Robot-Assisted Endodontic Treatment

Yi-Chan Li

Advisor: Cheng-Wei Chen

Graduate Institute of Electronics Engineering
National Taiwan University
Taipei, Taiwan

June 2021

Abstract

1.introduction: present tense
2.method and result: past tense
3.conclusion : present tense

Contents

Abstract **i**

List of Figures **v**

List of Tables **vii**

1 Introduction **1**

1.1 Motivation 2

1.2 Previous Work and Problem Definition 3

1.3 The Proposed Method 4

1.4 Main Contributions of the Thesis 4

1.5 Organization of the Thesis 4

2 State-of-the-Art **5**

3 Design and Analysis of the Dental Surgical Robot - DentiBot **7**

3.1 Requirement and Specification 7

3.2 Design of the DentiBot 8

3.3 Kinematics Analysis 9

3.3.1 Coordinate Definition 9

3.3.2 Forward and Inverse Kinematics 10

3.3.3 Jacobian matrix 11

3.4 Coordinate Transformation of the Robot Arm 15

3.4.1 Translation Analysis - Tool Center Point 16

3.4.2 Rotation Analysis 18

4 Force-Guided Robot Alignment **21**

4.1 Problem Definition 21

4.2 Integration of F/T sensor 22

4.3 Dragging Mode 30

4.3.1 Admittance Control based on F/T sensor 30

4.4 Self-Alignment Mode 34

4.4.1 Coordinate Transformation of F/T sensor 34

4.4.2 Motion Planning: Based on Admittance Control 37

4.5 Affections of Parameters Setting 38

5 Control of Endodontic File Rotation **41**

5.1 Problem Definition 41

5.2 The Proposed Method and Theorem 41

6 Preliminary Experiment Result **43**

6.1 Experimental Setup 43

6.2 Admittance Control 43

6.3 Automatically Direction Changing 44

6.4 Repetitive Experiment 44

7 Conclusions and Future works **45**

8 Appendix **47**

8.1 Forward Kinematics 47

8.2 Jacobian matrix 48

8.2.1 Jg0 48

8.2.2 Jg6 50

Reference **53**

List of Figures

1.1 The endodontic therapy steps 1

3.1 Workspace of Meca500 8

3.2 Coordinate Definition 9

3.3 Schematic diagram for Tool Center Point. The translation vector ${}^{6}\boldsymbol{p}_{\mathrm{H_{org}}}$ denotes the origin position relative to the frame$\{6\}$. 16

3.4 Schematic diagram for obtaining the tool vector. 18

3.5 Illustration of finding the rotation matrix 19

4.1 Data Analysis of F/T sensor. 23

4.2 Control scheme. $\boldsymbol{f_d}$ denotes the desired forces and torques vector. $\boldsymbol{f_s}$ denotes the real value detected by F/T sensor and is also a forces and torques vector. $\dot{\boldsymbol{x}}$ denotes $[\dot{x}, \dot{y}, \dot{z}, \dot{\theta}_x, \dot{\theta}_y, \dot{\theta}_z]$. $\mathbf{J_g}$ denotes the geometric Jacobian matrix. $\dot{\boldsymbol{q}}$ denotes $[\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5, \dot{\theta}_6]$. $\boldsymbol{q}$ denotes $[\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]$. 31

4.3 Illustration of changing reference frame from $\{\mathrm{S}\}$ to $\{\mathrm{T}\}$. f and m respectively denote a force and a moment measured at point p. . . 35

4.4 Illustration of changing reference frame from $\{\mathrm{S}\}$ to $\{\mathrm{T}\}$. $\boldsymbol{f}$ and $\boldsymbol{m}$ respectively denote a force and a moment measured at point p. 36

4.5 Flow chart of motion planning 37

4.6 Flow chart of motion planning 38

4.7 Control scheme. $\mathbf{M}, \mathbf{B}, \mathbf{K}$ are all diagonal positive definite matrices, whose diagonal indexes are respectively related to $x, y, z, \theta_x, \theta_y, \theta_z$. $\mathbf{M}, \mathbf{B}$ are related to the inertial and damping respectively, and $\mathbf{K}$ is a proportional gain. 38

List of Tables

3.1 Denavit-Hartenberg parameters of Meca500 10

4.1 Arc-function Comparison 29
4.2 Moving commands in Meca500 32
4.3 Parameters setting of Admittance control. 40

Chapter 1

Introduction

Endodontic treatment, also known as root canal treatment and nerve extraction, is performed to cure an infected tooth. The procedure of endodontic treatment is divided into three parts - Opening, Cleaning, and Filling shown in Figure 1.1.

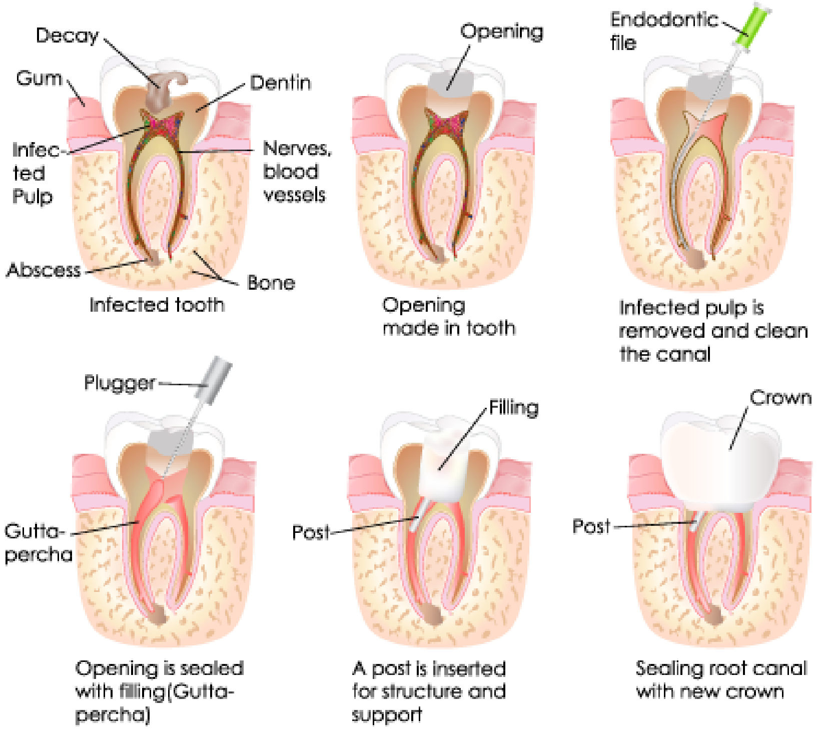
Decay
Gum
Dentin
Infec-ted Pulp
Nerves, blood vessels
Abscess
Bone
Infected tooth
Opening
Opening made in tooth
Endodontic file
Infected pulp is removed and clean the canal
Plugger
Gutta-percha
Opening is sealed with filling(Gutta-percha)
Filling
Post
A post is inserted for structure and support
Crown
Post
Sealing root canal with new crown

Figure 1.1: The endodontic therapy steps

An infected tooth arises from periodontal disease, attrition, trauma, or decay. Once the dental pulp is infected, it causes an irreversible inflammation and lets patients confront a root canal treatment. Figure 1.1 shows an infected tooth and its dental pulp, which consisted of blood vessels, nerves, connective tissues, and lymphatics. In the ”Opening” step, an experienced dentist drills the crown of the infected tooth to remove the dentin and expose the infected pulp inside the canal to the air. Next, in the ”Cleaning” step, the dentist uses an endodontic file, a flexible reamer, to remove the infected pulp. Then, in the ”Filling” step, the dentist uses a dental plugger to fill the empty root canal with Gutta-percha, a plastic substance. ”Filling” can prevent cross-infection between root canals because the cured tooth remains many invisible and inaccessible pulp tissue. Finally, the dentist seals the root canal with a new crown to protect cured root canal.

”Cleaning” is of paramount importance in a whole treatment because untreated cleaning will result in pulp necrosis, apical abscess, periodontal ligament inflammation, or even cellulitis. If there are many remained infected pulp after root canal treatment, the surgery should be operated on again.

1.1 Motivation

The performance of the endodontic treatment depends on the dentist’s long-term expertise. A qualified dentist can operate the endodontic treatment and accumulate their experience to increase the success rate. With many experiences, dentists can acquire an endodontist license. According to statistics from the Ministry of Health and Welfare, R.O.C. (Taiwan), the number of dentists in Taiwan is $15,178$. However, according to The Academy of Endodontology, R.O.C. (Taiwan), there are only 238 dentists to acquire an endodontist license due to the expertise of endodontics. Besides, Root canal treatment is tedious and time-consuming due to the complicated conditions of each tooth. A patient who suffered from an infected tooth spends countless hours see a dentist. It takes at least two to three rounds, even takes more than two months in the worst case.

Therefore, our team looks forward to designing a robot-assisted system to finish the root canal treatment. The definition of a robot-assisted system in the present thesis refers to be a dental assistant. With the system, we wish it can increase the success rate for dentists and provide patients a safe surgery.

1.2 Previous Work and Problem Definition

Use present tense. (Previous work: briefly mention the existing dental robots)

(Problem definition: 1. Assist dentists to operate RCT and focus on cleaning procedure 2. Root canal cannot be visually observed and is too small to clean well 3. Risk of file breakage)

Instruments fracture and perforation are two main problems that commonly occur during the therapy. Removal of broken files is both technically difficult, and therefore it is essential to reduce the probability of fracture. In addition, root canal treatment also requires repeatedly drilling to clean the canal thoroughly. This repetitive action of root canal treatment is tedious and time-consuming. Therefore, we designed an automatic endodontic robot to improve the time efficiency and prevent instrument fracture in endodontic surgery. The root canal cleaning is a big challenge in and of itself. There is one robotic system that is designed to perform endodontic therapy. In Intelligent Micro Robot Development for Minimum Invasive Endodontic Treatment [3], they proposed a microrobot performing root canal treatment with the assistance of a 3D computer model system. It is designed to accomplish endodontic therapy with a path planned according to the 3D model. However, the problem of instruments fracture remains. In this paper, a torque monitoring method is proposed. The leading causes of fractured files are torsional fracture and flexural fatigue, which account for 55.7% and 44.3% separately [5]. Therefore, we use current feedback to keep track of the file's torque during the endodontic treatment. This torque monitoring system is implemented on an endodontic robot prototype we built. We primarily focus on the cleaning and shaping step since it is the key to a successful root canal treatment. With the robot prototype and torque monitoring system, we could reduce the possibility of instrument fracture. Besides, the repetitive action during the drilling step can be performed by the robot.

1.3 The Proposed Method

(Solutions: (be consistent with problem definition) 1. Build a robot-assisted system and enable it to drill 2. Force-guided alignment 3. Control the file rotation speed B. Prospect:
(Prospect: Move to the infected teeth⟶Root canal searching⟶Repetitive drilling⟶Apex Detection)

1.4 Main Contributions of the Thesis

1. Integrate a 6-DoF robotic manipulator with 6-DoF F/T sensor for performing endodontic treatment.
2. Develop a framework for robot alignment regarding the position and orientation of root canal.
3. Protect the endodontic file from fracturing by controlling file rotation speed.

1.5 Organization of the Thesis

Chapter 2

State-of-the-Art

A. YOMI – commercial robot B. HK - dental implant robot C. Korean - dental implant robot D. NCTU – RCT robot

Chapter 3

Design and Analysis of the Dental Surgical Robot - DentiBot

3.1 Requirement and Specification

(Payload, resolution and workspace) (Why not RCM mechanism)

The dental surgical robot should enable dentists to perform delicate and complicated surgery operations because the average diameter of a root canal is $0.28(\pm 0.08)$ mm [1]. Therefore, our system should have a high resolution. Next, an appropriate workspace is required. In dental anatomy, teeth are located on the maxillary (lower jaw) and the mandibular (upper jaw). To perform surgery with both sides, we should rotate the end effector at least 180 degrees. Also, there is previous research, which shows the average range of maximum mouth opening is 50.3 mm ± 6.26 mm [2]. Hence, the end effector of our system should be less than the range. To achieve the workspace, we build a system composed of a robot arm, an F/T sensor, and a modified handpiece. We choose a robot arm and an F/T sensor because we want to mimic the dentist's motion. Due to six degrees of freedom (DOF) with the robot arm and the F/T sensor, we can achieve the action with less hardware restriction. Besides, by modifying the existing handpiece, which is a dental instrument for oral surgery, we do not worry about the workspace in the mouth.

