding was also performed and key insights about how force vibrations evolve during sanding was made. Experiments were made and showed that the novel control algorithm performed better than other algorithms currently in use. The path planning algorithm created investigates the use of probabilistic path planning methods for redundant, high degree-of-freedom mobile manipulators with end-effector trajectory constraints. This thesis ultimately provides the first steps to realizing fully autonomy in large industry sanding application.

Chapter 3. Vibration Controller

3.1. Introduction

Sanding operations is one of the last few manufacturing operations that is struggling to be autonomized. A requirement of robotic automation is that the robotic system needs to work as intelligently as humans while also being affordable and practical in a manufacturing sense. The work in [19] shows the methods that human operators employ during sanding but analyzing the forces that they feel during operations. The author concludes that these forces evolve as a correlation to a surface's roughness over time and this idea is used as motivation for this vibration controller. As shown in the literature review section, many of the robotic systems presented are not ready for full autonomous deployment as they are either not suitable for a manufacturing environment or still require human assistance. For the control algorithms presented, the common focus is only force regulations strategies using differing methods and sensor information The system presented in this section equips a robot manipulator, commonly used in industry, with an orbital sander. The robot manipulator contains a force torque sensor at the end-effector, where the sander is attached. Figure 3.1 below shows the manipulator setup used. This system is low-cost and performs surface quality estimation based on the force vibration readings taken with a force-torque sensor during sanding operations to perform intelligent sanding akin to expert human operators.

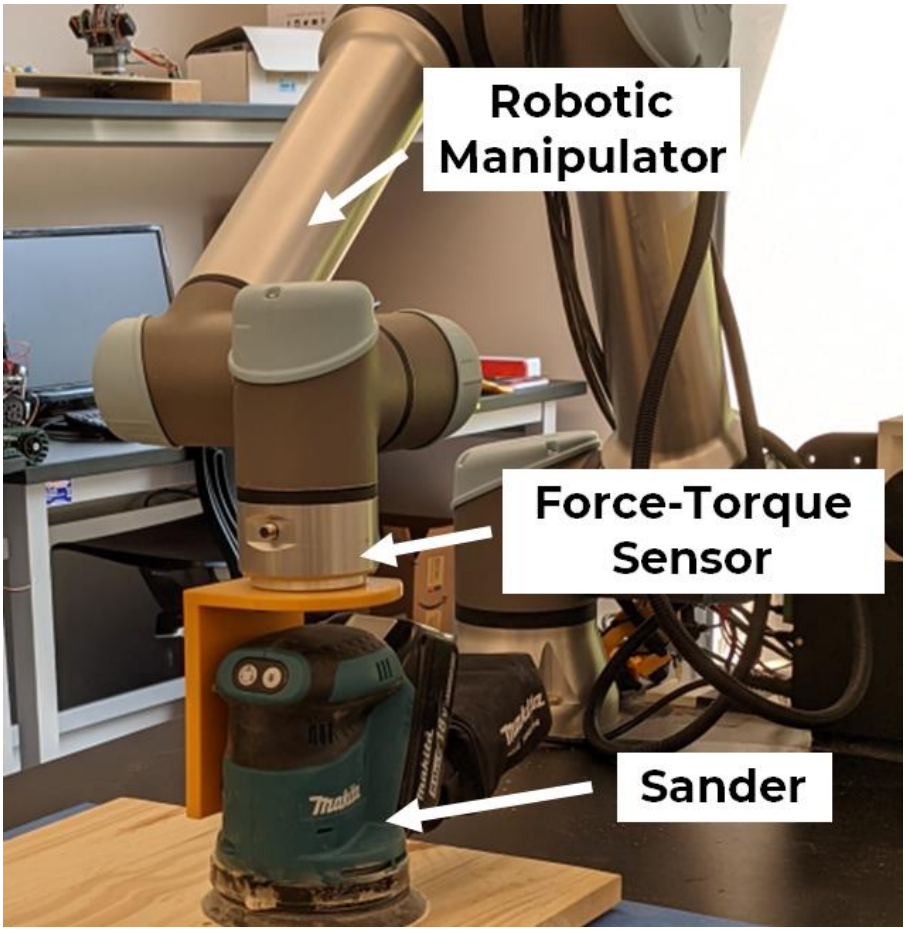

Figure 3.1. Robot manipulator with orbital sander used for experimentation

3.2. Methodology

3.2.1. Initial Experimentation to Correlate Surface Roughness to Vibration Readings

Force measurements taken at the end-effector, with an integrated force-torque sensor, during the sanding process are inherently noisy due to the vibrations caused by the sanders motor and the rough surface that is being sanded [17]. Because of this, it is recommended to look at the force signal in the frequency domain. This is done by transforming the force signal using the fast-Fourier transform (FFT) [27]. An example of the FFT graph produced during sanding can be seen in Figure 3.2. When the sander is turned on, regardless if the tool is in the air or in contact with the environment, vibrations are present. The frequency of these

vibrations depends on both the tool and the working surface if the tool is in contact with the surface. As the sander makes contact with the surface, the vibration frequency is dampened while the magnitude of the vibration increases. As the surface gets smoother, the dampening effect decreases, and the amplitude begins to diminish. This behavior has been observed across several experiments, where the robotic arm was commanded to sand a singular area while maintaining a normal force of 15 newtons for 50 seconds. The average value and standard deviation of the sanding frequency and its amplitude have been computed across all these experiments. As seen in Figure 3.3, these observations validate the assumption that sanding frequency increases and the amplitude decreases over time, indicating a less rough surface. Furthermore, similar behavior has been shown in [19] where a relationship between surface roughness of an alloy steel and force data coming from gauge sensor was found. The study shows that the frequency analysis of the force data could be potentially used to estimate the surface roughness, however, the study does not utilize this information for online estimation of

the roughness. It is this relationship that allows for an online observation of surface roughness and what is used to establish governing control laws for the proposed expert system.

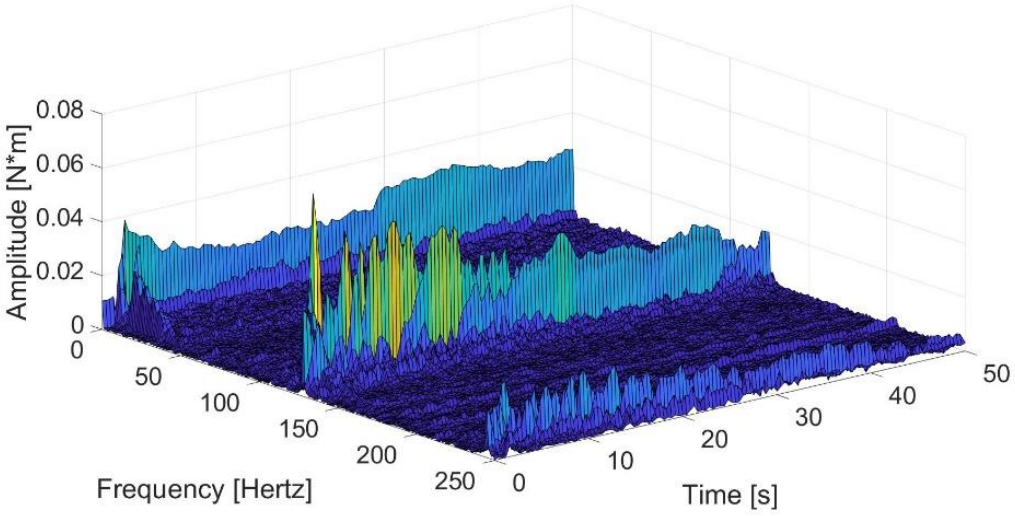

Figure 3.2. FFT of the measured normal torque when the sanding tool is turned on

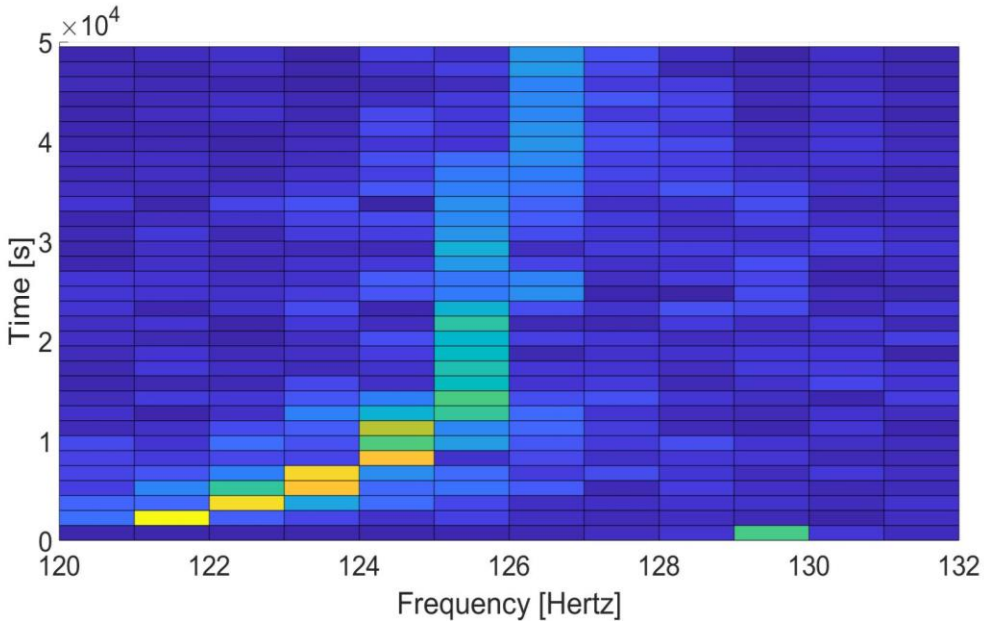

Figure 3.3. Top-down view heat map of the frequency response around the vibration frequency. The amplitude of this vibration is high, and the frequency is dampened initially and regresses over time.