3.2 Design of the DentiBot

As discussed in the previous section, we have to select suitable devices according to those requirements. First, We choose Meca500 manufactured by Mecadamic Inc. as our 6 DOF robot arm. Its feature is high repeatability (precision: 5 µm), and it is equipped with zero-backlash speed reducers. In addition, it is compact and portable for laboratory investigation. Second, Mini40 manufactured by ATI Inc. is the corresponding F/T sensor with three force and three torque detections. As for the end effector, we modify an existing dental handpiece that equips a tool change mechanism. The modified handpiece weighs around 139 grams. We also design adapters to assemble these devices.

DentiBot has seven degrees of freedom. 6 DOF comes from Meca500, and the other DOF is owing to our modified handpiece. The rotation of the root canal file is driven by a servo motor whose maximum rotation speed is more than 600 rpm.

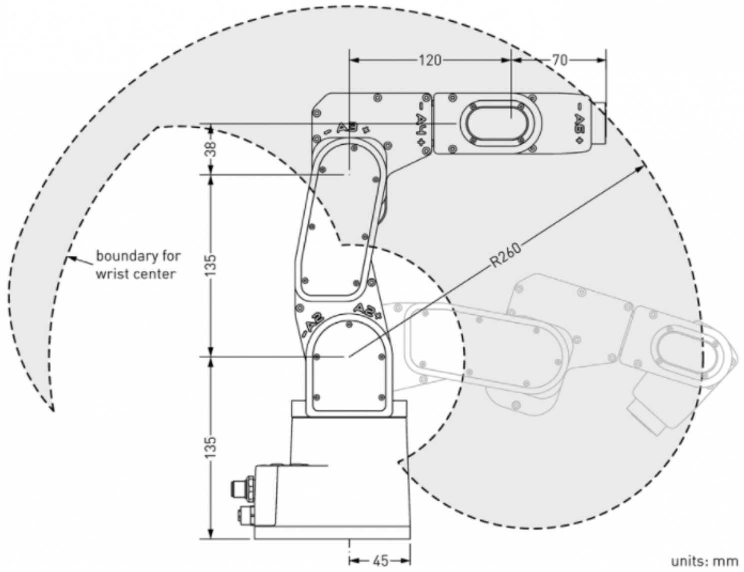

Figure 3.1: Workspace of Meca500

3.3 Kinematics Analysis

The purpose of this section and section 3.4 is to serve as a tutorial and provide some crucial approaches when combining a robot arm and an end effector. We derive the forward and inverse kinematics in section 3.3.2 and describe the Jacobian matrix in section 3.3.3.

3.3.1 Coordinate Definition

In Fig 3.2 , we define frame{0} to frame{6} which represent each frame of axes of the Meca500, frame{S} which represent the frame of the ATI-mini40 and frame{H} which represent the frame of the handpiece.

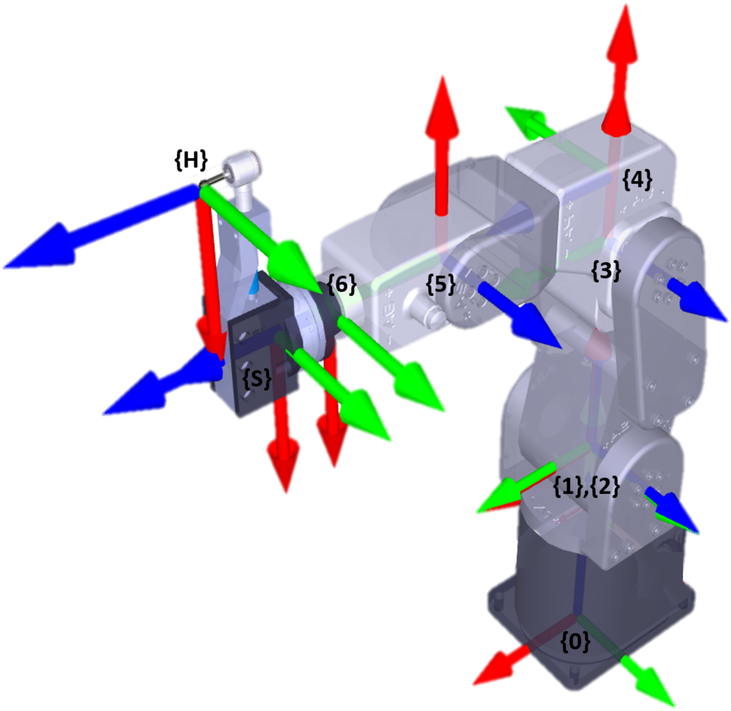

Figure 3.2: Coordinate Definition

Table 3.1: Denavit-Hartenberg parameters of Meca500

i (link number)	α_{i-1} (deg)	a_{i-1} (mm)	θ_i (deg)	d_i (mm)
1	0	0	θ_1	135
2	-90	0	θ_2	0
3	0	135	θ_3	0
4	-90	38	θ_4	120
5	90	0	θ_5	0
6	-90	0	θ_6	70

3.3.2 Forward and Inverse Kinematics

Denavit-Hartenberg parameters are shown as Table 3.1. Then, the forward kinematics of Meca500 is derived as

$$ {}^{0}_{6}\mathbf{T} = {}^{0}_{1}\mathbf{T} \cdot {}^{1}_{2}\mathbf{T} \cdot {}^{2}_{3}\mathbf{T} \cdot {}^{3}_{4}\mathbf{T} \cdot {}^{4}_{5}\mathbf{T} \cdot {}^{5}_{6}\mathbf{T} = \begin{bmatrix} {}^{0}_{6}\mathbf{R} & {}^{0}\boldsymbol{p}_{6\text{org}} \\ 0 & 1 \end{bmatrix} \tag{3.1} $$

where ${}^{0}_{6}\mathbf{R}$ is the rotation matrix from frame$\{6\}$ to frame$\{0\}$, ${}^{0}\boldsymbol{p}_{6\text{org}}$ is the origin of the frame$\{6\}$ observed from frame$\{0\}$. All detailed indexes of ${}^{0}_{6}\mathbf{T}$ are shown as Appendix 8.1

Incidentally, there is an alternative to calculate the transformation matrix of Meca500. We can use command ”GetPose” to obtain (x,y,z,α,β,γ). Then, we can use this information to derive the following equation.

$$ \begin{aligned} {}^{0}_{6}\mathbf{T} &= \begin{bmatrix} & x \\ \mathbf{R_x}(\alpha) \cdot \mathbf{R_y}(\beta) \cdot \mathbf{R_z}(\gamma) & y \\ & z \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} c_\beta c_\gamma & -c_\beta s_\gamma & s_\beta & x \\ c_\alpha s_\gamma + s_\alpha s_\beta c_\gamma & c_\alpha c_\gamma - s_\alpha s_\beta s_\gamma & -s_\alpha c_\beta & y \\ s_\alpha s_\gamma - c_\alpha s_\beta c_\gamma & s_\alpha c_\gamma + c_\alpha s_\beta s_\gamma & c_\alpha c_\beta & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{3.2} $$

where $C_\star$, $S_\star$ denote $\cos(\star)$, $\sin(\star)$; α, β, γ are in representation of Euler angle.

3.3.3 Jacobian matrix

Here we evaluate geometric Jacobian based on frame$\{0\}$, geometric Jacobian based on frame$\{6\}$ and analytical Jacobian.

First and foremost, we should clarify the difference between geometric Jacobian and analytical Jacobian. They both use the same linear velocity but consider different angular velocity. The angular velocity which geometric Jacobian applies is relevant to the axis angles (θ_x,θ_y,θ_z). In contrast, the angular velocity which analytical Jacobian contemplates is related to the orientation (α,β,γ) of the end effector.

Geometric Jacobian Based on Frame$\{0\}$

Before examining the geometric Jacobian matrix, it will be necessary to find the relationship between the position and joints' angles and the relationship between the axis angle and the joints' angles.

Based on the translation matrix in Equation 3.3.2, we obtain the relationship between the position and joints' angles.

$$
{}^0\boldsymbol{p}_{6\mathrm{org}} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x(\theta_1, \theta_2, \cdots, \theta_6) \\ y(\theta_1, \theta_2, \cdots, \theta_6) \\ z(\theta_1, \theta_2, \cdots, \theta_6) \end{bmatrix} \tag{3.3}
$$

Moreover, we dissect the relationship between the axis angle and joints' angles.

$$
\begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \end{bmatrix} = {}^0_1\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ \theta_1 \end{bmatrix} + {}^0_2\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ \theta_2 \end{bmatrix} + {}^0_3\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ \theta_3 \end{bmatrix} + {}^0_4\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ \theta_4 \end{bmatrix} + {}^0_5\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ \theta_5 \end{bmatrix} + {}^0_6\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ \theta_6 \end{bmatrix} \tag{3.4}
$$

We can differentiate Equation 3.3.3 and 3.3.3 to obtain Jacobian matrices.

$$\boldsymbol{v} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \mathbf{J_{gv}} \cdot \dot{\boldsymbol{q}}, \ \boldsymbol{w} = \begin{bmatrix} \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix} = \mathbf{J_{gw}} \cdot \dot{\boldsymbol{q}} \tag{3.5}$$

where

$$\mathbf{J_{gv}} = \begin{bmatrix} \frac{\partial x}{\partial \theta_1} & \frac{\partial x}{\partial \theta_2} & \cdots & \frac{\partial x}{\partial \theta_6} \\ \frac{\partial y}{\partial \theta_1} & \frac{\partial y}{\partial \theta_2} & \cdots & \frac{\partial y}{\partial \theta_6} \\ \frac{\partial z}{\partial \theta_1} & \frac{\partial z}{\partial \theta_2} & \cdots & \frac{\partial z}{\partial \theta_6} \end{bmatrix}, \ \mathbf{J_{gw}} = \begin{bmatrix} \frac{\partial \theta_x}{\partial \theta_1} & \frac{\partial \theta_x}{\partial \theta_2} & \cdots & \frac{\partial \theta_x}{\partial \theta_6} \\ \frac{\partial \theta_y}{\partial \theta_1} & \frac{\partial \theta_y}{\partial \theta_2} & \cdots & \frac{\partial \theta_y}{\partial \theta_6} \\ \frac{\partial \theta_z}{\partial \theta_1} & \frac{\partial \theta_z}{\partial \theta_2} & \cdots & \frac{\partial \theta_z}{\partial \theta_6} \end{bmatrix}, \ \dot{\boldsymbol{q}} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix}$$

As a result, the geometric Jacobian matrix based on frame0 ${}^{0}\mathbf{J_g}$ is derived as

$$\dot{\boldsymbol{x}} = {}^{0}\mathbf{J_g} \cdot \dot{\boldsymbol{q}}, \ \ \dot{\boldsymbol{q}} = {}^{0}\mathbf{J_g}^{-1} \cdot \dot{\boldsymbol{x}} \tag{3.6}$$

where

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix}_{\{0\}}, \ \dot{\boldsymbol{q}} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix} \tag{3.7}$$

There is a further derivation. In terms of the Jacobian matrix, we can derive the other property.

$$\boldsymbol{\tau} = {}^{0}\mathbf{J_g^{\top}} \cdot \boldsymbol{f} \quad \Leftrightarrow \quad \begin{bmatrix} \tau_{\theta_1} \\ \tau_{\theta_2} \\ \tau_{\theta_3} \\ \tau_{\theta_4} \\ \tau_{\theta_5} \\ \tau_{\theta_6} \end{bmatrix} = {}^{0}\mathbf{J_g^{\top}} \cdot \begin{bmatrix} f_x \\ f_y \\ f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \tag{3.8}$$

where $\boldsymbol{\tau}$ is the vector of joints' torques and $\boldsymbol{f}$ is the vector composed of inertial forces and torques of the robot arm.

Geometric Jacobian Based on Frame$\{6\}$

Why we need geometric Jacobian based on frame$\{6\}$ is that in section 4.3.1 we will apply admittance control and it will use F/T sensor mounted on frame$\{6\}$ instead of frame$\{0\}$ to detect forces and torques. To start with Equation 3.6

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix}_{\{0\}} = {}^{\mathbf{0}}\mathbf{J}_{\mathbf{g}} \cdot \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix} \tag{3.9}$$

Then, left-multiply a matrix.

$$\begin{bmatrix} {}^0_6\mathbf{R} & 0_{3\times3} \\ 0_{3\times3} & {}^0_6\mathbf{R} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix}_{\{0\}} = \begin{bmatrix} {}^0_6\mathbf{R} & 0_{3\times3} \\ 0_{3\times3} & {}^0_6\mathbf{R} \end{bmatrix} {}^{\mathbf{0}}\mathbf{J}_{\mathbf{g}} \cdot \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix} \tag{3.10}$$

According to the transformation coordinate relationship between frame$\{0\}$ and frame$\{6\}$,

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix}_{\{6\}} = \begin{bmatrix} {}^0_6\mathbf{R} & 0_{3\times3} \\ 0_{3\times3} & {}^0_6\mathbf{R} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix}_{\{0\}} \tag{3.11}$$

substitute it into Eq 3.10, we can observe an essential equation.

$$
{}^{6}\mathbf{J_g} = \begin{bmatrix} {}^{0}_{6}\mathbf{R} & 0_{3\times3} \\ 0_{3\times3} & {}^{0}_{6}\mathbf{R} \end{bmatrix} \cdot {}^{0}\mathbf{J_g} \tag{3.12}
$$

Notably,

$$
\dot{\boldsymbol{x}} = {}^{6}\mathbf{J_g} \cdot \dot{\boldsymbol{q}}, \;\; \dot{\boldsymbol{q}} = {}^{6}\mathbf{J_g^{-1}} \cdot \dot{\boldsymbol{x}} \tag{3.13}
$$

where

$$
\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix}_{\{6\}}, \; \dot{\boldsymbol{q}} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix}
$$

Analytical Jacobian

The linear velocity of analytical Jacobian and geometric Jacobian is the same as shown in Eq 3.3.3. Nevertheless, as for angular velocity, analytical Jacobian takes the orientation (α,β,γ) of the end effector into consideration. First of all, we investigate the relationship between the axis angle and the orientation of the end effector as following.

$$
\begin{aligned}
\begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \end{bmatrix} &= \begin{bmatrix} \alpha \\ 0 \\ 0 \end{bmatrix} + R_x(\alpha) \begin{bmatrix} 0 \\ \beta \\ 0 \end{bmatrix} + R_x(\alpha)R_y(\beta) \begin{bmatrix} 0 \\ 0 \\ \gamma \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & S_\beta \\ 0 & C_\alpha & -S_\alpha C_\beta \\ 0 & S_\alpha & C_\alpha C_\beta \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}
\end{aligned} \tag{3.14}
$$

Then, utilize this generalized vector to get its Jacobian matrix $\mathbf{J_{we}}$.

$$\begin{bmatrix} \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix} = \mathbf{J_{we}} \cdot \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \tag{3.15}$$

where

$$\mathbf{J_{we}} = \begin{bmatrix} 1 & \gamma C_\beta & S_\beta \\ -\beta S_\alpha - \gamma C_\alpha C_\beta & C_\alpha + \gamma S_\alpha S_\beta & -C_\beta S_\alpha \\ \beta C_\alpha - \gamma C_\beta S_\alpha & S_\alpha - \gamma C_\alpha S_\beta & C_\alpha C_\beta \end{bmatrix} \tag{3.16}$$

Therefore,

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & \mathbf{J_{we}} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \tag{3.17}$$

Finally, we obtain the relationship between geometric and analytical Jacobian.

$$\mathbf{J_g} = \begin{bmatrix} \mathbf{I}_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & \mathbf{J_{we}} \end{bmatrix} \mathbf{J_a}, \; \mathbf{J_g} = \begin{bmatrix} \mathbf{I}_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & \mathbf{J_{we}^{-1}} \end{bmatrix} \mathbf{J_a} \tag{3.18}$$

3.4 Coordinate Transformation of the Robot Arm

So far, with the forward and inverse kinematics, the robot arm can translate and rotate around frame$\{6\}$. However, we do not see the origin of the frame$\{6\}$ as an operating point. Because the F/T sensor and a detachable end effector will be both mounted on the wrist, the tooltip position is what we want. That means we should let the robot arm know how to translate and rotate in frame$\{$T$\}$ instead of frame$\{6\}$. If we have translation and rotation information of the tooltip, there is an easy way to directly give the above information to the robot arm. SetTRF (x,y,z,α,β,γ) is the command of the robot arm, whose (x,y,z) is translation vector and (α,β,γ) is rotation vector in representation of Euler angle .

To obtain translation and rotation vector, we respectively introduce Tool Center Point in section 3.4.1 to find the translation vector and propose an approach in section 3.4.2 to find the rotation vector.

3.4.1 Translation Analysis - Tool Center Point

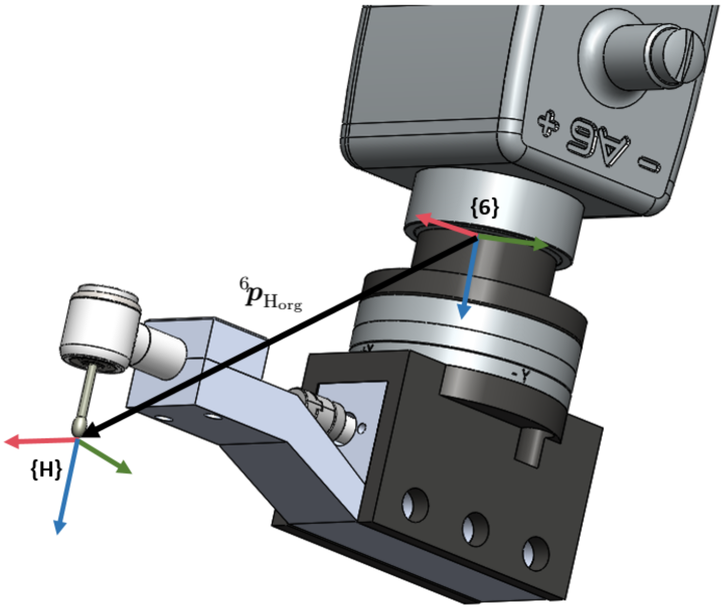

Figure 3.3: Schematic diagram for Tool Center Point. The translation vector $^{6}\boldsymbol{p}_{\mathrm{H_{org}}}$ denotes the origin position relative to the frame$\{6\}$.

Tool Center Point (TCP) is a critical problem for robot arm control [3]. In the previous section, we have calculated the forward and inverse kinematics of the robot arm. By Calculating kinematics, we can keep track of the origin of the frame$\{6\}$, which is observed from the base frame. The robot arm has the capability to translate and rotate with the origin of the frame$\{6\}$. The overhead motions are like a remote center motion (RCM). We should find the position of the tooltip and make it be an RCM point. Nevertheless, it's inefficient to recalculate the transformation matrix via mechanism dimension when changing an end effector or a tool (root canal reamer).

In order to overcome this problem, we interpret the four-points method to obtain the tooltip position, which is also the translation vector.
From Fig 3.2, we can obtain the following transformation matrix,

$$ {}^{\mathrm{B}}_{\mathrm{H}}\mathbf{T} = {}^{\mathrm{B}}_{6}\mathbf{T} \cdot {}^{6}_{\mathrm{H}}\mathbf{T} \tag{3.19} $$

and it can be rewritten as

$$ \begin{aligned} \begin{bmatrix} {}^{\mathrm{B}}_{\mathrm{H}}\mathbf{R} & {}^{\mathrm{B}}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} \\ 0 & 1 \end{bmatrix} &= \begin{bmatrix} {}^{\mathrm{B}}_{6}\mathbf{R} & {}^{\mathrm{B}}\boldsymbol{p}_{6_{\mathrm{org}}} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{6}_{\mathrm{H}}\mathbf{R} & {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} {}^{\mathrm{B}}_{6}\mathbf{R} \cdot {}^{6}_{\mathrm{H}}\mathbf{R} & {}^{\mathrm{B}}_{6}\mathbf{R} \cdot {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} + {}^{\mathrm{B}}\boldsymbol{p}_{6_{\mathrm{org}}} \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{3.20} $$

Consequently, we get a crucial equation:

$$ {}^{\mathrm{B}}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} = {}^{\mathrm{B}}_{6}\mathbf{R} \cdot {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} + {}^{\mathrm{B}}\boldsymbol{p}_{6_{\mathrm{org}}} \tag{3.21} $$

Now, we move the tooltip to a fixed point with four different poses, including position and orientation. Then, we will get four different rotation matrices and vectors in real-time.

$$ \begin{aligned} {}^{\mathrm{B}}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} &= {}^{\mathrm{B}}_{\mathrm{F}}\mathbf{R}^{1} \cdot {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} + {}^{\mathrm{B}}\boldsymbol{p}^{1}_{\mathrm{F}_{\mathrm{org}}} \\ &= {}^{\mathrm{B}}_{\mathrm{F}}\mathbf{R}^{2} \cdot {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} + {}^{\mathrm{B}}\boldsymbol{p}^{2}_{\mathrm{F}_{\mathrm{org}}} \\ &= {}^{\mathrm{B}}_{\mathrm{F}}\mathbf{R}^{3} \cdot {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} + {}^{\mathrm{B}}\boldsymbol{p}^{3}_{\mathrm{F}_{\mathrm{org}}} \\ &= {}^{\mathrm{B}}_{\mathrm{F}}\mathbf{R}^{4} \cdot {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} + {}^{\mathrm{B}}\boldsymbol{p}^{4}_{\mathrm{F}_{\mathrm{org}}} \end{aligned} \tag{3.22} $$

In order to extract ${}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}}$ from Eq.3.22, we subtract the second to forth equation from the first equation.

$$ \begin{bmatrix} {}^{\mathrm{B}}_{6}\mathbf{R}^{1} - {}^{\mathrm{B}}_{6}\mathbf{R}^{2} \\ {}^{\mathrm{B}}_{6}\mathbf{R}^{1} - {}^{\mathrm{B}}_{6}\mathbf{R}^{3} \\ {}^{\mathrm{B}}_{6}\mathbf{R}^{1} - {}^{\mathrm{B}}_{6}\mathbf{R}^{4} \end{bmatrix} \cdot {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org}}} = \begin{bmatrix} {}^{\mathrm{B}}\boldsymbol{p}^{2}_{6_{\mathrm{org}}} - {}^{\mathrm{B}}\boldsymbol{p}^{1}_{6_{\mathrm{org}}} \\ {}^{\mathrm{B}}\boldsymbol{p}^{3}_{6_{\mathrm{org}}} - {}^{\mathrm{B}}\boldsymbol{p}^{1}_{6_{\mathrm{org}}} \\ {}^{\mathrm{B}}\boldsymbol{p}^{4}_{6_{\mathrm{org}}} - {}^{\mathrm{B}}\boldsymbol{p}^{1}_{6_{\mathrm{org}}} \end{bmatrix} \tag{3.23} $$

where we define

$$\mathbf{R} = \begin{bmatrix} {}^{\mathrm{B}}_{6}\mathbf{R}^{1} - {}^{\mathrm{B}}_{6}\mathbf{R}^{2} \\ {}^{\mathrm{B}}_{6}\mathbf{R}^{1} - {}^{\mathrm{B}}_{6}\mathbf{R}^{3} \\ {}^{\mathrm{B}}_{6}\mathbf{R}^{1} - {}^{\mathrm{B}}_{6}\mathbf{R}^{4} \end{bmatrix}_{9\times 3}, \boldsymbol{p} = \begin{bmatrix} {}^{\mathrm{B}}\boldsymbol{p}^{2}_{6_{\mathrm{org}}} - {}^{\mathrm{B}}\boldsymbol{p}^{1}_{6_{\mathrm{org}}} \\ {}^{\mathrm{B}}\boldsymbol{p}^{3}_{6_{\mathrm{org}}} - {}^{\mathrm{B}}\boldsymbol{p}^{1}_{6_{\mathrm{org}}} \\ {}^{\mathrm{B}}\boldsymbol{p}^{4}_{6_{\mathrm{org}}} - {}^{\mathrm{B}}\boldsymbol{p}^{1}_{6_{\mathrm{org}}} \end{bmatrix}_{9\times 1}$$

Therefore,

$$\begin{aligned} {}^{6}\boldsymbol{p}_{\mathrm{H_{org}}} &= \mathbf{R}^{\dagger} \cdot \boldsymbol{p} \\ &= \left(\mathbf{R}^{\top}\mathbf{R}\right)^{-1} \mathbf{R}^{\top} \cdot \boldsymbol{p} \end{aligned}$$

As a result, we can utilize the four-points method to obtain the translation vector.

3.4.2 Rotation Analysis

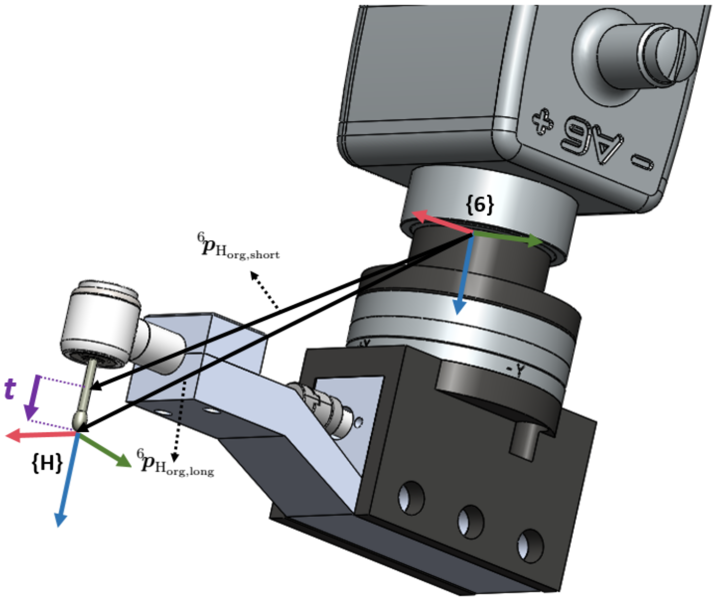
+ A6 -
{6}
${}^{6}\boldsymbol{p}_{\mathrm{H_{org,short}}}$
$\boldsymbol{t}$
{H}
${}^{6}\boldsymbol{p}_{\mathrm{H_{org,long}}}$
Y
-Y

Figure 3.4: Schematic diagram for obtaining the tool vector.