3.2.2. Control System Design

3.2.2.1 Coordinate Frame Definition

The proposed expert system is designed based on the local coordinate system of the working surface. In this paragraph we describe this coordinate system, known as the n-t coordinate system or the path coordinate system, and its usage for the design of the proposed architecture. When the tool of the manipulator moves along a path (curved or straight), it can be convenient to describe its motion using the normal, tangential, and bi- normal coordinate system, also known as n-t coordinate system. We define a path coordinate system where the tangent unit vector, *t*, maintains a direction along the path on the surface while the normal unit vector, *n*, is aligned with the normal of the surface at the current location of the tool. The bi-normal unit vector, b, is determined by the right-hand rule such that $b = t \times n$. Figure 3.4 shows the path coordinate system in the set-up of sanding tasks for flat surfaces. It has to be noted that the manipulator used in this research only accepts commands in its base coordinate frame. To accommodate this, the velocity signal that is developed in the path coordinate frame is transformed into the base coordinate cartesian frame, leading to the commanded end-effectors linear and angular velocities given by $v_c = \left[v_x, v_y, v_z, v_\psi, v_\theta, v_\psi\right]^T$. The transformation between the n-t coordinate system and the cartesian coordinate system is given by $\left[v_x, v_y, v_z, v_\psi, v_\theta, v_\psi\right]^T = R(\cdot) \otimes \left[v_t, v_n, v_b, v_\alpha, v_\beta, v_\gamma\right]^T$, where $R(\cdot)$ is the rotation matrix between the n-t coordinate frame and the cartesian frame. v_n is the velocity in the normal direction, set to a constant value and $v_t, v_b, v_\alpha, v_\beta, v_\gamma$ are the outputs of the expert system presented in the following paragraphs.

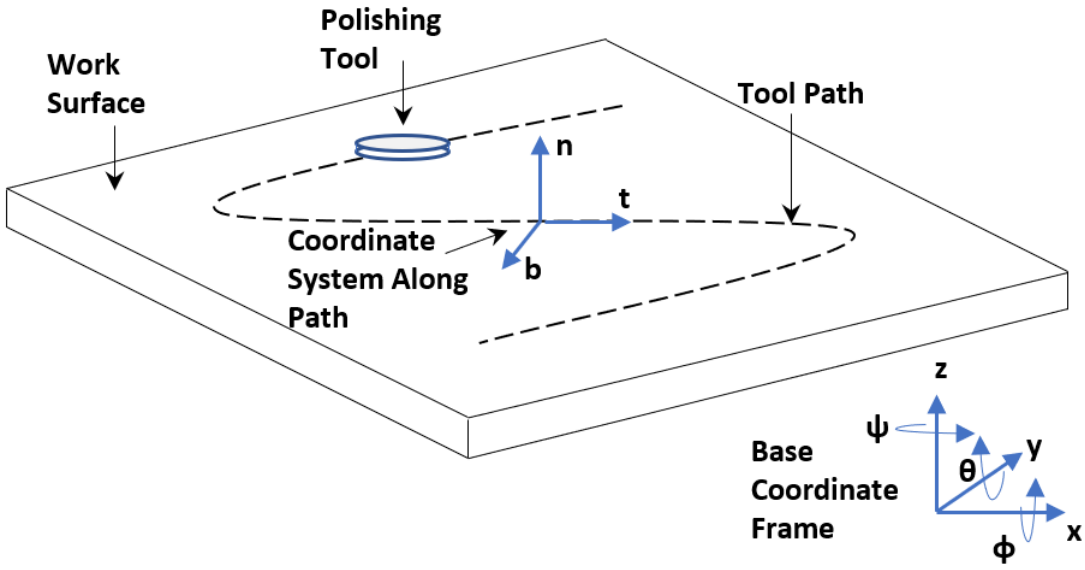

Figure 3.4. Motion of the sanding tool on the work surface, describing the coordinate systems used

3.2.2.2 Governing Control Laws

The proposed expert system architecture, seen in Figure 3.5, drives the velocity profile in the n-t coordinate frame. The system has two governing control laws: 1) a pose-driven control law and 2) a vibration- driven control law. The following sub-sections present these control laws and their formulations.

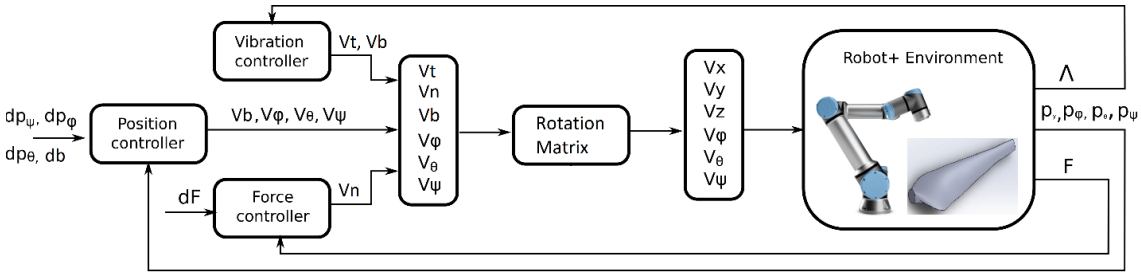

Figure 3.5. The proposed expert system: the vibration-driven control law acts in the tangent axes based on the FFT information, the pose- driven control law acts in the ϕ-, θ-, ψ- axes and in bi-normal axis, The rotation matrix transforms the n-t coordinate frame.

3.2.2.2.1 Vibration-Driven Control Law

The proposed vibration-driven control law is responsible for the definition of the tool's velocity in the tangential direction to the work piece along a given path. Based on the vibration-roughness observations presented in 3.2.1, the proposed vibration-driven control law output is inversely proportional to the amplitude and frequency information. Specifically, as the surface becomes smoother the velocity of the end-effector increases, as the robot should spend less time sanding those areas. The formulation of the vibration-control law is given by:

$$v_t = \frac{1}{u_\lambda + u_A + \epsilon}$$

$$u_\lambda = K_\lambda * e_\lambda, \quad u_A = K_A * e_A$$

$$e_\lambda = \lambda_m - \lambda_d, \quad e_A = A_d - A_m$$

Equation 1: Vibration Control Law

where ϵ is a small constant (i. e. 10−6) that ensures singularities are avoided when u_λ and u_A are zero. λ_d is the free spin sanding frequency that is dependent on the sanding tool used and found by allowing the sander to free spin in air. λ_m is the measured sanding frequency taken from the FFT. The measured sanding frequency is taken to be the frequency with the highest amplitude within a predefined range. The frequency range is determined empirically based on the properties of the work surface, sanding tool, and sandpaper grit size. A_d is the free spin amplitude of the sanding frequency that is dependent on the tool used. A_m is the measured amplitude of the sanding frequency determined above. e_λ represents the error in frequency domain and e_A is the error in amplitude, K_A is the normalization factor for the amplitude component of the controller, and K_λ is the normalization factor for the frequency

component. v_t is the commanded velocity in the tangent-axis. A large e_λ or e_A indicates a rough surface, meaning more sanding is required, the commanded velocity, v_t being low. If both errors are zero, the velocity becomes $\frac{1}{\epsilon}$ and the end-effector rapidly advances to the next rough area where errors become present. For implementation purposes, an upper limit is placed on the velocity to ensure safe practices. The magnitude of the frequency dampening effect and the amplitude increase depend on the sanding parameters. Equation (1) shows the proposed vibration-based control law which uses a combination of both the measurement of the dampening effect and amplitude increase. By combining the information provided by both parameters we ensure meaningful information is received by the expert system, achieving reliable sanding behavior. As presented in [19] this can happen for several materials including alloy steel. The control signals u_λ and u_A, are added together to ensure that all the information from the FFT is considered in a decoupled manner.

3.2.2.2.2 Pose-based Control Law

The pose-based controller is responsible for determining the angular velocity of the end-effector and the bi-normal velocity with respect to the path. This component ensures that the sanding tool does not deviate from the predefined path and orientation by rapidly driving the manipulator back if it senses a difference between its position and the path. The pose-based formulation uses two parallel rules, one for angular velocity computation and another for the bi-normal velocity formulation. The angular velocity component is a Proportional-Derivative (PD) controller described by:

$$e_p = d_p - p$$

$$v_p = K_{p,p} * e_p + K_{d,p} * \dot{e}_p$$

Equation 2

where $p = [p_\alpha, p_\beta, p_\gamma]^T$ is the end-effector orientation in normal, bi-normal, and tangential directions with respect to the Cartesian coordinate system. $dp = \left[d_{p_\alpha}, d_{p_\beta}, d_{p_\gamma}\right]^T$ is the desired orientation of the end- effector in the normal, bi-normal, and tangential directions. $K_{p,p}$ and $K_{d,p}$ are 3 × 3 diagonal matrices representing the gains of the controller. The output of the controller is the commanded velocity $v_p = [v_\alpha, v_\beta, v_\gamma]^T$ consisting of the angular velocities. This control structure was chosen to ensure a smooth motion of the end-effector avoiding any overshoot and robust transient response.