We are turning now to the discussion about the rotation vector. Above all, we have to find the vector of tool insertion direction $\boldsymbol{t}$. We can obtain the translation vector from the origin of frame$\{6\}$ to the tooltip through the TCP method. Accordingly, we use two root canal files with different lengths and apply the TCP method to separately obtain two vectors illustrated as Fig 3.4. Hence,

$$\boldsymbol{t} = {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org,long}}} - {}^{6}\boldsymbol{p}_{\mathrm{H}_{\mathrm{org,short}}} \tag{3.24}$$

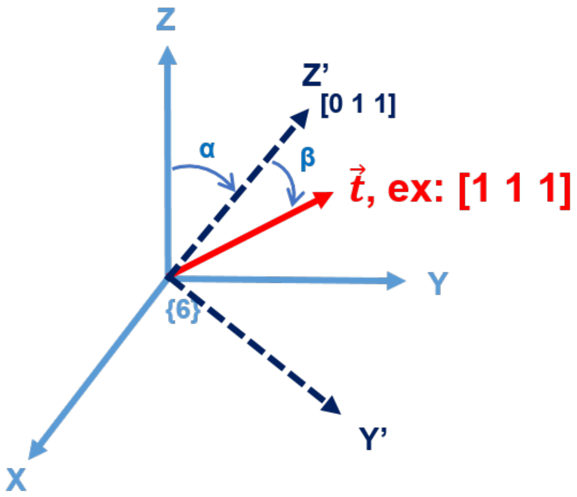

Figure 3.5: Illustration of finding the rotation matrix

For analyzing it easily, we depict it in Fig 3.5. Note that here we only discuss rotation, so we assume that we have done translation and matched the frame$\{\mathrm{S}\}$ with frame$\{6\}$. Because we hope to send a Z-axis command to achieve tool insertion, we should align the original Z-axis to the target vector. Nevertheless, Z-axis alignment without other restrictions will produce many solutions. We choose one of the solutions to align Z-axis to the target vector. According to the figure, we assume the target vector $\boldsymbol{t}$ is $[x, y, z]$, whose projection to yz-plane $\mathrm{proj}_{(\mathrm{y-z})}\boldsymbol{t}$ is $[0, y, z]$. Initially, we rotate α degree around X-axis to make original Z-axis align the projection $[0, y, z]$. Next, we rotate β degree around Y’ axis and finally align the original Z-axis to the target vector [1,1,1]. The following equation

$${}^{\mathrm{T}}_{6}\mathbf{R} = \mathbf{R}_{\mathbf{x}}(\alpha) \cdot \mathbf{R}_{\mathbf{y}}(\beta) \tag{3.25}$$

where

$$\begin{aligned} \alpha &= -\mathrm{sign}(t_y) \cdot \cos^{-1}\left(\frac{\hat{k} \cdot \mathrm{proj}_{(\mathrm{y-z})}\boldsymbol{t}}{\left\|\hat{k}\right\| \cdot \left\|\mathrm{proj}_{(\mathrm{y-z})}\boldsymbol{t}\right\|}\right) \\ \beta &= \mathrm{sign}(t_x) \cdot \cos^{-1}\left(\frac{\mathbf{t} \cdot \mathrm{proj}_{(\mathrm{y-z})}\boldsymbol{t}}{\|\mathbf{t}\| \cdot \left\|\mathrm{proj}_{(\mathrm{y-z})}\boldsymbol{t}\right\|}\right) \end{aligned} \tag{3.26}$$

Assume $\boldsymbol{t} = [x, y, z]$,

$$\begin{aligned} \alpha &= -\mathrm{sign}(y) \cdot \cos^{-1}\left(\frac{z^2}{\sqrt{y^2 + z^2}}\right) \mathrm{rad} \\ \beta &= \mathrm{sign}(x) \cdot \cos^{-1}\left(\frac{y^2 + z^2}{\sqrt{x^2 + y^2 + z^2}\sqrt{x^2 + y^2 + z^2}}\right) \mathrm{rad} \end{aligned} \tag{3.27}$$

α and β are Euler angles, which meet the command demand.

In this section, we have demonstrated two key aspects of reference frame changing of the robot arm. It's easy to input the results of section 3.4.1 and section 3.4.2 via the command setTRF, the robot arm will recognize frame$\{\mathrm{T}\}$. Having discussed how to combine a robot arm with an end effector, the next section addresses ways of combining an F/T sensor.

Chapter 4

Force-Guided Robot Alignment

This chapter follows from the previous issue about integration. We continue to demonstrate some technical solutions for system integration. On top of that, the Forces and Torques (F/T) sensor will be included to discuss. Therefore, you can see the chapter as an operating manual when you simultaneously own a robot arm, an F/T sensor, and an end effector. First of all, we will explain why we need to use the F/T sensor in our project in section 4.1. Furthermore, we will introduce how to compensate the gravity affection while moving the robot in senction 4.2. Admittance control based on F/T sensor will be described in section 4.3.1. Coordinate transformation of F/T sensor will be interpreted in section 4.4.1. Last but not least, we will discuss affection of setting admittance control parameters in section 4.5.

4.1 Problem Definition

In modern robot-assisted surgery, to simulate the eyes of doctors, image processing is often considered. With the image's information, the surgeon can obtain real-time data that humans can not observe clearly. Surgeons use the image process to do many things, such as tracking whether the patient is moving, positioning an incision point, and navigating an insertion path. As for root canal treatment, the endodontist checks the number of root canals with computed tomography. Before starting to clean the pulp, the endodontist tries to insert root files into root canals. Then the endodontist takes the patient to have computed tomography(CT). CT scan can clearly show the tooth with the root files. With the preoperative image, the endodontist can check whether all root canals are all found and determine all root canals' lengths. However, the above image application is preoperative. The endodontist can only use the preoperative image to guess the direction of the insertion. The intraoperative image processing is absent due to the dimension restriction of a tooth. It is difficult to get a real-time image when the dentist inserts a root file into the opened tooth because it will obstacle the observation. Besides, the diameter of a root canal is 1 mm and is smaller than a root file. It is impossible to get the image from any shooting angle.

To solve the first problem - without image processing, we propose a method based on force-guided. We surveyed the peg-in-hole based on F/T sensor feedback [4] because the cleaning procedure is similar to it. On the other hand, we provide two operating modes - ”Dragging Mode” and ”Self-Alignment Mode” to cooperate with a dentist. In ”Dragging Mode”, we can hold the handpiece to the desired position and orientation. We hope to use this mode to let the dentist move the handpiece to the injured tooth. In ”Self-Alignment Mode”, the handpiece can automatically align the root canal direction even though there are many complicated conditions with different teeth.

4.2 Integration of F/T sensor

Gravity compensation is a critical technical issue when combining an F/T sensor with a robot arm and an end effector [5]. Fundamentally, we should receive stable data when a static F/T sensor bears the same load or force. Nevertheless, our F/T sensor is installed on the robot arm and move with the pose of the robot arm. On account of its mobility, the gravity of the end effector will significantly affect

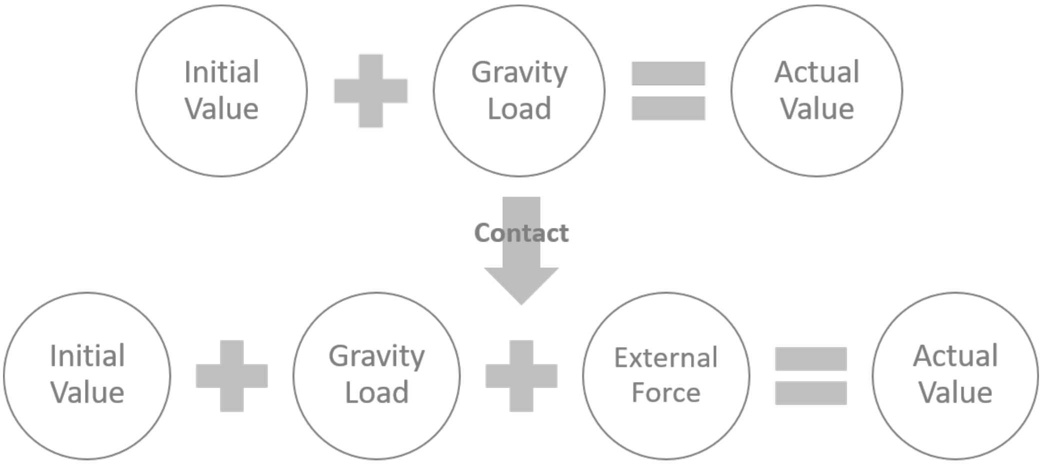
Initial Value
Gravity Load
Actual Value
Contact
Initial Value
Gravity Load
External Force
Actual Value

Figure 4.1: Data Analysis of F/T sensor.

the actual value we received. Moreover, starting without resetting the F/T sensor to zero would lead to an initial value. If we could not analyze the actual value to an initial value and a gravity load, we would obtain an unstable actual value, not to mention the external force caused by contact force.

Therefore, we illustrate a method, which is to analyze the actual value of the F/T sensor to an initial value and gravity load in real-time. It's worth noting that with this approach, we can get the installation angle between the F/T sensor and the robot arm, including assembly error, and we no longer need to reset the F/T sensor to zero every time.

The Centroid Position of End Effector

To start with the first equation in Figure 4.1,

$$\begin{aligned}&\text{Initial value} + \text{Gravity Load} = \text{Actual Value}\\ &\Rightarrow \begin{cases}\boldsymbol{f_0} + \boldsymbol{f_g} &= \boldsymbol{f}\\ \boldsymbol{\tau_0} + \boldsymbol{\tau_g} &= \boldsymbol{\tau}\end{cases} \Rightarrow \begin{cases} f_{0x} + f_{gx} &= f_x\\ f_{0y} + f_{gz} &= f_y\\ f_{0z} + f_{gy} &= f_z\\ \tau_{0x} + \tau_{gx} &= \tau_x\\ \tau_{0y} + \tau_{gy} &= \tau_y\\ \tau_{0z} + \tau_{gz} &= \tau_z\end{cases}\end{aligned} \tag{4.1}$$

where $\boldsymbol{f}$ and $\boldsymbol{\tau}$ are force and torque vector respectively.