The bi-normal velocity component is designed for maintaining the tools position along the path, using the position error in the bi-normal direction. The controller architecture takes the form of:

$$e_b = d_b - b$$

$$v_b = K_{p,b} * e_b + \int e_b dt + K_{d,b} * \dot{e}_b$$

Equation 3

where v_b is the velocity in the bi-normal direction, $K_{p,b}$, $K_{i,b}$ and $K_{d,b}$ are the gains of the controller, and e_b is the error in position in the bi-normal direction, defined by the desired position in the bi-normal direction, d_b, and the current position, b. For this application we

consider that the desired position in the bi-normal direction is zero as we do not wish for the tool to stray off from the planned path.

3.2.2.2.3 Force Regulation

The force regulator is responsible for maintaining the desired force in the normal direction during sanding. With the use of the UR5e for testing, the UR5e implements its own force control algorithm and is used in tandem with the pose and vibration control laws. The UR5e's force regulator accepts a user input of the desired force and the direction on which this force has to be applied.

3.3. Experimentation and Results

3.3.1. Experimental Apparatus

For experimentation setup, a UR5e robot from Universal Robotics was used and can be seen in Figure 3.1. A Makita XOB01T orbital sander was attached to the UR5e's force-torque sensor located at the end-effector using a custom-made 3D printed bracket. The force-torque sensor allows for the force measurement in the three spatial directions and torque measurement in the three rotational directions. The orbital sander speed was set to 9,500 orbits per minute for each test. The sanding speed is fixed by the manufacturer and cannot be adaptively controlled. A 127 mm diameter 80 grit sand paper pad was attached to bottom of the sander and blocks of wood and metal were used as the sanding surface. The surfaces vary in size with a minimum width and length of 140 × 480 mm. The proposed expert system was implemented and data was recorded using the UR5e's Real Time Data Exchange network protocol with the use of an externally connected Microsoft Surface laptop using Python 3.1

3.3.2. Experimental Procedures

During the experimental testing, the manipulator was tasked with applying and maintaining a 15-newton reference force and moved along the block of wood/metal (Figure 3.6), while the orbital sander was operational. The reference path of the sander was a straight line of 25 centimeters. The aim of this experiment was to decrease the roughness as much as possible. It is noted that depending on application, the desired roughness value could be any specific target. The proposed architecture was compared to a baseline approach in sanding presented in [6], where the manipulator moves at a constant velocity along the work surface. For consistency, each experiment ran both approaches on the same block of wood/metal but at different locations. The proposed strategy was run first for all experiments. We then calculated the mean tangential speed for that particular experiment ($\bar{v}_t^n$). This value was then used as the requested velocity for the baseline constant velocity tests for the corresponding experiment. A Micro Photonics Nanovea optical profilometer was used to measure the roughness of wood/metal before and after sanding for each controller

Figure 3.6. Wood and steel samples used for testing

3.3.3. Experimental Results

This section presents the results obtained when the proposed approach has been tested in the environment described in the previous paragraph. Table 1 shows the results of the experiments performed on wood and metal blocks. The first six experiments were performed on wood, and the last six were performed on metal. The mean roughness measurements were obtained with the optical profilometer described previously. The table shows the average of three random measurements taken by the profilometer on the work surface. The optical profilometer scans a 2 mm × 2 mm area and measures the distance between the scanned area and a light source. This provides an elevation map of the scanned areas, from which the surface roughness can be calculated. The surface roughness of the area is calculated using:

$$S_a = \frac{1}{A}\int_A |z|\, dA$$

Equation 4

where A is the area of the measured surface and z is the difference in height from the average height of the measured surface. The mean performance difference metric was calculated by taking the difference between the after roughness of the proposed approach and the constant velocity approach, dividing by the original roughness, and taking the average for all experiments for each material. The metric directly compares how well the proposed controller is able to decrease surface roughness with the constant velocity controller.

3.3.4. Discussion

From Table 1 it can be seen that the proposed approach was successfully able to further decrease the surface roughness of the environment when compared to the baseline approach. Furthermore, the mean surface roughness after sanding shows that the expert system performs better than a constant velocity approach. In experiment 8, the proposed approach was slightly worse compared to the baseline strategy. This may be caused by the natural irregularities found in metal and particularly in this singular specimen. Figure 3.7a shows the overall behavior of the end-effector tool when the proposed approach is used to sand the environment. It can be seen that the end-effector is maintaining a constant force, while doing a single pass over the area to be sanded at a variable velocity. Specifically, Figure 3.7a shows that the desired force (15 N) in the normal axis is achieved and maintained throughout the entire sanding process. The beginning of the graph presents force oscillations around 0 N, showing that the manipulator had yet to make contact with the surface. These oscillations are mainly due to the vibrations of the sander. After 2 seconds, the graph shows the sander achieves contact with the surface. This

force is maintained for the duration of the experiment. Figure 3.7b shows the recorded motion versus reference path for the end- effector, displayed in the tangential - bi-normal axes. The proposed approach ensures the position along the path is maintained, the error in the path being negligible to the degree of 10^{-4} m. Figure 3.7c shows the tangential and bi-normal velocity of the manipulator over time. The velocities are set to 0 m/s until the UR5e's force-torque sensor measures a 15 N contact force. This graph shows how the robotic arm velocity needs to be adjusted when the system detects a rougher surface. At the 14 second mark in this graph, the system detects that it is in contact with a rough surface and decreases the tangential velocity to maintain its location. Once the controller senses that the surface is smooth, it then increases its tangential velocity to the end of the path.

The velocity in the bi-normal direction is small in comparison to the tangential velocity due to the lack of a deviation in the bi-normal position. The proposed approach's main advantages are that it is able to sand a given surface intelligently and further decrease surface roughness when compared to a constant speed approach. The force measurements and tangential velocity may also indicate the conditions of the surface without the need of a profilometer, and simply relying on a force/torque sensor placed at the end-effector of the robotic arm. Furthermore, the simplicity of the system allows for real-time implementation. The main limitation of this approach is that prior information about the vibration characteristics is needed for every specific material, which may not be readily available. Another limitation is that the proposed approach will require more time to complete the sanding operation when compared to a constant speed strategy. In industrial settings, however, an operator will

typically perform multiple sanding passes over a single area to obtain a high-quality surface while the proposed method would ideally only need one, saving time overall.

Table 1. Roughness measurements [µm2] for wood, steel and aluminum samples: before sanding, after sanding using the proposed variable velocity approach, and using a constant velocity as presented in [6].

	Wood Ra [µm^2]						Steel Ra [µm^2]		Aluminum Ra [µm^2]			
Experiment #	1	2	3	4	5	6	7	8	9	10	11	12
Before	15.16	18.71	14.16	16.56	17	16	554	320	1.011	0.7415	3.348	1.445
Proposed	12	8.8	8.31	9.17	7.71	11.52	176.8	204	0.5596	0.6066	1.004	0.8086
Constant	13.13	9.46	8.76	14.53	11.49	13.59	187.86	199.87	0.7631	0.6517	1.178	0.9381
Mean Preform. Diff.	13.61%						0.35%		14.22%			

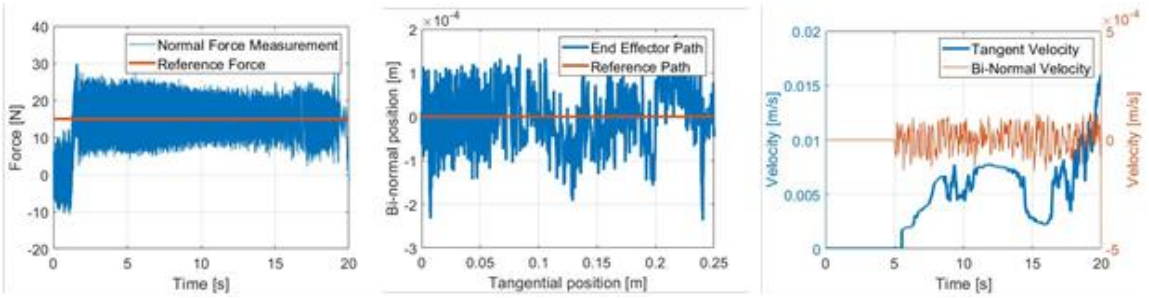

Figure 3.7. The end-effector behavior using the proposed expert system: (a, left) contact force measured during sanding, (b, middle) the end-effector position in the bi-normal and tangential coordinate system, (c, right) the velocity of the manipulator in the tangent and bi-normal direction

3.3.5. Conclusions

The proposed strategy is based on the correlation found between vibrations and surface roughness. An analysis using the FFT of the forces measured with a force/torque sensor is performed, showing the correlation between the sanding frequency and the amplitude of the frequency with the surface roughness. Additionally, an expert system consisting of a vibration-driven and a pose-driven control law is presented. The system leverages the surface roughness observations performed with the FFT to adjust the velocity of the end-effector based on the roughness of the area to be sanded. This formulation allows for adaptive velocity, which ensures reliable sanding with one single pass of the end-effector. Further improvements in performance can be obtained if more is known about the vibration response, i.e., the frequency dampening and amplitude increase. Unfortunately, this requires a great deal of more testing as each different material being sanded must have different gain parameters. Furthermore, a comparison with a constant speed controller shows that our proposed control strategy is able to obtain a smoother surface. Future works will be discussed in the Future Works section 5.2.

Chapter 4. Path Planner

4.1. Introduction

The work presented in Chapter 3 focuses solely on the use of a stationary robot manipulator. For realization of a more applicable solution, the robot manipulator will need to be attached to a mobile base to allow the system to perform work on large manufacturing pieces. The proposed method falls into the category of a coupled path planning method for the mobile base and manipulator that uses probabilistic planning methods. Generally speaking, this method uses configuration sampling along a given end-effector trajectory in an attempt to find a path, thus the probabilistic aspect. This method was chosen due to the robust probabilistic completeness of this method which ensures a solution if one exists. Path planning for high-dimensional systems is typically computationally expensive and optimization algorithms can have difficulty searching through multiple dimensions to find a solution. Probabilistic planners are able to handle high-dimensional spaces better due to the lack of need to search high-dimensional space.

4.2. Methodology

The path planner introduced here is broken down into two steps: 1) sampling and 2) graph search. The overview of this path planning algorithm is that a pre-determined end-effector path is broken into N number of points. Each point is an end-effector pose that must be achieved by the mobile manipulator's configuration. A configuration is represented by a vector θ that contains the values for the degrees-of-freedom for the system. For each point, an arbitrary number of M configurations are generated that achieves the end-effector pose. A configuration path is achieved by connecting a sampled robot configuration, Θ, from the beginning end-effector path point to the next until the last path point is reached. A weighted

distance norm that compares the robot system's joint and base travel distance is used to determine which sampled robot configuration to choose from.

4.2.1. Sampling

A pre-determined path is divided into N number of points. For each point, M sampled robot configurations are found. This section goes over how the samples are generated. A point that the mobile manipulator must reach is a six-dimensional point in space consisting of $[x_N, y_N, z_N, r_N, \rho_N, \gamma_N]^T$. The first step is to generate a random base configuration around said point. The mobile base's configuration is described $[x_b, y_b, z_b, r_b, \rho_b, \gamma_b]^T$ in Cartesian coordinates. The values for z_b, r_b, and ρ_b are zero for this mobile base used due to grounding constraints, leaving $[x_b, y_b, 0, 0, 0, \gamma_b]$. The variables z_b, r_b, and ρ_b refer to the height of the base, rotation about the forward direction, x, and rotation about the side direction, y, respectively. The sampling range of x_b and y_b is a normal distribution with mean of the desired point path's x_N and y_N, respectively, with range of the manipulator's reachable length. This is due to the fact that the mobile manipulator must be in some reachable space of the target point to achieve the desired end-effector pose. The reachable length for the UR5e, in this case, is 0.35 meters. γ_b is sampled with a mean of $\tan^{-1}\left(\frac{y_{N+1}-y_N}{x_{N+1}-x_N}\right)$. This is done to anticipate changes in orientation so that the mobile manipulator's x-direction (forward) can be parallel to the end-effector path. This effect is desirable as it grants more anticipation of how the robot will act which can lead to safer operation as well as play a vital role if holonomic constraints are present in the system that does not allow the robot to turn easily.

Once a base configuration has been sampled, inverse kinematics is used to find the associated joint configuration of the attached manipulator in order to reach the desired end-

effector pose. The inverse kinematic method to find the required joint positions used is the Newton-Raphson method [28] with an initial guess of $[0, -90, 90, -90, -90, 0]°$ for the joint angles of the UR5e. Wrapping to and checking of joint limits is then performed to ensure that the solution is applicable. The joint limits of the manipulator were generally from $-\pi$ to π but slightly different depending on the specific joint. In cases where the pre-defined joint limits have been broken or there was not a solution found, another sample is created and the above process is performed, replacing the faulty sample. This method is performed until M number of solutions are generated for N number of points along a given path.

4.2.2. Path Search

This section will detail how a path is created from the $N \times M$ number of samples created in the previous section. Each N number of path points has an associated M number of sampled configurations. The methodology to find a path works by first picking an initial configuration from the M configurations at the first path point. We then calculate a weighted joint distance to all the M configurations in the next path point. The configuration with the smallest weighted distance is chosen and the process is repeated until N number of configurations have been chosen, giving a configuration path for the mobile manipulator. This method of path search is a greedy optimizer. The weighted distance calculation from one robot configuration is shown in Equation 5. Δq is the difference in configuration vector of the mobile manipulator from the initial robot configuration to the candidate configuration. A is the weight matrix which allows for more emphasis to be had on the movement of specific joints depending on a desired response.

$$Weighted\ Distance = \sqrt{\frac{1}{2}\Delta q^T A \Delta q}$$

Equation 5

A is a $n \times n$ matrix where n is the degrees of freedom of the mobile manipulator. In this case n equals 12, six for the base and six for the manipulator. The weights of matrix A were determined experimentally. It was determined that movement in the orientation (or γ) and y direction was to be limited; thus, their relative weights were set to be higher. The justification for this is due to the holonomic constraints on the mobile manipulators planned for use. The holonomic constraints are due to the slip steering system on the mobile manipulator which limits its ability to turn. The robot used in simulation do not have these constraints as additional work must be done to model these constraints.

4.3. Experimentation and Results

4.3.1. Experimental Procedures

Mobile manipulator path-planning experiments were performed on computer simulation. The simulation software used is Gazebo, an open-source robot simulation tool. The robot used to perform the test was the RB-Kairos, a mobile manipulator that consists of a base with four mecanum wheels and a UR5e manipulator attached on top. Figure 4.1 shows the simulation software and robot. The model used was provided by Robotnik, a company that manufactures the same robot. The robot accepts commands in simulation through the ROS middleware framework. The mobile manipulator has 9 degrees-of-freedom and consists of the

3 ground plane coordinates and orientation, x_b, y_b, and γ_b, and the 6 joints present in the UR5e.

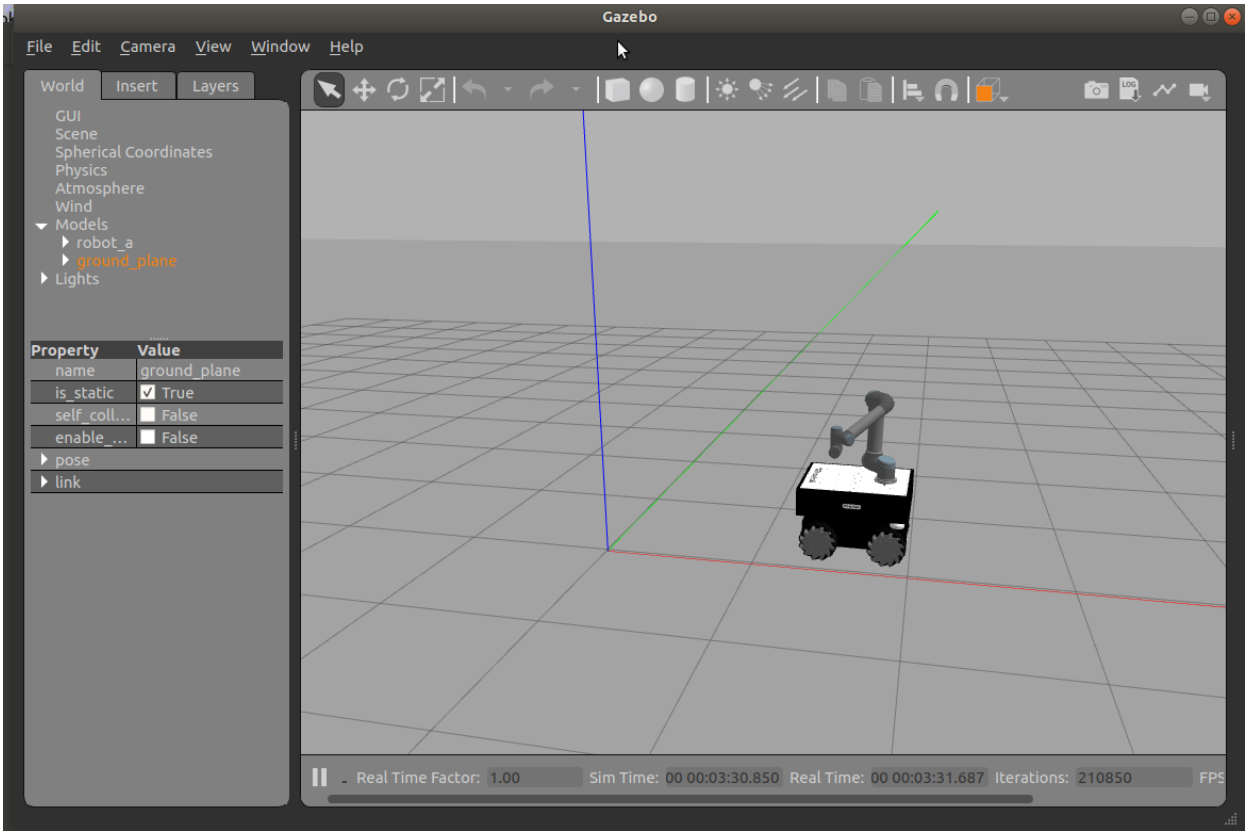

Figure 4.1. Simulation Software with Mobile Manipulator

The mobile manipulator is set to follow three paths, a straight-line, a sinusoidal path, and a sinusoidal path that increases in amplitude. Figure 4.2, Figure 4.5, and Figure 4.8 show the Cartesian representation of each path, respectively, and is in units of meters. Each path was divided into 100 equidistant points with 500 sampled configurations. Path search was then performed to find a suitable path for the mobile manipulator's joints. Initially, experiments were first performed on the straight-line path and the weights for the distance path search were analyzed and tuned until a qualitatively desirable path was found. This was qualitatively

chosen such that there would be minimal movement in the orientation of the base of the mobile manipulator. Once a desired weight matrix has been found, experiments on the three paths were performed. The next section shows the path results of the joints for the mobile manipulator. The paths found for each degree-of-freedom were then simulated to ensure if the path was successful.