And, by terms of moment arm formula,

$$\because \boldsymbol{\tau_g} = \boldsymbol{r} \times \boldsymbol{f_g} \tag{4.2}$$

where $\boldsymbol{r}$ denotes the centroid position of the end effector in the sensor frame.

$$\begin{aligned}\therefore\ \boldsymbol{\tau} &= \boldsymbol{\tau_0} + \boldsymbol{\tau_g}\\ &= \boldsymbol{\tau_0} + \boldsymbol{r} \times \boldsymbol{f_g}\end{aligned} \tag{4.3}$$

Then, Substitute the first line of Equation 4.1 into the above equation, we will obtain

$$\boldsymbol{\tau} = \boldsymbol{\tau_0} + \boldsymbol{p} \times (\boldsymbol{f} - \boldsymbol{f_0}) \tag{4.4}$$

, which could be extended as

$$\begin{cases}\tau_x = \tau_{0x} + (f_z - f_{0z}) \cdot y - (f_y - f_{0y}) \cdot z\\ \tau_y = \tau_{0y} + (f_x - f_{0x}) \cdot z - (f_z - f_{0z}) \cdot x\\ \tau_z = \tau_{0z} + (f_y - f_{0y}) \cdot x - (f_x - f_{0x}) \cdot y\end{cases} \tag{4.5}$$

and be overwritten as

$$\begin{bmatrix}\tau_x\\ \tau_y\\ \tau_z\end{bmatrix} = \begin{bmatrix}0 & f_z & -f_y & 1 & 0 & 0\\ -f_z & 0 & f_x & 0 & 1 & 0\\ f_y & -f_x & 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x\\ y\\ z\\ k_1\\ k_2\\ k_3\end{bmatrix} \tag{4.6}$$

where

$$\begin{cases} k_1 = \tau_{0x} - (f_{0z}\cdot y + f_{0y}\cdot z)\\ k_2 = \tau_{0y} - (f_{0x}\cdot z + f_{0z}\cdot x) \quad \text{are all constant}\\ k_3 = \tau_{0z} - (f_{0y}\cdot x + f_{0x}\cdot y)\end{cases} \tag{4.7}$$

With extracting $[x, y, z, k_1, k_2, k_3]$ in mind, we move the robot arm to n $(n \geq 3)$ positions with different poses. By recording n torque vectors and corresponding n force vectors from F/T sensor, we can expand Equation 4.6 as

$$\begin{bmatrix}\tau_x^1\\ \tau_y^1\\ \tau_z^1\\ \tau_x^2\\ \tau_y^2\\ \tau_z^2\\ \vdots\\ \tau_x^n\\ \tau_y^n\\ \tau_z^n\end{bmatrix} = \begin{bmatrix}0 & f_z^1 & -f_y^1 & 1 & 0 & 0\\ -f_z^1 & 0 & f_x^1 & 0 & 1 & 0\\ f_y^1 & -f_x^1 & 0 & 0 & 0 & 1\\ 0 & f_z^2 & -f_y^2 & 1 & 0 & 0\\ -f_z^2 & 0 & f_x^2 & 0 & 1 & 0\\ f_y^2 & -f_x^2 & 0 & 0 & 0 & 1\\ & & \vdots & & & \\ 0 & f_z^3 & -f_y^3 & 1 & 0 & 0\\ -f_z^3 & 0 & f_x^3 & 0 & 1 & 0\\ f_y^3 & -f_x^3 & 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x\\ y\\ z\\ k_1\\ k_2\\ k_3\end{bmatrix} \tag{4.8}$$

, which is defined as

$$\boldsymbol{m}_{(3n\times1)} = \mathbf{F}_{(3n\times6)}\cdot\boldsymbol{p}_{(6\times1)} \tag{4.9}$$

As a consequence of the full column rank of $\mathbf{F}$, we can apply Moore-Penrose pseudoinverse. Then,

$$\begin{aligned} \boldsymbol{p} &= \mathbf{F}^{\dagger} \cdot \boldsymbol{m} \\ &= \left(\mathbf{F}^{\top}\mathbf{F}\right)^{-1} \mathbf{F}^{\top} \cdot \boldsymbol{m} \end{aligned}$$

From now on, we have already known the centroid position of end effector in the sensor frame and values of the constants k_1, k_2, k_3.

Gravity Compensation and Initial Value Reset

Next, let us return to the first equation in Figure 4.1. We continue to use this formula and contemplate it from the perspective of coordinate transformation relation. Here we hypothesize that the end effector weighs g kilograms relative to the frame$\{0\}$ which is also the world frame. That means the gravity vector of the end effector ${}^{\mathbf{0}}\boldsymbol{g}$ is $[0, 0, -g]$ in the frame$\{0\}$. Then, we can derive it as following.

$$\begin{aligned} &\text{Initial value} + \text{Gravity Load} = \text{Actual Value} \\ &\Rightarrow \boldsymbol{f}_{\mathbf{0}} + {}^{\mathrm{S}}_{6}\mathbf{R} \cdot {}^{6}_{0}\mathbf{R} \cdot {}^{\mathbf{0}}\boldsymbol{g} = \boldsymbol{f} \end{aligned} \tag{4.10}$$

which is relative to frame $\{6\}$. Hence, we assume an installation angle θ including assembly error.

$${}^{\mathrm{S}}_{6}\mathbf{R} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.11}$$

Besides, we can easily calculate ${}^{6}_{0}\mathbf{R}$ from Equation 3.3.2.

$$\begin{aligned} {}^{6}_{0}\mathbf{R} &= {}^{0}_{6}\mathbf{R}^{\top} \\ &:= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \end{aligned} \tag{4.12}$$

Therefore,

$$\begin{bmatrix} f_{0x} \\ f_{0y} \\ f_{0z} \end{bmatrix} + \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \tag{4.13}$$

Further, we rewrite it as

$$\begin{bmatrix} -r_{13} & -r_{23} & 0 & 1 & 0 & 0 \\ -r_{23} & r_{13} & 0 & 0 & 1 & 0 \\ 0 & 0 & -r_{33} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} g\cos(\theta) \\ g\sin(\theta) \\ g \\ f_{0x} \\ f_{0y} \\ f_{0z} \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \tag{4.14}$$

In the same way as Equation 4.6, we can extract $[g\cos(\theta), g\sin(\theta), g, f_{0x}, f_{0y}, f_{0z}]$ by the least square solution. Note that, we can simultaneously record these data when using the method in Equation 4.6. By recording $n(n \geq 3)$ third columns of ration matrices and corresponding n force vectors from F/T sensor, we can expand Equation 4.14 as

$$\begin{bmatrix} -r_{13}^1 & -r_{23}^1 & 0 & 1 & 0 & 0 \\ -r_{23}^1 & r_{13}^1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -r_{33}^1 & 0 & 0 & 1 \\ -r_{13}^2 & -r_{23}^2 & 0 & 1 & 0 & 0 \\ -r_{23}^2 & r_{13}^2 & 0 & 0 & 1 & 0 \\ 0 & 0 & -r_{33}^2 & 0 & 0 & 1 \\ & & \vdots & & & \\ -r_{13}^n & -r_{23}^n & 0 & 1 & 0 & 0 \\ -r_{23}^n & r_{13}^n & 0 & 0 & 1 & 0 \\ 0 & 0 & -r_{33}^n & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} g\cos(\theta) \\ g\sin(\theta) \\ g \\ f_{0x} \\ f_{0y} \\ f_{0z} \end{bmatrix} = \begin{bmatrix} f_{0x}^1 \\ f_{0y}^1 \\ f_{0z}^1 \\ f_{0x}^2 \\ f_{0y}^2 \\ f_{0z}^2 \\ \vdots \\ f_{0x}^n \\ f_{0y}^n \\ f_{0z}^n \end{bmatrix} \tag{4.15}$$

, which is defined as

$$\mathrm{M}_{(3n\times 6)} \cdot \boldsymbol{p}_{(6\times 1)} = \boldsymbol{f}_{(3n\times 1)} \tag{4.16}$$

As a consequence of the full column rank of $\mathbf{M}$, we can apply Moore-Penrose pseudoinverse. Then,

$$\begin{aligned} \boldsymbol{p} &= \mathrm{M}^{\dagger} \cdot \boldsymbol{f} \\ &= \left(\mathbf{M}^{\top}\mathbf{M}\right)^{-1} \mathbf{M}^{\top} \cdot \boldsymbol{f} \end{aligned}$$

Apparently, we can directly obtain g, f_{0x}, f_{0y}, f_{0z}. Afterwards, we substitute them into Equation 4.7 to calculate

$$\begin{cases} \tau_{0x} = k_1 + (f_{0z} \cdot y + f_{0y} \cdot z) \\ \tau_{0y} = k_2 + (f_{0x} \cdot z + f_{0z} \cdot x) \\ \tau_{0z} = k_3 + (f_{0y} \cdot x + f_{0x} \cdot y) \end{cases} \tag{4.17}$$

Finally, we successfully obtain the weight of the end effector g, the initial value of the F/T sensor $[f_{0x}, f_{0y}, f_{0z}, \tau_{0x}, \tau_{0y}, \tau_{0z}]$.

As for installation angle θ including assembly error, there is an further discussion. Undoubtedly, we can derive it as

$$\theta = \cos^{-1}\left(\frac{g\cos(\theta)}{g}\right) \text{ or } \sin^{-1}\left(\frac{g\sin(\theta)}{g}\right) \tag{4.18}$$

In theory, By either the arccos function or arcsin function, we can derive the same value. However, we estimate $[g\cos(\theta), g\sin(\theta), g, f_{0x}, f_{0y}, f_{0z}]$ by using the least square solution which produce a approximated answer rather than the correct answer. A subtle bias caused by the least square solution will be enlarged through the arc function. Thankfully, we originally designed an adapter to connect the robot arm and F/T sensor, and the installation angle is exactly zero degrees. Therefore, here we only need to concern about the subtle bias result from assembly errors. Here we take the same bias and compare arccos to arcsin. Obviously, if we separately give the same bias (0.1 rad) into the arccos and arcsin function, the arccos function will enlarge the bias significantly larger than the arcsin function. Hence we should use

$$\theta = \sin^{-1}\left(\frac{g\sin(\theta)}{g}\right) \tag{4.19}$$

Table 4.1: Arc-function Comparison

bias n (rad)	$\sin^{-1}(n)$ (degree)	$\cos^{-1}(1-n)$ (degree)
0	0	0
0.001	0.057	2.56
0.01	0.57	8.11
0.1	5.7	25.84

Notably, as a result of the zero installation angle we initially designed from Table 4.1, we could infer using the arcsin function. In contrast, if the installation angle is not zero, the above assumption will be invalid.

Ultimately, we have successfully dissected the actual value of the F/T sensor into the initial value of the F/T sensor and the gravity of the end effector.

External Force Evaluation

Next, we continue to review the second equation in Figure 4.1.

$$\text{Initial value} + \text{Gravity Load} + \text{External Force} = \text{Actual Value}$$
$$\Rightarrow \begin{cases} \boldsymbol{f}_0 + \boldsymbol{f}_g + \boldsymbol{f}_e & = \boldsymbol{f} \\ \boldsymbol{\tau}_0 + \boldsymbol{\tau}_g + \boldsymbol{\tau}_e & = \boldsymbol{\tau} \end{cases} \tag{4.20}$$

Therefore,

$$\begin{aligned} \boldsymbol{f}_e &= \boldsymbol{f} - \boldsymbol{f}_0 - \boldsymbol{f}_g \\ &= \boldsymbol{f} - \boldsymbol{f}_0 - {}^{S}_{6}\mathbf{R} \cdot {}^{6}_{0}\mathbf{R} \cdot {}^{0}\boldsymbol{g} \\ \boldsymbol{\tau}_e &= \boldsymbol{\tau} - \boldsymbol{\tau}_0 - \boldsymbol{\tau}_g \\ &= \boldsymbol{\tau} - \boldsymbol{\tau}_0 - \boldsymbol{r} \times \boldsymbol{f}_g \end{aligned} \tag{4.21}$$

$$\Rightarrow \begin{cases} f_{ex} = f_x - f_{0x} + g\cos(\theta)r_{13} + g\sin(\theta)r_{23} \\ f_{ey} = f_y - f_{0y} - g\sin(\theta)r_{13} + g\cos(\theta)r_{23} \\ f_{ez} = f_z - f_{0z} + gr_{33} \\ \tau_{ex} = f_x - \tau_{0x} - (g_z \cdot y - g_y \cdot z) \\ \tau_{ey} = f_y - \tau_{0y} - (g_x \cdot z - g_z \cdot x) \\ \tau_{ez} = f_z - \tau_{0z} - (g_y \cdot x - g_x \cdot y) \end{cases}$$

4.3 Dragging Mode

In the hope that dentist could drag our system to an infected tooth by holding the end effector, we usher in admittance control based on F/T sensor.

4.3.1 Admittance Control based on F/T sensor

Control Scheme

Admittance control makes the robot move like a spring-mass-damper system. Forces and torques can be mapped into the movements such as position or velocity. Most importantly, admittance control enables a robot arm to cooperate with humans in a safe work environment. Since Meca500 is an industrial robot arm without admittance control, we subsequently combine the robot arm with the F/T sensor to adopt admittance control. With force and torque feedback, the F/T sensor makes Meca500 be like a collaborative robot arm. Therefore, we propose a control scheme depicted as Figure 4.2. It's worth noting that in this approach, the admittance control function is triggered by the end effector mounted on the F/T sensor instead of detecting each wrist torque of the robot arm.