4.3.2. Experimental Results

The following figures show the joint paths for the mobile manipulator for the respective end effector path. The weight matrix used was a diagonal matrix of $[10, 10, 10, 1, 1, 10, 1, 100, 0, 0, 0, 100]$. The first 6 weights refer to the joints of the manipulator from the base to the end-effector, respectively. The next 6 refers to the cartesian positions of the base, $[x_b, y_b, z_b, r_b, \rho_b, \gamma_b]^T$, respectively. A weight emphasis was put on the bottom three joints of the manipulator due to the inertia associated with that joint as that can be susceptible to jerk. Another emphasis was put onto the y and γ positions of the base as it was desired that the base does not slide or change direction if unnecessary due to the slip steering system of the physical mobile manipulator. Figure 4.3 and Figure 4.4 shows the joint path for when the end-effector must follow a straight line. The straight-line trajectory is 5 meters in length at a height of 0.7 meters from the base mounting of the arm. The orientation of the end-effector is kept at a neutral orientation where it is facing a side of the base, specifically where the roll angle is -90° and the others orientation angles are zero.

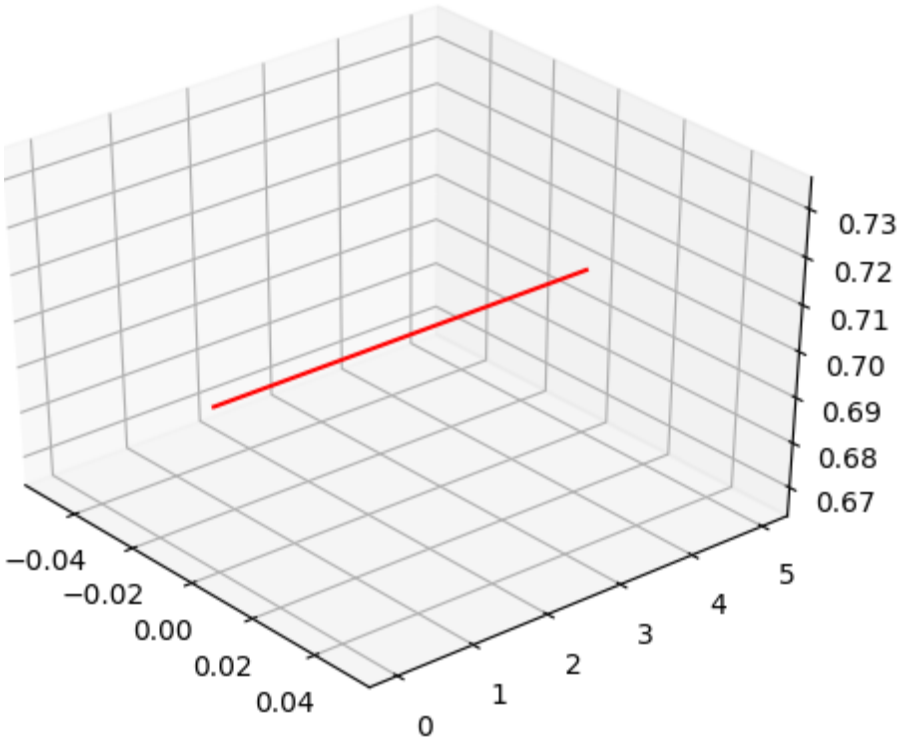

Figure 4.2. Cartesian Coordinates for Straight-Line End-Effector Trajectory

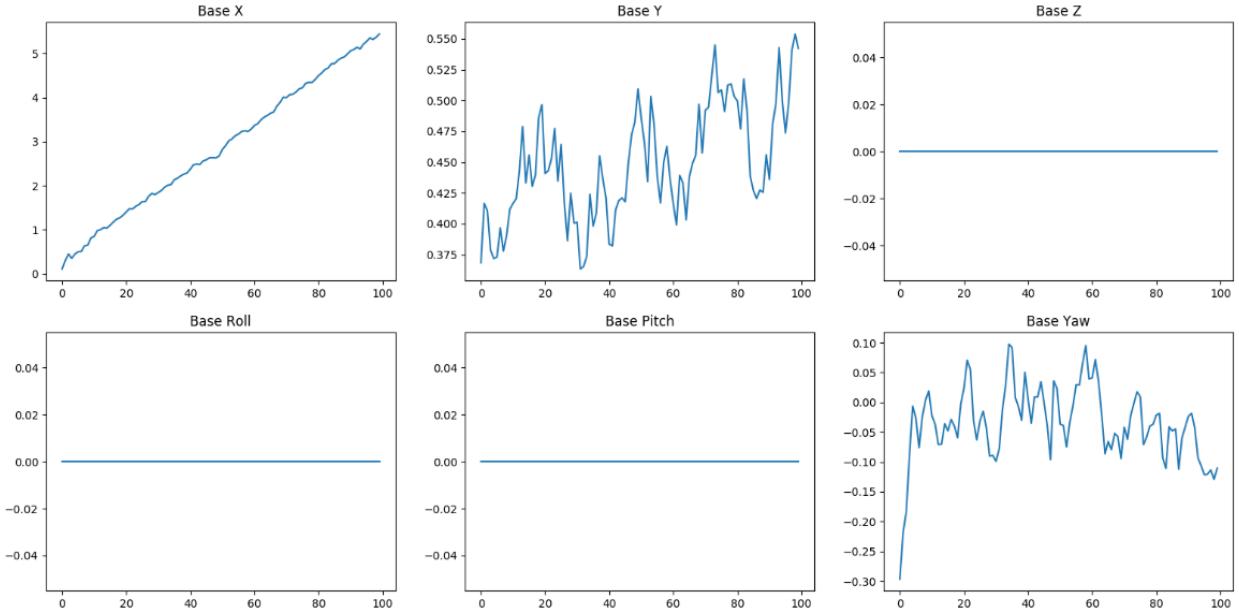

Figure 4.3. Mobile Manipulator Base Path for a Straight-Line End-Effector Trajectory

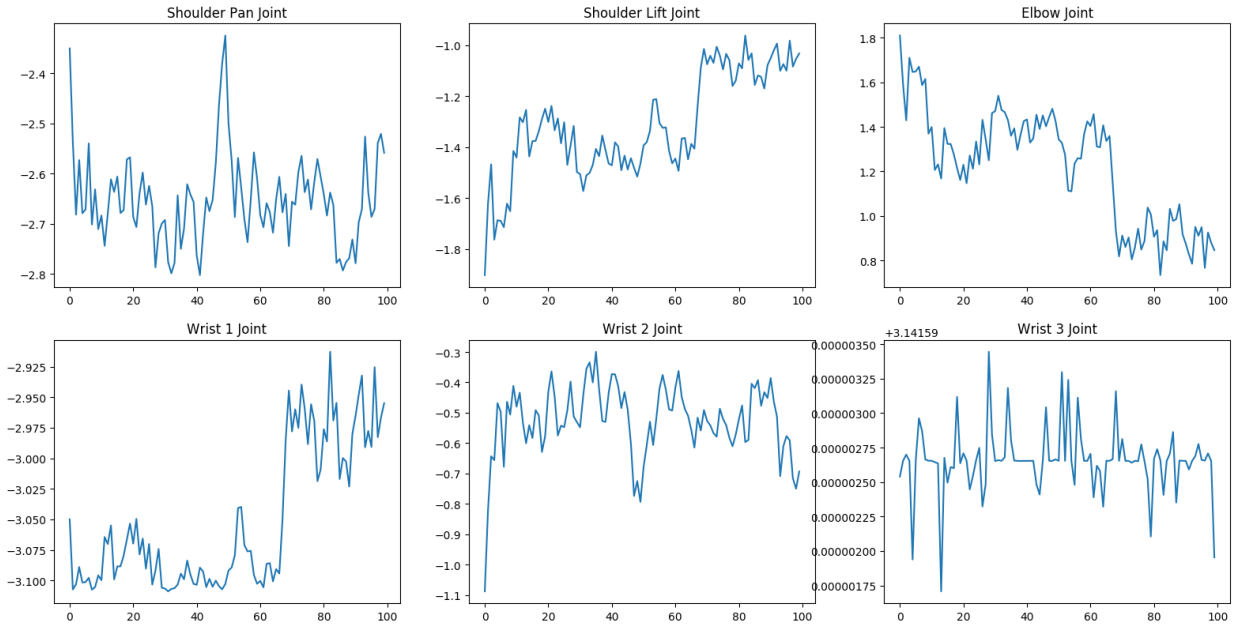

Figure 4.4. Manipulator Joint Path for a Straight-Line End-Effector Trajectory

Figure 4.5, Figure 4.6, and Figure 4.7 shows the desired end-effector trajectory of a sinusoidal, mobile manipulator base path, and manipulator joint path respectively. The weights of matrix A are kept the same as the straight-line path.

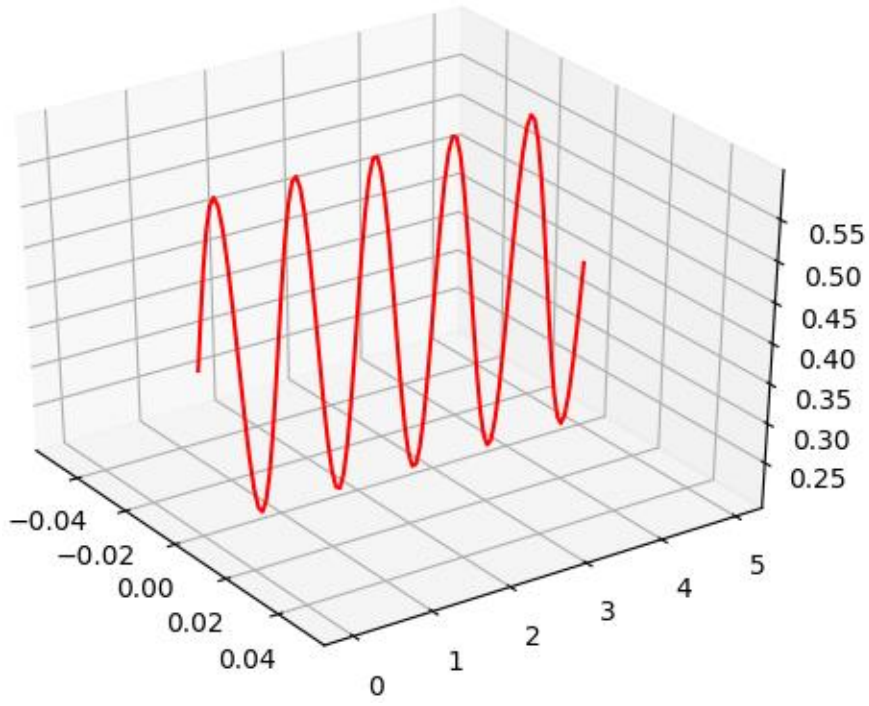

Figure 4.5. Cartesian Coordinates for Increasing Sinusoidal End-Effector Trajectory

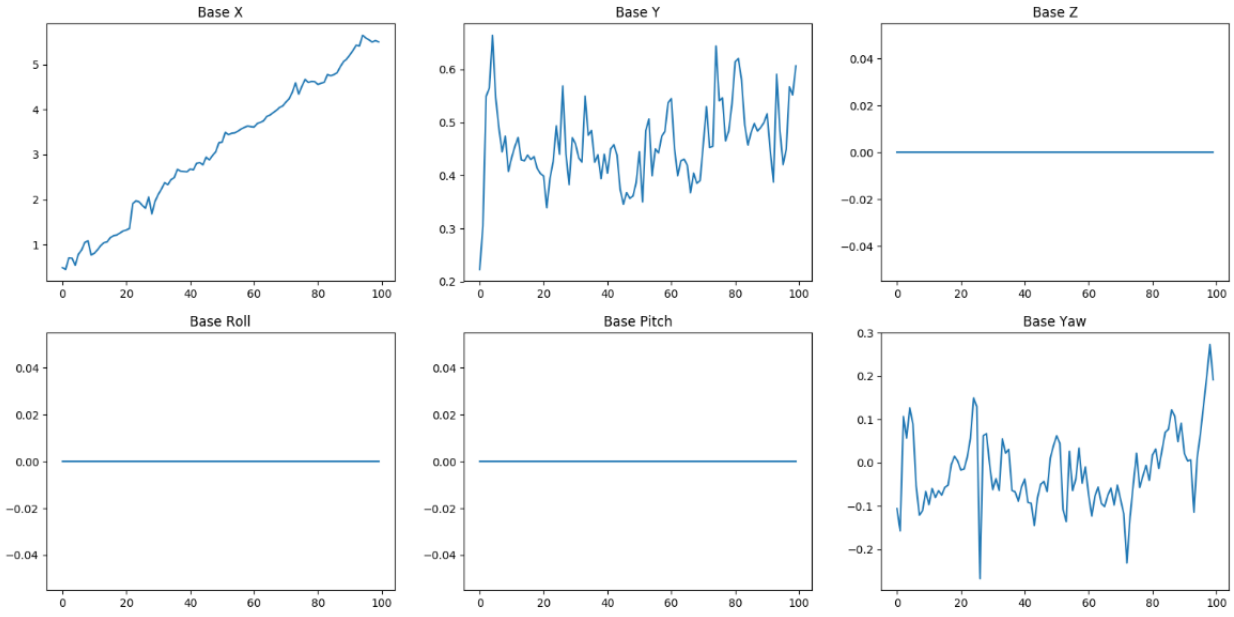

Figure 4.6. Mobile Manipulator Base Path for a Sinusoidal End-Effector Trajectory

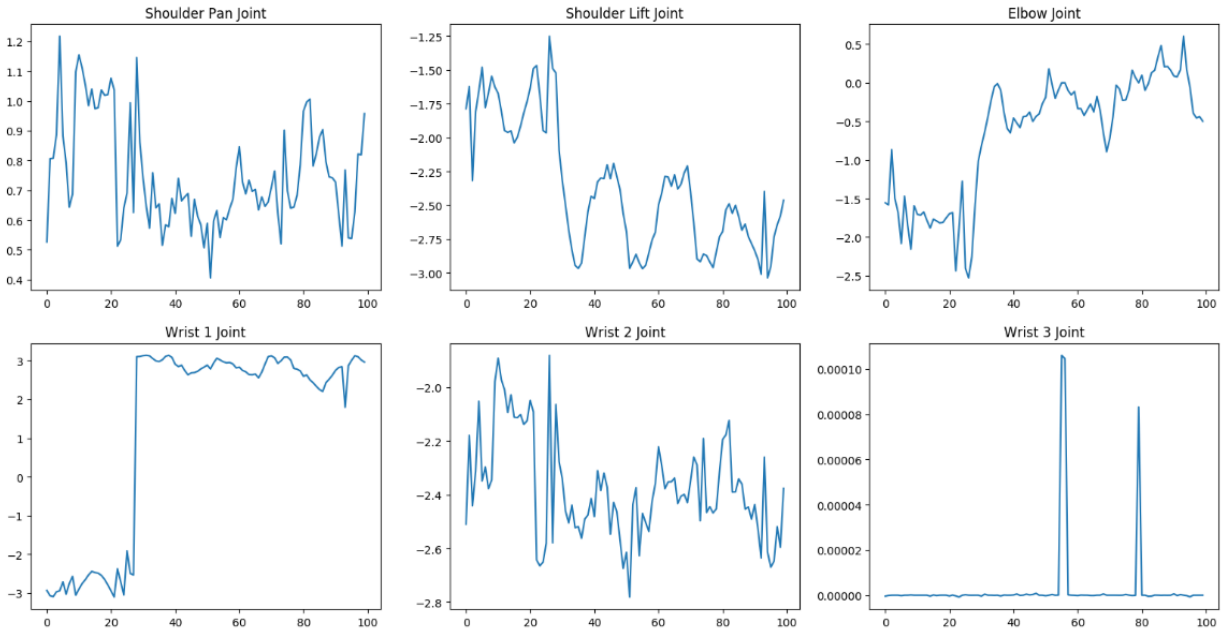

Figure 4.7. Manipulator Joint Path for a Sinusoidal End-Effector Trajectory

Figure 4.8, Figure 4.9, and Figure 4.10 shows the desired end-effector trajectory of an increasing sinusoidal, mobile manipulator base path, and manipulator joint path respectively.

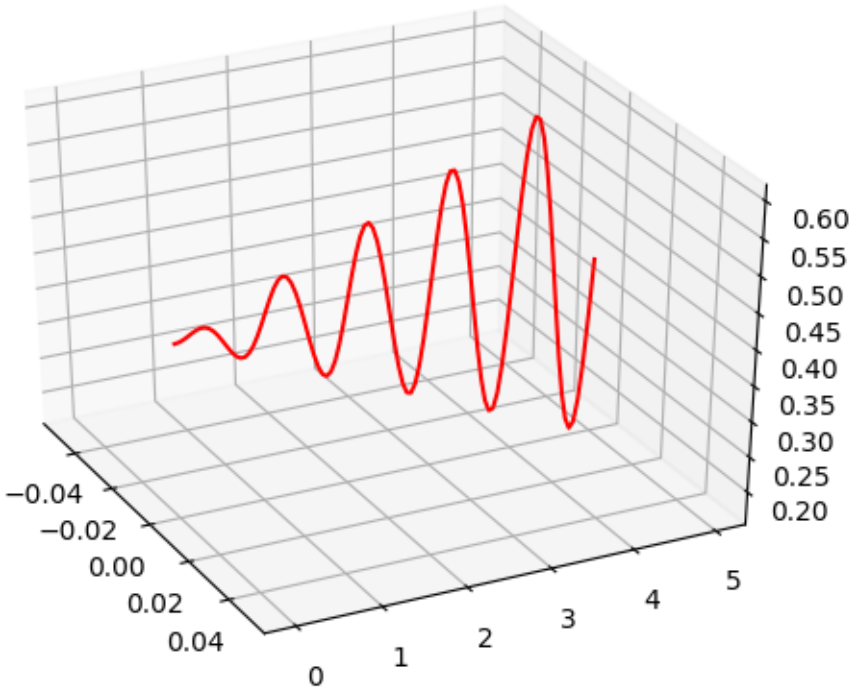

Figure 4.8. Cartesian Coordinates for Increasing Sinusoidal End-Effector Trajectory