A standard equation of admittance control is represented as Equation 4.22. The values we obtain from the F/T sensor are $[f_x, f_y, f_z, \tau_x, \tau_y, \tau_z]$, whose forces $[f_x, f_y, f_z]$ are related to the translations $[x, y, z]$ and torques $[\tau_x, \tau_y, \tau_z]$ are related

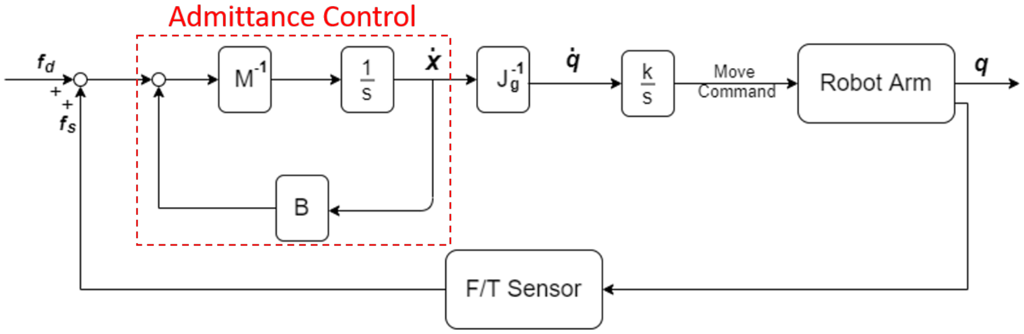

Figure 4.2: Control scheme. $\boldsymbol{f_d}$ denotes the desired forces and torques vector. $\boldsymbol{f_s}$ denotes the real value detected by F/T sensor and is also a forces and torques vector. $\dot{\boldsymbol{x}}$ denotes $[\dot{x}, \dot{y}, \dot{z}, \dot{\theta}_x, \dot{\theta}_y, \dot{\theta}_z]$. $\mathbf{J_g}$ denotes the geometric Jacobian matrix. $\dot{\boldsymbol{q}}$ denotes $[\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5, \dot{\theta}_6]$. $\boldsymbol{q}$ denotes $[\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]$.

to the axis angle $[\theta_x, \theta_y, \theta_z]$.

$$\begin{bmatrix} x \\ y \\ z \\ \theta_x \\ \theta_y \\ \theta_z \end{bmatrix} = \frac{1}{\mathbf{M}\mathrm{S}^2 + \mathbf{B}\mathrm{S} + \mathbf{K}} \begin{bmatrix} f_x \\ f_y \\ f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \tag{4.22}$$

Selection of Admittance Control Model

In our proposed approach, we omit parameter $\mathbf{K}$ which is relevant to spring stiffness, considering that it's not necessary to bounce such as a spring. Therefore, our system should behave like a mass-damper system as following.

$$\begin{bmatrix} x \\ y \\ z \\ \theta_x \\ \theta_y \\ \theta_z \end{bmatrix} = \frac{1}{\mathbf{M}\mathbf{S}^2 + \mathbf{B}\mathbf{S}} \begin{bmatrix} f_x \\ f_y \\ f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \tag{4.23}$$

where $\mathbf{M}, \mathbf{B}, \mathbf{K}$ are diagonal positive definite matrices. Affections of these parameters will be discussed in section 4.5.

Robot Command Decision

After determining our admittance control model, we should select a correspondent command to move the robot. However, there are many commands of moving the robot arm, as shown in Table 4.2.

Table 4.2: Moving commands in Meca500

Types	Commands	Input parameters
Position	MoveJoints	$\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6$
	MovePose	$x, y, z, \alpha, \beta, \gamma$
Velocity	MoveJointsVel	$\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5, \dot{\theta}_6$
	MoveLinVelTRF	$\dot{x}, \dot{y}, \dot{z}, \dot{\theta}_x, \dot{\theta}_y, \dot{\theta}_z$

Considering that the singularity problem is an imperative subject in robotics, we intend to use MoveJoints or MoveJointsVel to set the axes' angles directly. Despite that it's easier to implement admittance control via other commands, the system would touch the singularity point at any time. It undoubtedly exposes patients to danger because of the uncertainty of the system. In our case, we could detect whether it is a singularity or not before commanding the robot. Therefore, position command - MoveJoints and velocity command - MoveJointsVel is our option. Why we choose the command MoveJoints is that Meca500 has a default time-out value to ensure its safety. Even though we could set this value from 0.001 to 2 seconds, Meca500 is still restricted to move with this value. For example, the value of time-out is set 0.1 sec. While the robot receives a command, the robot move for 0.1 sec then immediately stops. It's not easy to control via velocity command due to this default property.

Finally, we designate MoveJointsVel $\left(\dot{\theta}_1, \dot{\theta}_2, \cdots, \dot{\theta}_6\right)$ as our main command. Thanks to the property of Jacobian matrix shown in Equation 3.13, we can transform $\dot{\boldsymbol{x}}$ into $\dot{\boldsymbol{q}}$. Then we multiply it an integrator $\frac{1}{\mathrm{S}}$ to obtain $\boldsymbol{q}$. Note that, As F/T sensor is mounted on the frame$\{6\}$, we should use Equation 3.13 rather than Equation 3.6.

Implementation In Discrete Time

To implement it in practice, we use the bilinear transformation, also known as Tustin method, to control it in discrete time.

$$\begin{aligned} \frac{\dot{\boldsymbol{x}}(\boldsymbol{s})}{\dot{\boldsymbol{f}}(\boldsymbol{s})} &= \frac{1}{\mathrm{MS}+\mathrm{B}} \\ \Rightarrow \frac{\dot{\boldsymbol{x}}(\boldsymbol{z})}{\dot{\boldsymbol{f}}(\boldsymbol{z})} &= \frac{\mathrm{T}(1+z^{-1})}{(\mathrm{BT}+2\mathrm{M})+(\mathrm{BT}-2\mathrm{M})z^{-1}} \end{aligned} \tag{4.24}$$

and,

$$\begin{aligned} \frac{\boldsymbol{q}(\boldsymbol{s})}{\dot{\boldsymbol{q}}(\boldsymbol{s})} &= \frac{\mathrm{K}}{\mathrm{S}} \\ \Rightarrow \frac{\boldsymbol{q}(\boldsymbol{z})}{\dot{\boldsymbol{q}}(\boldsymbol{z})} &= \frac{\mathrm{TK}}{2}\frac{1+z^{-1}}{1-z^{-1}} \end{aligned} \tag{4.25}$$

Therefore,

$$\begin{aligned} &\dot{\boldsymbol{x}}[\boldsymbol{k}+1] = \frac{\mathrm{T}}{\mathrm{BT}+2\mathrm{M}} \cdot (\boldsymbol{f}[\boldsymbol{k}+1]+\boldsymbol{f}[\boldsymbol{k}]) - \frac{\mathrm{BT}-2\mathrm{M}}{\mathrm{BT}+2\mathrm{M}} \cdot \dot{\boldsymbol{x}}[\boldsymbol{k}] \\ &\boldsymbol{q}[\boldsymbol{k}+1] = \boldsymbol{q}[\boldsymbol{k}] + \frac{\mathrm{KT}}{2}(\dot{\boldsymbol{q}}[\boldsymbol{k}+1]+\dot{\boldsymbol{q}}[\boldsymbol{k}]) \end{aligned} \tag{4.26}$$

where T is the sampling period.

4.4 Self-Alignment Mode

The purpose of this section is developing a framework for robot self-alignment regarding the position and orientation of root canal, which is one of the main contributions of the thesis. This function assists endodontists to automatically operate the cleaning procedure and ensures a postoperative recovery for patients. The main idea is to amend the direction of insertion by itself to lower the contact resistance.

We propose a complete robotic procedure. First and foremost, the main role of this procedure is the root canal reamer. We rotate the root canal reamer to clean a pulp, move it to do reciprocation and self-alignment. All of things we care about are all on the root canal such as it position, orientation , rotation speed and contact force. Nevertheless, the robot arm and the F/T sensor have their own coordinates initially. They originally recognize its own coordinate rather than the root canal reamer's frame, so they do their work themselves respectively. Therefore, we need to let them understand the tool frame. Thanks to section 3.4, we have demonstrated how to change the reference frame of the robot arm. After that, we are capable of identifying the translation and rotation information of the root canal reamer and subsequently obtaining the transformation matrix from the robot arm to the tool. As for F/T sensor, we have explained how to do gravity compensation in section 4.2, we will interpret how to change its fiducial to the tool tip in section 4.4.1. Besides, we will advance a motion planning in accordance with dentist's motion and standard surgical solution in section 4.4.2

4.4.1 Coordinate Transformation of F/T sensor

Similar to Coordinate transformation of F/T sensor in section 3.4. The F/T sensor originally recognizes its coordinate rather than the root canal reamer's frame. Namely, the F/T sensor receives unmatched data when the tool contacts an obstacle due to the wrong reference frame. Hence, here we illustrate how to obtain the coordinate transformation of the F/T sensor to decouple the force and toque from frame$\{\mathrm{S}\}$ to the frame$\{\mathrm{T}\}$. We analyze it from two perspectives - reference frame changing and measurement point changing.

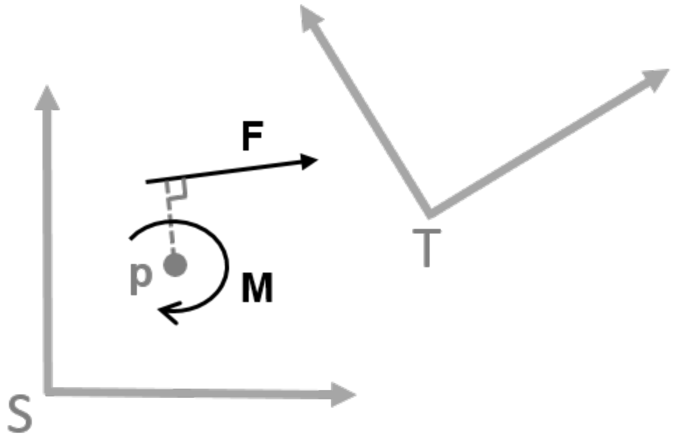

Figure 4.3: Illustration of changing reference frame from $\{\mathrm{S}\}$ to $\{\mathrm{T}\}$. f and m respectively denote a force and a moment measured at point p.

In view of reference frame changing in Figure 4.3, we can derive the following equation.

$$\begin{bmatrix} \boldsymbol{f}_\mathrm{p} \\ \boldsymbol{m}_\mathrm{p} \end{bmatrix}_{\{\mathrm{T}\}} = \begin{bmatrix} {}^\mathrm{T}_\mathrm{S}\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}^\mathrm{T}_\mathrm{S}\mathbf{R} \end{bmatrix} \begin{bmatrix} \boldsymbol{f}_\mathrm{p} \\ \boldsymbol{m}_\mathrm{p} \end{bmatrix}_{\{\mathrm{S}\}} \tag{4.27}$$

On the other hand, in Figure 4.4 we describe a force and a torque observed in the same frame but different point.

$$\begin{aligned} \boldsymbol{f} &= \boldsymbol{f}_\mathrm{p} = \boldsymbol{f}_\mathrm{q} \\ \boldsymbol{m}_\mathrm{q} &= \boldsymbol{r}_\mathrm{q} \cdot \boldsymbol{f} \\ &= \boldsymbol{r}_\mathrm{p} \cdot \boldsymbol{f} + \overrightarrow{\mathrm{qp}} \times \boldsymbol{f}_\mathrm{p} \\ &= \boldsymbol{m}_\mathrm{p} + \overrightarrow{\mathrm{qp}} \times \boldsymbol{f}_\mathrm{p} \end{aligned} \tag{4.28}$$

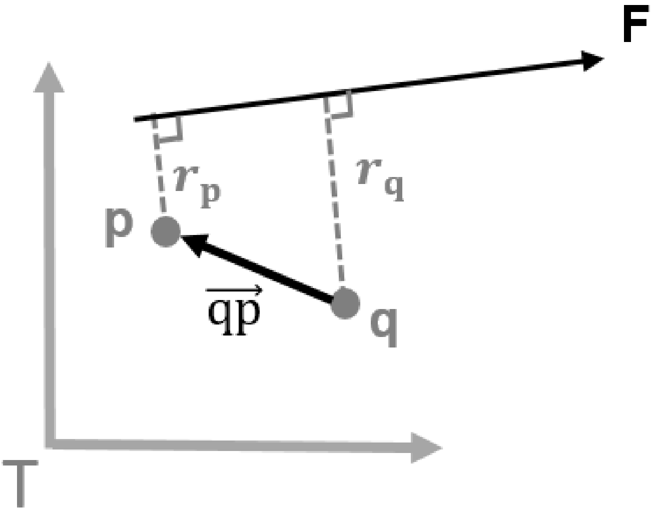

Figure 4.4: Illustration of changing reference frame from $\{\mathrm{S}\}$ to $\{\mathrm{T}\}$. $\boldsymbol{f}$ and $\boldsymbol{m}$ respectively denote a force and a moment measured at point p.