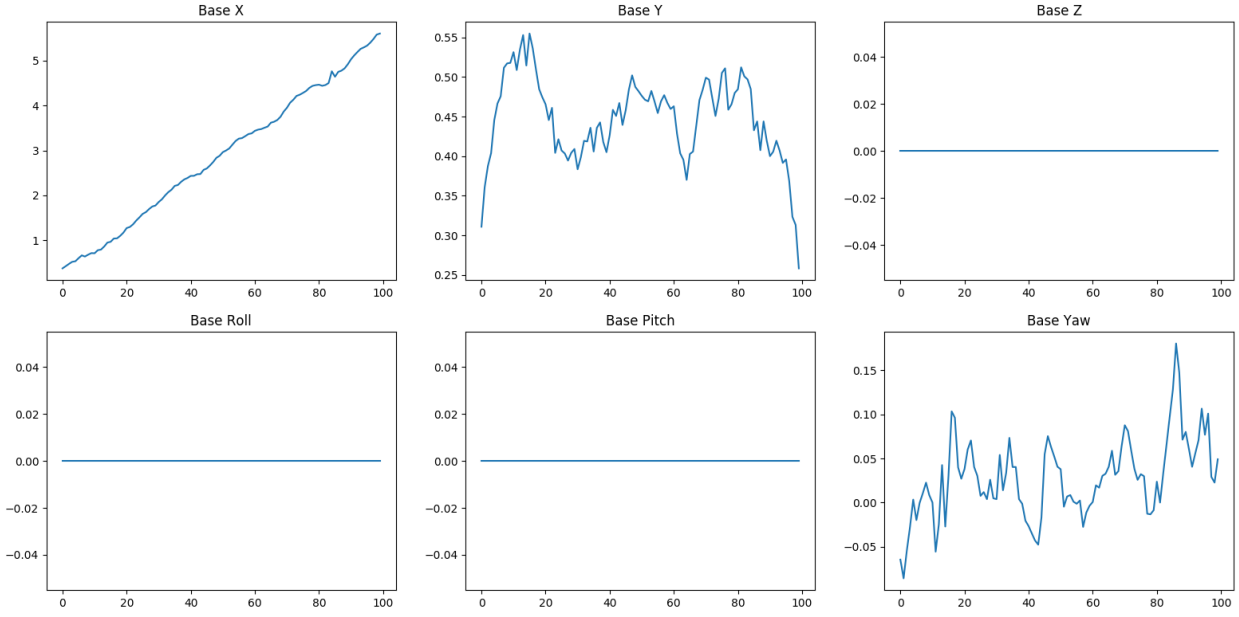

Figure 4.9. Mobile Manipulator Base Path for an Increasing Sinusoidal End-Effector Trajectory

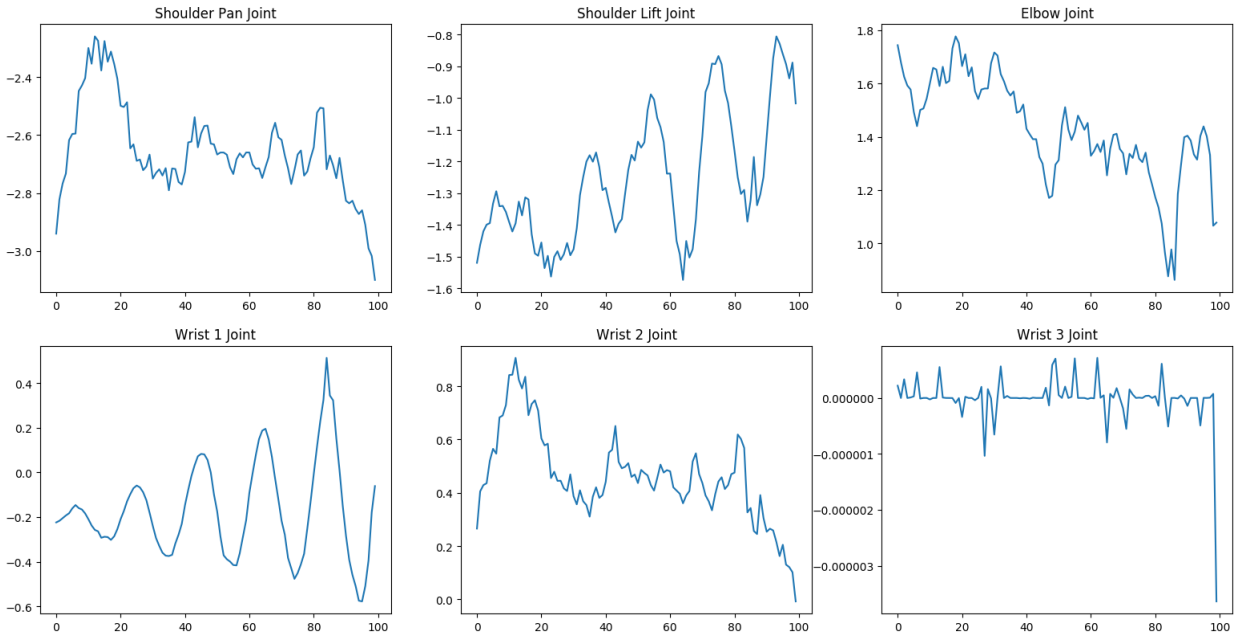

Figure 4.10. Manipulator Joint Path for an Increasing Sinusoidal End-Effector Trajectory

4.3.3. Discussion

The graphs show the path developed by the proposed algorithm. Observing the results of the algorithm for the straight-line path, we can see that the determined weights have the desired effects of influencing the y- and γ- direction to a small range. The straight-line path was used to determine the weights as the effects of the weights are much clearer when different weights are compared. Looking at the path generated for the sinusoidal paths, the joint paths for the manipulator can be seen to have sinusoidal paths as well. Simulation of these paths showed that these paths successfully maintained the end-effector to its desired trajectory and was evaluated qualitatively. There are small disconnects in the trajectory due as the mobile base and arm joints are not timed to reach the end-effector point at the same time.

It is important to note that the algorithm used is greedy and does not optimize the weighted distance over the entire path but only from point to point. The weight matrix A used in experimentation was a diagonal matrix. The diagonal matrix penalizes the distance of only one degree-of-freedom. Off-diagonal weights would penalize the use of any pair of degrees-of-freedoms. Further experimentation would need to be done to understand if the use of off-diagonal weights would produce more desirable results.

The time to produce these paths from start to finish is relatively high when compared to other path planners readily available. Planning performed from similar works takes generally 2 minutes while the proposed method currently takes about 4-6 minutes. This is mainly due to the lack of optimization done in the coding of the algorithm. Currently, there are many checks within the code that require additional computational costs as well as tasks that can be dynamically programmed for better use of available computational resources. There exists a much other things that can be removed/changed that would improve the efficiency of the code as well. The path planners readily available have had the opportunities to be peer reviewed and optimized by others.

This algorithm has yet to be tested for environments that contains obstacles, but it is believed that the proposed method would be able to sufficiently generate a path as probabilistic path planning methods tend to have an easy time navigating through such environments. Probabilistic planners, such as this one, have a tendency to be jerky as the planner outputs point-to-point paths. To solve for this, many others have processed the given path through a 2nd algorithm [20] that would smooth the path. There are many techniques that

can accomplish this, such as Dubin's Curves, splines, and clothoids [29]. Smoothing of the path was not done due to time constraints.

4.3.4. Conclusion

The algorithm developed has shown to create a path for each degree-of-freedom of a mobile manipulator for a given end-effector path. Simulation of the paths have shown the success of the paths generated. Use of a weighted distance for path searching has successfully shown that the joint paths can be influenced to a more desirable trajectory. More work needs to be done to implementation optimization, weight tuning/experimentation for non-diagonal matrices, and testing for environments with obstacles. Future works to better improve usability includes a smoothing algorithm so that the instant acceleration combined with the inertia of the systems of the joints and body, respectively, do not cause harm to itself and/or nearby users.

Chapter 5. Conclusion

5.1. Brief Conclusion

In this thesis, algorithms for control for intelligent robotic sanding and path planning for mobile manipulators were developed. The control algorithm successfully performed sanding while also outperforming current strategies. The path planning algorithm developed successfully generated paths for all the degrees-of-freedom for a mobile manipulator whose end-effector path was pre-determined. The combination of these work provides the first steps to realizing a fully autonomous, industry-ready, robotic sanding system that will assist humans performing this work and removing them from the issues, liabilities, and costs associated with it.

5.2. Suggestion for Future Work

The need of future work can pertain to all levels of work presented in this thesis. This section will present ideas for future work that can be built off the work shown. The ideas for future work will be separated into three categories: future work for 1) intelligent control for sanding, 2) path planning for mobile manipulators for sanding, and 3) the combination of the priors.

Two main limitations of the controls work are the abilities to ensure sanding over a curved surface and lack of a model to estimate surface roughness during sanding. Material removal rate of sanding operations depend mainly on the pressure exerted by the tool and an undesired pressure due to complex contact geometry can cause unwanted and unrepairable cutting, thus a need for a more refined approach is needed for curved surfaces. The controller presented depends on trends observed on experimental sanding data. A need for a model that can map specific roughness values from the vibratory force response would enable the control

system to have precision and to reach a desired roughness value. There exist cases in which a low roughness value can be too low as roughness such as in adhesion tasks where a certain level roughness is needed.

The path planning algorithm has yet to be performance optimized, in a programmable sense, which currently limits it usage. Within the program used to perform the algorithm, there still exists many redundancies and checks to assist with debugging that no longer needs to be used, which increases run time. Investigations on the outputs of the algorithm when the weight matrix is non-diagonal should be performed to see if more desirable results can be obtained. The non-diagonal elements would penalize coupled movement if implemented. The addition of obstacles to see how the algorithm performs should also be investigated.