Assume

$$\overrightarrow{\mathrm{qp}} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

then

$$\overrightarrow{\mathrm{qp}} \times \boldsymbol{f}_{\mathrm{p}} = \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \tag{4.29}$$

As s result, we consider reference frame changing and measurement point changing, then we can obtain the following essential equation.

$$\begin{bmatrix} \boldsymbol{f}_{\mathrm{q}} \\ \boldsymbol{m}_{\mathrm{q}} \end{bmatrix}_{\{\mathrm{T}\}} = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0} \\ \begin{matrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{matrix} & \mathbf{I}_{3\times3} \end{bmatrix} \begin{bmatrix} {}^{\mathrm{T}}_{\mathrm{S}}\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}^{\mathrm{T}}_{\mathrm{S}}\mathbf{R} \end{bmatrix} \begin{bmatrix} \boldsymbol{f}_{\mathrm{p}} \\ \boldsymbol{m}_{\mathrm{p}} \end{bmatrix}_{\{\mathrm{S}\}} \tag{4.30}$$

4.4.2 Motion Planning: Based on Admittance Control

Finally, we plan a series of motions to simulate the cleaning procedure of the root canal treatment. Endodontist uses their professional experience to determine how many force to apply and when to reverse the file to release the torque caused by the contact force. Because the root canal resembles a cone, the profundity of the file is larger, the resultant force composed of contact forces is more significant. Namely, the endodontist will apply more and more pressure in the insertion direction during the surgery. To simulate this series of motions, we could directly command the robot arm to move with the series of actions. However, commanding the robot arm is a dilemma because we have already use admittance control which also commands the robot arm to move. Therefore, to solve the conflict, we continue to utilize admittance control and propose a method based on admittance control to simulate the series of motions shown in Fig 4.5.

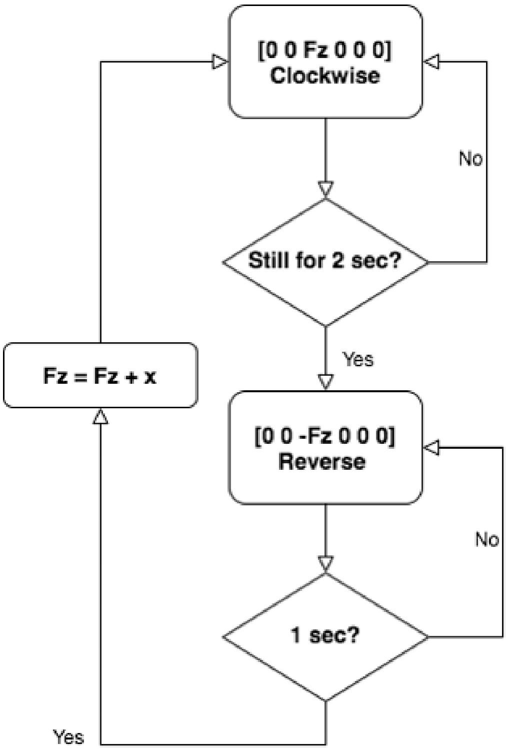

Figure 4.5: Flow chart of motion planning

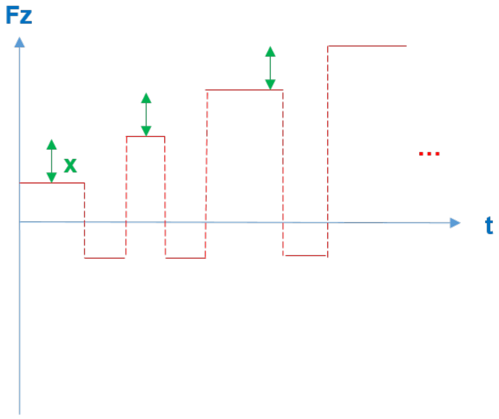

Figure 4.6: Flow chart of motion planning

4.5 Affections of Parameters Setting

Because our system is similar to a mass- damper system, we focus on the performance of the velocity $\dot{x}$ whereby the system could be considered as

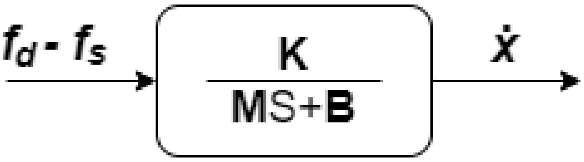

Figure 4.7: Control scheme. $\mathbf{M}, \mathbf{B}, \mathbf{K}$ are all diagonal positive definite matrices, whose diagonal indexes are respectively related to $x, y, z, \theta_x, \theta_y, \theta_z$. $\mathbf{M}, \mathbf{B}$ are related to the inertial and damping respectively, and $\mathbf{K}$ is a proportional gain.

It is a first-order control system and its step response is

$$\begin{aligned}\mathcal{L}^{-1}\left[\frac{1}{\mathrm{S}} \cdot \frac{\mathbf{K}}{\mathrm{M}\mathrm{S}+\mathbf{B}}\right] &= \mathcal{L}^{-1}\left[\frac{\mathbf{K}}{\mathbf{B}}\left(\frac{1}{\mathrm{S}} - \frac{1}{\mathrm{S}+\frac{\mathbf{B}}{\mathrm{M}}}\right)\right] \\ &= \frac{\mathbf{K}}{\mathbf{B}}\left(1 - e^{-\frac{\mathbf{B}}{\mathrm{M}}t}\right)\end{aligned} \tag{4.31}$$

From the above derivation, we can know

$$\begin{aligned}\text{time constant } \boldsymbol{\tau} &= \frac{\mathbf{M}}{\mathbf{B}} \\ \text{gain} &= \frac{\mathbf{K}}{\mathbf{B}}\end{aligned} \tag{4.32}$$

Hence,the transient response can be derived as

$$\begin{aligned}\text{Rise time} = 2.3\boldsymbol{\tau} &= \frac{2.3\mathbf{M}}{\mathbf{B}} \\ \text{Settling time} = 4\boldsymbol{\tau} &= \frac{4\mathbf{M}}{\mathbf{B}}\end{aligned} \tag{4.33}$$

In theory,when $\boldsymbol{\tau}$ is larger, the pole is farther the from origin in S domain, the system is more stable. Besides, when $\boldsymbol{\tau}$ is larger, the response speed is faster. Furthermore, when gain $\frac{\mathbf{K}}{\mathbf{B}}$ is larger, the robot arm move more. Therefore, coarse-tune parameter $\mathbf{K}$ to adjust the whole gain of the system and determine the mode of the system is ”Dragging Mode” or ”Self-Alignment Mode”. Last but not least, we fine-tune diagonal parameters of $\mathbf{B}$ separately because the inertial and spring properties of each axis are discrepant.

In practice, we set $(\mathbf{M} : \mathbf{B} = 1 : 1000)$ to make $(\boldsymbol{\tau} = \frac{\mathbf{M}}{\mathbf{B}} = 0.001)$, then the system rises up to 98% after 0.003 second. Further, we can utilize $\mathbf{K}$ to easily . In Table 4.3, we show detailed parameters with ”Dragging Mode” and ”Self-Alignment Mode”.

In ”Dragging Mode”, we hold the handpiece to move such as shown in Fig. Due to different level arm, the forces we apply in each axis are all different. Therefore, in order to provide users a good experience, we set each parameter of $mathbfB$ of ”Dragging mode” different value. With these different values, the users can comfortably move the robot arm to the desired position and orientation. On the other hand, in ”Self-Alignment Mode”, because external force is applied at the tool and we have already transform the reference frame of F/T sensor, we set the the first three variables which is relative to the linear velocity as the same values and set the last three variables which is relative to the angular velocity as the other same values.

Table 4.3: Parameters setting of Admittance control.

Parameter	Dragging Mode	Self-Alignment Mode
k	1200	350
b_1	1.2	0.7
b_2	0.6	0.7
b_3	1.1	0.7
b_4	1.7	2
b_5	4	2
b_6	1.8	2
	$\mathbf{K} = \text{diag}(k, k, k, k, k, k),$	$\mathbf{B} = \text{diag}(b_1, b_2, b_3, b_4, b_5, b_6)$

Chapter 5

Control of Endodontic File Rotation

5.1 Problem Definition

(Main cause of Files Fracture instrument fracture)

(File property)

5.2 The Proposed Method and Theorem

(CACS2020 Prototype 1)

(Motion Planning: sections)(Current threshold setting)

Once the current of the file is in excess of the threshold, it will inversely rotate to release torque. A decline(decrease/increase/drop) in the current of the motor is indicative of contact resistance of the file.

Chapter 6

Preliminary Experiment Result

6.1 Experimental Setup

Use present tense.

(Communication protocol – EtherCAT, RTOS – NI target)

For 6.2 experiment: (Stewart-Platform + PhaseSpace + markers)

For 6.3 6.4 experiments: (Acrylic root canal model + truth tooth)

6.2 Admittance Control

In order to prove the validation of admittance control, we set up this experiment. Here we built a Stewart platform, which had six degree of freedom and provided a small movement. We used Stewart platform to simulate a motion of patient. Basically, when the patient move to a position, our system should move to the same position. Therefore, we planned to observe the target's and the handpiece's position to validate that our system can track the patient. Besides, we use PhaseSpace to obtain their motion in real time. PhaseSpace is a motion capture device whose resolution is around 1 mm. Before starting this experiment, we should fix a relative position between the file and the acrylic tooth. We made the file rotate and get stuck in the root canal of acrylic model. Then we used "Doctor dragging" mode to install the acrylic model on the Stewart platform. Therefore, we can guarantee that the relative position between the file and the acrylic tooth.

We moved the Stewart platform in horizontal and vertical direction separately. The motion planning in horizontal direction is a square which is a 10×10 mm and in vertical direction is a linear motion from 0 to 40 mm.

6.3 Automatically Direction Changing

Validation of Self-alignment Mode
(Metrics: time, completeness and file breakage)
(Completeness definition: comparison of pixel area before and after experiment via image)

6.4 Repetitive Experiment

validation of repetitive experiment
(Metrics: file breakage, compare with and without reverse)

Chapter 7

Conclusions and Future works

(Patient move tracking via cable, root canals searching)

Chapter 8

Appendix

8.1 Forward Kinematics

$$
{}^{0}_{6}\mathbf{T} ={}^{0}_{1}\mathbf{T}\cdot{}^{1}_{2}\mathbf{T}\cdot{}^{2}_{3}\mathbf{T}\cdot{}^{3}_{4}\mathbf{T}\cdot{}^{4}_{5}\mathbf{T}\cdot{}^{5}_{6}\mathbf{T} = \begin{bmatrix} {}^{0}_{6}\mathbf{R}_{3\times 3} & {}^{0}\boldsymbol{p}_{6_{\mathrm{org}}} \\ 0_{1\times 3} & 1 \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} & t_{13} & t_{14} \\ t_{21} & t_{22} & t_{23} & t_{24} \\ t_{31} & t_{32} & t_{33} & t_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
\begin{aligned}
t_{11} = & -S_6(C_4S_1 + S_4(C_1S_3 - C_2 - C_1C_3S_2)) - C_6(C_5(S_1S_4 - C_4(C_1S_3 \\
& - C_2 - C_1C_3S_2)) - S_5(C_1C_3 - C_2 + C_1S_2S_3)) \\
t_{12} = & \, S_6(C_5(S_1S_4 - C_4(C_1S_3 - C_2 - C_1C_3S_2)) - S_5(C_1C_3 - C_2 + C_1S_2S_3)) \\
& - C_6(C_4S_1 + S_4(C_1S_3 - C_2 - C_1C_3S_2)) \\
t_{13} = & -S_5(S_1S_4 - C_4(C_1S_3 - C_2 - C_1C_3S_2)) - C_5(C_1C_3 - C_2 + C_1S_2S_3) \\
t_{14} = & \, 135C_1S_2 - 70S_5(S_1S_4 - C_4(C_1S_3 - C_2 - C_1C_3S_2)) - 70C_5(C_1C_3 - C_2 \\
& + C_1S_2S_3) - 120C_1C_3 - C_2 - 120C_1S_2S_3 - 38C_1S_3 - C_2 + 38C_1C_3S_2 \\
t_{21} = & \, S_6(C_1C_4 + S_4(C_3S_2S_1 - S_1S_3 - C_2)) + C_6(C_5(C_1S_4 - C_4(C_3S_2S_1 \\
& - S_1S_3 - C_2)) + S_5(C_3S_1 - C_2 + S_2S_1S_3)) \\
t_{22} = & \, C_6(C_1C_4 + S_4(C_3S_2S_1 - S_1S_3 - C_2)) - S_6(C_5(C_1S_4 - C_4(C_3S_2S_1 \\
& - S_1S_3 - C_2)) + S_5(C_3S_1 - C_2 + S_2S_1S_3))
\end{aligned}
$$

$$t_{23} = S_5(C_1S_4 - C_4(C_3S_2S_1 - S_1S_3 - C_2)) - C_5(C_3S_1 - C_2 + S_2S_1S_3)$$

$$\begin{aligned} t_{24} = {} & 135S_2S_1 + 70S_5(C_1S_4 - C_4(C_3S_2S_1 - S_1S_3 - C_2)) - 70C_5(C_3S_1 - C_2 \\ & + S_2S_1S_3) + 38C_3S_2S_1 - 120C_3S_1 - C_2 - 120S_2S_1S_3 - 38S_1S_3 - C_2 \end{aligned}$$

$$\begin{aligned} t_{31} = {} & C_6(S_5(C_3S_2 - S_3 - C_2) + C_4C_5(C_3 - C_2 + S_2S_3)) - S_4S_6(C_3 - C_2 \\ & + S_2S_3) \end{aligned}$$

$$\begin{aligned} t_{32} = {} & -S_6(S_5(C_3S_2 - S_3 - C_2) + C_4C_5(C_3 - C_2 + S_2S_3)) - C_6S_4(C_3 - C_2 \\ & + S_2S_3) \end{aligned}$$

$$t_{33} = C_4S_5(C_3 - C_2 + S_2S_3) - C_5(C_3S_2 - S_3 - C_2)$$

$$\begin{aligned} t_{34} = {} & 120S_3 - C_2 - 120C_3S_2 - 38C_3 - C_2 - 38S_2S_3 - 135 - C_2 \\ & - 70C_5(C_3S_2 - S_3 - C_2) + 70C_4S_5(C_3 - C_2 + S_2S_3) + 135 \end{aligned}$$

8.2 Jacobian matrix

8.2.1 Jg0

$$\begin{aligned} j_{g0,21} = {} & 135C_1S_2 - 120C_1S_2S_3 - 70S_1S_4S_5 + 120C_1C_2C_3 + 38C_1C_2S_3 \\ & + 38C_1C_3S_2 + 70C_1C_2C_3C_5 - 70C_1C_5S_2S_3 - 70C_1C_2C_4S_3S_5 \\ & - 70C_1C_3C_4S_2S_5 \end{aligned}$$

$$\begin{aligned} j_{g0,22} = {} & -S_1(120C_2S_3 - 38C_2C_3 - 135C_2 + 120C_3S_2 + 38S_2S_3 + 70C_2C_5S_3 \\ & + 70C_3C_5S_2 + 70C_2C_3C_4S_5 - 70C_4S_2S_3S_5) \end{aligned}$$

$$\begin{aligned} j_{g0,23} = {} & -2S_1(60C_2S_3 - 19C_2C_3 + 60C_3S_2 + 19S_2S_3 + 35C_2C_5S_3 \\ & + 35C_3C_5S_2 + 35C_2C_3C_4S_5 - 35C_4S_2S_3S_5) \end{aligned}$$

$$j_{g0,24} = 70S_5(C_1C_4 + C_2S_1S_3S_4 + C_3S_1S_2S_4)$$

$$j_{g0,25} = -70C_5(C_2C_4S_1S_3 - C_1S_4 + C_3C_4S_1S_2) - 70C_{34}S_1S_5$$

$$j_{g0,26} = 0$$

$$j_{g0,31} = 0$$

$$\begin{aligned} j_{g0,32} = {} & 120S_2S_3 - 120C_2C_3 - 38C_2S_3 - 38C_3S_2 - 135S_2 + 70C_5S_2S_3 \\ & - 70C_2C_3C_5 + 70C_2C_4S_3S_5 + 70C_3C_4S_2S_5 \end{aligned}$$

$$\begin{aligned}
j_{g0,33} &= 120S_2S_3 - 38C_2S_3 - 38C_3S_2 - 120C_2C_3 + 70C_5S_2S_3 - 70C_2C_3C_5 \\
&\quad + 70C_2C_4S_3S_5 + 70C_3C_4S_2S_5 \\
j_{g0,34} &= 70C_{34}S_4S_5 \\
j_{g0,35} &= 70S_{34}S_5 - 70C_{34}C_4C_5 \\
j_{g0,36} &= 0 \\
j_{g0,41} &= \theta_4S_1S_2S_3 - \theta_3C_1 - \theta_5C_1C_4 - \theta_4C_2C_3S_1 - \theta_6C_1S_4S_5 - \theta_2C_1 \\
&\quad - \theta_6C_2C_3C_5S_1 - \theta_5C_2S_1S_3S_4 - \theta_5C_3S_1S_2S_4 + \theta_6C_5S_1S_2S_3 \\
&\quad + \theta_6C_2C_4S_1S_3S_5 + \theta_6C_3C_4S_1S_2S_5 \\
j_{g0,42} &= \theta_5C_1C_2C_3S_4 - \theta_4C_1C_2S_3 - \theta_4C_1C_3S_2 - S_1 - \theta_6C_1C_2C_5S_3 \\
&\quad - \theta_6C_1C_3C_5S_2 - \theta_5C_1S_2S_3S_4 - \theta_6C_1C_2C_3C_4S_5 + \theta_6C_1C_4S_2S_3S_5 \\
j_{g0,43} &= \theta_5C_1C_2C_3S_4 - \theta_4C_1C_2S_3 - \theta_4C_1C_3S_2 - S_1 - \theta_6C_1C_2C_5S_3 \\
&\quad - \theta_6C_1C_3C_5S_2 - \theta_5C_1S_2S_3S_4 - \theta_6C_1C_2C_3C_4S_5 + \theta_6C_1C_4S_2S_3S_5 \\
j_{g0,44} &= \theta_5(S_1S_4 + C_1C_2C_4S_3 + C_1C_3C_4S_2) - C_1S_2S_3 + C_1C_2C_3 \\
&\quad + \theta_6S_5(C_1C_2S_3S_4 - C_4S_1 + C_1C_3S_2S_4) \\
j_{g0,45} &= \; - C_4S_1 - \theta_6(C_5(S_1S_4 - C_4(C_1S_3 - C_2 - C_1C_3S_2)) - S_5(C_1C_3 \\
&\quad - C_2 + C_1S_2S_3)) - S_4(C_1S_3 - C_2 - C_1C_3S_2) \\
j_{g0,46} &= C_{34}C_1C_5 - S_5(S_1S_4 + C_1C_2C_4S_3 + C_1C_3C_4S_2) \\
j_{g0,51} &= \theta_4C_1C_2C_3 - \theta_3S_1 - \theta_5C_4S_1 - theta2S_1 - \theta_4C_1S_2S_3 - \theta_6S_1S_4S_5 \\
&\quad + \theta_6C_1C_2C_3C_5 + \theta_5C_1C_2S_3S_4 + \theta_5C_1C_3S_2S_4 - \theta_6C_1C_5S_2S_3 \\
&\quad - \theta_6C_1C_2C_4S_3S_5 - \theta_6C_1C_3C_4S_2S_5 \\
j_{g0,52} &= C_1 - \theta_4C_2S_1S_3 - \theta_4C_3S_1S_2 + \theta_5C_2C_3S_1S_4 - \theta_6C_2C_5S_1S_3 \\
&\quad - \theta_6C_3C_5S_1S_2 - \theta_5S_1S_2S_3S_4 - \theta_6C_2C_3C_4S_1S_5 + \theta_6C_4S_1S_2S_3S_5 \\
j_{g0,53} &= C_1 - \theta_4C_2S_1S_3 - \theta_4C_3S_1S_2 + \theta_5C_2C_3S_1S_4 - \theta_6C_2C_5S_1S_3 \\
&\quad - \theta_6C_3C_5S_1S_2 - \theta_5S_1S_2S_3S_4 - \theta_6C_2C_3C_4S_1S_5 + \theta_6C_4S_1S_2S_3S_5 \\
j_{g0,54} &= \theta_5(C_2C_4S_1S_3 - C_1S_4 + C_3C_4S_1S_2) + \theta_6S_5(C_1C_4 + C_2S_1S_3S_4 \\
&\quad + C_3S_1S_2S_4) - S_1S_2S_3 + C_2C_3S_1
\end{aligned}$$

$$\begin{aligned}
j_{g0,55} &= C_1C_4 + S_{34}S_1S_4 - \theta_6C_{34}S_1S_5 + \theta_6C_1C_5S_4 - \theta_6C_2C_4C_5S_1S_3 \\
&\quad - \theta_6C_3C_4C_5S_1S_2 \\
j_{g0,56} &= C_{34}C_5S_1 - S_5(C_2C_4S_1S_3 - C_1S_4 + C_3C_4S_1S_2) \\
j_{g0,61} &= 1 \\
j_{g0,62} &= \theta_6S_{34}C_4S_5 - \theta_6C_{34}C_5 - \theta_5S_{34}S_4 - \theta_4C_{34} \\
j_{g0,63} &= \theta_6S_{34}C_4S_5 - \theta_6C_{34}C_5 - \theta_5S_{34}S_4 - \theta_4C_{34} \\
j_{g0,64} &= \theta_5C_2C_3C_4 - C_3S_2 - C_2S_3 - \theta_5C_4S_2S_3 + \theta_6C_2C_3S_4S_5 - \theta_6S_2S_3S_4S_5 \\
j_{g0,65} &= \theta_6(S_{34}S_5 - C_{34}C_4C_5) + C_{34}S_4 \\
j_{g0,66} &= -S_{34}C_5 - C_{34}C_4S_5
\end{aligned}$$

8.2.2 Jg6

$$\begin{aligned}
j_{g6,11} &= 135C_4S_2S_6 + 120C_2C_3C_4S_6 + 70C_2C_3C_6S_4 + 38C_2C_4S_3S_6 \\
&\quad + 38C_3C_4S_2S_6 + 135C_5C_6S_2S_4 - 120C_4S_2S_3S_6 - 70C_6S_2S_3S_4 \\
&\quad - 70C_2S_3S_5S_6 - 70C_3S_2S_5S_6 + 70C_2C_3C_4C_5S_6 + 120C_2C_3C_5C_6S_4 \\
&\quad + 38C_2C_5C_6S_3S_4 + 38C_3C_5C_6S_2S_4 - 70C_4C_5S_2S_3S_6 \\
&\quad - 120C_5C_6S_2S_3S_4 \\
j_{g6,12} &= 70C_4C_6 - 38C_6S_5 - 120S_4S_6 - 70C_5S_4S_6 + 135S_3S_4S_6 + 120C_4C_5C_6 \\
&\quad - 135C_3C_6S_5 - 135C_4C_5C_6S_3 \\
j_{g6,13} &= 70C_4C_6 - 38C_6S_5 - 120S_4S_6 - 70C_5S_4S_6 + 120C_4C_5C_6 \\
j_{g6,14} &= 70S_5S_6 \\
j_{g6,15} &= 70C_6 \\
j_{g6,16} &= 0 \\
j_{g6,21} &= 135C_4C_6S_2 + 120C_2C_3C_4C_6 + 38C_2C_4C_6S_3 + 38C_3C_4C_6S_2 \\
&\quad - 70C_2C_3S_4S_6 - 120C_4C_6S_2S_3 - 70C_2C_6S_3S_5 - 70C_3C_6S_2S_5 \\
&\quad - 135C_5S_2S_4S_6 + 70S_2S_3S_4S_6 + 70C_2C_3C_4C_5C_6 - 120C_2C_3C_5S_4S_6 \\
&\quad - 70C_4C_5C_6S_2S_3 - 38C_2C_5S_3S_4S_6 - 38C_3C_5S_2S_4S_6
\end{aligned}$$

$$
\begin{aligned}
&+ 120C_5S_2S_3S_4S_6 \\
j_{g6,22} &= 38S_5S_6 - 120C_6S_4 - 70C_4S_6 + 135C_6S_3S_4 + 135C_3S_5S_6 \\
&- 120C_4C_5S_6 - 70C_5C_6S_4 + 135C_4C_5S_3S_6 \\
j_{g6,23} &= 38S_5S_6 - 120C_6S_4 - 70C_4S_6 - 120C_4C_5S_6 - 70C_5C_6S_4 \\
j_{g6,24} &= 70C_6S_5 \\
j_{g6,25} &= -70S_6 \\
j_{g6,26} &= 0
\end{aligned}
$$

Reference

[1] M.-K. Wu, D. Barkis, A. Roris, and P. Wesselink, “Does the first file to bind correspond to the diameter of the canal in the apical region?” *International Endodontic Journal*, vol. 35, no. 3, pp. 264–267, 2002. 7

[2] J. Agrawal, P. K. Shenai, L. Chatra, P. Y. Kumar *et al.*, “Evaluation of normal range of mouth opening using three finger index: South india perspective study,” *Indian Journal of Dental Research*, vol. 26, no. 4, p. 361, 2015. 7

[3] C. Yang, J. Wang, L. Mi, X. Liu, Y. Xia, Y. Li, S. Ma, and Q. Teng, “A four-point measurement model for evaluating the pose of industrial robot and its influence factor analysis,” *Industrial Robot: An International Journal*, 2017. 16

[4] T. Tang, H. Lin, Yu Zhao, Wenjie Chen, and M. Tomizuka, “Autonomous alignment of peg and hole by force/torque measurement for robotic assembly,” in *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, 2016, pp. 162–167. 22

[5] P. Zou, Q. Zhu, J. Wu, and J. Jin, “An approach for peg-in-hole assembling based on force feedback control,” in *2019 Chinese Automation Congress (CAC)*, 2019, pp. 3269–3273. 22