Lastly, to realize fully autonomous sanding, a combination of intelligent control and path planning of the mobile manipulator must be done. As of now, the control algorithm is only developed of a still manipulator and the path planner does not incorporate the idea of maintaining a force to a given surface.

Chapter 6. Bibliography

[1] H. Zhang, L. Li, J. Zhao, J. Zhao, S. Liu, and J. Wu, "Design and implementation of hybrid force/position control for robot automation grinding aviation blade based on fuzzy PID," *Int. J. Adv. Manuf. Technol.*, vol. 107, no. 3–4, pp. 1741–1754, 2020, doi: 10.1007/s00170-020-05061-y.

[2] B. Luo, L. Li, H. Liu, M. Xu, and F. Xing, "Analysis of sanding parameters, sanding force, normal force, power consumption, and surface roughness in sanding wood-based panels," *BioResources*, vol. 9, no. 4, pp. 7494–7503, 2014, doi: 10.15376/biores.9.4.7494-7503.

[3] S. S. Leite, G. M. K. Jesus, M. C. de S. Alves, I. de D. Valarelli, F. A. Moizes, and V. M. Salvadeo, "Experimental investigation of parameters impacting the roughness of Pinus elliottii wood," *BioResources*, vol. 14, no. 1, pp. 2051–2061, 2019, doi: 10.15376/biores.14.1.2051-2061.

[4] B. Giublin, J. A. Vieira, T. G. Vieira, L. G. Trabasso, and C. A. Martins, "Experimental analysis of the automated process of sanding aircraft surfaces," *Aeronaut. J.*, vol. 118, no. 1199, pp. 53–64, 2014, doi: 10.1017/S0001924000008927.

[5] I. Mohsin, K. He, Z. Li, and R. Du, "Path planning under force control in robotic polishing of the complex curved surfaces," *Appl. Sci.*, vol. 9, no. 24, Dec. 2019, doi: 10.3390/app9245489.

[6] B. Maric, A. Mutka, and M. Orsag, "Collaborative Human-Robot Framework for Delicate Sanding of Complex Shape Surfaces," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2848–2855, 2020, doi: 10.1109/LRA.2020.2969951.

[7] P. Trudeau, C. Demaria, G. Palardy, H. Salek, M. A. Jetté, and P. Hubert, "Process induced deformation of aircraft structural components," *ICCM Int. Conf. Compos. Mater.*, vol. 2013-July, pp. 1329–1336, 2013.

[8] J. H. Ahn, M. C. Lee, H. D. Jeong, S. R. Kim, and K. K. Cho, "Intelligently automated polishing for high quality surface formation of sculptured die," *J. Mater. Process. Technol.*, vol. 130–131, pp. 339–344, 2002, doi: 10.1016/S0924-0136(02)00821-X.

[9] J. H. Ahn, Y. F. Shen, H. Y. Kim, H. D. Jeong, and K. K. Cho, "Development of a sensor information integrated expert system for optimizing die polishing," *Robot. Comput. Integr. Manuf.*, vol. 17, no. 4, pp. 269–276, 2001, doi: 10.1016/S0736-5845(00)00057-0.

[10] T. Thomessen, T. K. Lien, and P. K. Sannæs, "Robot control system for grinding of large hydro power turbines," *Ind. Rob.*, vol. 28, no. 4, pp. 328–334, 2001, doi: 10.1108/01439910110397183.

[11] Y. Sun, D. J. Giblin, and K. Kazerounian, "Accurate robotic belt grinding of workpieces with complex geometries using relative calibration techniques," *Robot. Comput. Integr. Manuf.*, vol. 25, no. 1, pp. 204–210, 2009, doi: 10.1016/j.rcim.2007.11.005.

[12] R. P. Ubeda, S. C. Gutiérrez Rubert, R. Z. Stanisic, and Á. P. Ivars, "Behavioural study of the force control loop used in a collaborative robot for sanding materials," *Materials (Basel).*, vol. 14, no. 1, pp. 1–19, 2021, doi: 10.3390/ma14010067.

[13] A. Wahrburg, E. Morara, G. Cesari, B. Matthias, and H. Ding, "Cartesian contact force estimation for robotic manipulators using Kalman filters and the generalized momentum," *IEEE Int. Conf. Autom. Sci. Eng.*, vol. 2015-Octob, no. August, pp. 1230–1235, 2015, doi: 10.1109/CoASE.2015.7294266.

[14] F. Tian, C. Lv, Z. Li, and G. Liu, "Modeling and control of robotic automatic polishing for curved

surfaces," *CIRP J. Manuf. Sci. Technol.*, vol. 14, pp. 55–64, 2016, doi: 10.1016/j.cirpj.2016.05.010.

[15] L. Zhang, H. Y. Tam, C. M. Yuan, Y. P. Chen, and Z. D. Zhou, "An investigation of material removal in polishing with fixed abrasives," *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.*, vol. 216, no. 1, pp. 103–112, 2002, doi: 10.1243/0954405021519591.

[16] J. P. Huissoon, F. Ismail, A. Jafari, and S. Bedi, "Automated polishing of die steel surfaces," *Int. J. Adv. Manuf. Technol.*, vol. 19, no. 4, pp. 285–290, 2002, doi: 10.1007/s001700200036.

[17] Y. Dong, T. Ren, K. Hu, D. Wu, and K. Chen, "Contact force detection and control for robotic polishing based on joint torque sensors," *Int. J. Adv. Manuf. Technol.*, vol. 107, no. 5–6, pp. 2745–2756, Mar. 2020, doi: 10.1007/s00170-020-05162-8.

[18] J. Ernesto Solanes, L. Gracia, P. Muñoz-Benavent, J. Valls Miro, C. Perez-Vidal, and J. Tornero, "Robust Hybrid Position-Force Control for Robotic Surface Polishing," *J. Manuf. Sci. Eng. Trans. ASME*, vol. 141, no. 1, pp. 1–14, 2019, doi: 10.1115/1.4041836.

[19] B. de Agustina, M. M. Marín, R. Teti, and E. M. Rubio, "Analysis of force signals for the estimation of surface roughness during Robot-Assisted Polishing," *Materials (Basel).*, vol. 11, no. 8, pp. 5–7, 2018, doi: 10.3390/ma11081438.

[20] T. Sandakalum and M. H. Ang, "Motion Planning for Mobile Manipulators—A Systematic Review," *Machines*, vol. 10, no. 2. 2022, doi: 10.3390/machines10020097.

[21] B. Chen, J. Qi, and X. Hu, "Polishing trajectory planning method based on the geometry and physics," *2016 IEEE Int. Conf. Inf. Autom. IEEE ICIA 2016*, no. August, pp. 1461–1466, 2017, doi: 10.1109/ICInfA.2016.7832049.

[22] B. Denkena, M. A. Dittrich, and H. N. Nguyen, "Technological CAD/CAM chain for automated polishing of geometrically complex workpieces," in *Procedia CIRP*, 2018, vol. 78, pp. 313–317, doi: 10.1016/j.procir.2018.09.049.

[23] A. Kharidege, D. T. Ting, and Z. Yajun, "A practical approach for automated polishing system of free-form surface path generation based on industrial arm robot," *Int. J. Adv. Manuf. Technol.*, vol. 93, no. 9–12, pp. 3921–3934, 2017, doi: 10.1007/s00170-017-0726-y.

[24] G. Oriolo and C. Mongillo, "Motion planning for mobile manipulators along given end-effector paths," *Proc. - IEEE Int. Conf. Robot. Autom.*, vol. 2005, no. April, pp. 2154–2160, 2005, doi: 10.1109/ROBOT.2005.1570432.

[25] M. Stilman, "Task constrained motion planning in robot joint space," *IEEE Int. Conf. Intell. Robot. Syst.*, pp. 3074–3081, 2007, doi: 10.1109/IROS.2007.4399305.

[26] D. Kanoulas, J. Sustarevas, and S. Julier, "Task-Consistent Path Planning for Mobile 3D Printing," no. Mm, 2021, [Online]. Available: https://www.researchgate.net/publication/353924035.

[27] R. S. Figliola and D. E. Beasley, "Theory and design for mechanical measurements," *Meas. Sci. Technol.*, vol. 7, no. 7, 1996, doi: 10.1088/0957-0233/7/7/016.

[28] A. A. Goldenberg, B. Benhabib, and R. G. Fenton, "A Complete Generalized Solution to the Inverse Kinematics of Robots," *IEEE J. Robot. Autom.*, vol. 1, no. 1, pp. 14–20, 1985, doi: 10.1109/JRA.1985.1086995.

[29] A. Ravankar, A. A. Ravankar, Y. Kobayashi, Y. Hoshino, and C. C. Peng, "Path smoothing

techniques in robot navigation: State-of-the-art, current and future challenges," *Sensors (Switzerland)*, vol. 18, no. 9, pp. 1–30, 2018, doi: 10.3390/s18093170.

Chapter 7. Vita

Joshua Nguyen is a Vietnamese American citizen born in Baton Rouge, Louisiana on May 15th, 1996. Joshua received his bachelor's degree in Mechanical Engineering in May 2020 with a minor in Robotics and Business Administration. He decided to further his robotics education by pursuing his master's degree in Mechanical Engineering with a focus on control and path planning. Joshua Nguyen received his Master of Science in Mechanical Engineering in August 2022.