Final Academic Internship Project (PFE)

Robotics & Embedded Systems
2020 - 2021

Development of a task-centered grasp analysis

Technical specifications for in-hand and dexterous manipulation

---

MIDTERM REPORT

Author:

Ricardo RICO URIBE

Promotion:

2021

Host Organism Tutor:

Mathieu GROSSARD

ENSTA Paris Tutor:

Thibault TORALBA

April 12 2021 - October 22 2021

CEA List

2 Boulevard Gobert
91120 Palaiseau - France

Contents

Context & Introduction

1.1 Context

When the pandemic started everyone was scrambling to find a solution, the European Union was not an exception, in the search of some medical advancement to develop a vaccine they found themselves waiting because some steps of the research process took too much time, some waiting period is inevitable but when the slowest part of a process is a human doing a "basic" task, we know that there is room for improvement. That is what inspired the creation of the project TRACEBOT[1]. A static robot with two manipulators, capable of working in an sterile environment, take samples and use medical equipment like needles, pumps and scissors, all while being traceable to ensure the highest standards of quality.

The project has a enormous team working on different parts of the subject, and I was selected to help one of the project colaborators, the Atomic Energy Commissary (CEA). I am part of a sub team dedicated to the design, construction and control of the robotic manipulators, the hands of the robot that will manipulate all the objects.

1.2 Introduction

This report presents my support to the development of the manipulators at the CEA and the project overall.

My project focus on the design of the manipulators via mechanical analysis and optimization to develop a platform that delivers the technical specifications of a manipulator that is task-oriented. Task-oriented means that we are not using a pre-existing robot, nor focusing on the geometry of the objects (object-oriented), we are focusing on forces that affect the objects, and that relay directly on the selection of the motors used on the manipulator.

You will find and extensive theoretical base for robot manipulators and the mathematical description needed to analyse and object, a task and a manipulator. With this knowledge it will be easier to understand our findings on which manipulator design is best suited for each task, and a conclusive manipulator to encompass all task and objects.

The project being a software development, a basic explanation of the code is included but if needed more information and the code itself can be found in the repository hosted at the lab.

2 Grasping

In this section we will present our main mathematical tool, the grasp matrix G. Because of the theoretical nature of this section and my inability to explain better than a published book, the information presented came from *Springer Handbook of Robotics*[3], mainly the chapter of *Grasping*. This section summarizes and puts in order the knowledge needed to understand the concept of Grasping, feel free to expand on the information by using the chapter directly. The notation used in this section and the project in general is the one used in the chapter.

2.1 Definition

In short, the concept of grasping refers to the action of grasp (take hold) of an object, and move it, being moved by the effect of moving all the manipulator or being moved by the changes of position of each joint; the latter being called dexterous manipulation. Grasping is represented trough the grasp matrix G.

What is G?

Assume we have a rigid body *(object)* in a world with a frame $\{N\}$ described by $\{Ow, X, Y, Z\}$, the object has its own frame $\{B\}$: $\{p, x, y, z\}$ and the object is being grasp at various points throughout its surface, at each contact point $\{Ci\}_{i:1...nc}$ there is a tangent plane, that allows us to describe each contact point with its own frame, the normal vector to the tangent plane, points to the interior of the object and it is the X equivalent of each C_i, which leaves us with $\{c_i, n_i, o_i, t_i\}$ as the frame of each contact point. (The contact points are described in $\{N\}$).

At last, consider g as the sum of all external efforts applied to the object and v the velocity of the object at point p both described in $\{N\}$. Figure 1 shows this configuration.

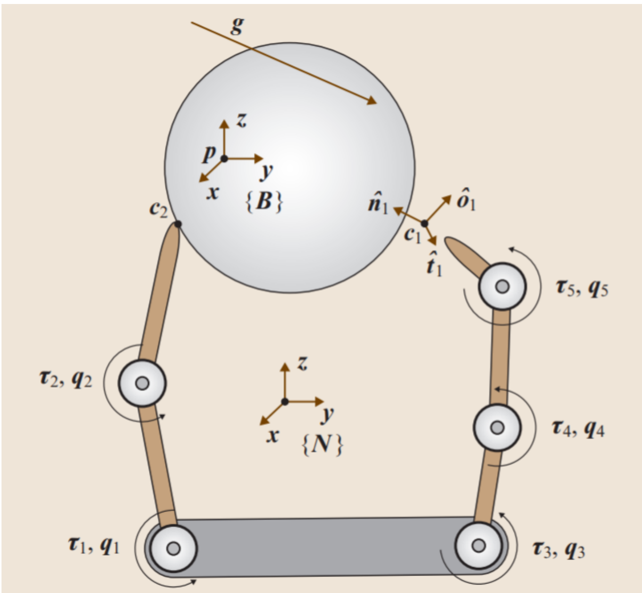

Figure 1: Main Notions for Grasp Analysis

In these conditions, the *complete* grasp matrix $\widetilde{G}$ links the velocities of each contact point expressed in its own frame to the velocity of the object expressed in the world frame, in the following way:

$$v_{contact} = \widetilde{G}^T v \tag{1}$$

Where $v_{contact} = [v_{contact_{1,...,contact_{nc}}}]$ is the concatenation of all contact points velocity vectors, expressed in the corresponding contact frame.

$\widetilde{G}^T$ is obtained by the following calculations:

$$\widetilde{G}^T = \begin{pmatrix} \widetilde{G}_1^T \\ \vdots \\ \widetilde{G}_{nc}^T \end{pmatrix} \tag{2}$$

$$\widetilde{G}_i^T = \overline{R}_i^T P_i^T \tag{3}$$

Where: $\overline{R}_i = Blockdiag(R_i, R_i) = \left(\begin{smallmatrix} R_i & 0 \\ 0 & R_i \end{smallmatrix}\right)$, $P_i = \left(\begin{smallmatrix} I_{3x3} & 0 \\ S(ci-p) & I_{3x3} \end{smallmatrix}\right)$, $S(r) = \left(\begin{smallmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{smallmatrix}\right)$, and, R_i is the rotation of the contact frame $\{C_i\}$ in respect to $\{N\}$.

Previously we indicated that $\widetilde{G}$ is the *complete* grasp matrix, this is because $\widetilde{G}$ maps all the 6 contact velocities (3 linear and 3 angular) of each C_i to the object velocity, in practice the transmitted velocities heavily rely on the contact type, topic that will be addressed further down. For now lets indicate that:

$$G^T = H\widetilde{G}^T \tag{4}$$

where H is the component selection matrix, obtained with: $H = Blockdiag(H_1, \ldots, H_{nc})$ and $Hi = \left(\begin{smallmatrix} H_{iF} & 0 \\ 0 & H_{iM} \end{smallmatrix}\right)$

2.2 Contact Types

There are three main contact types in grasping, Point with-out Friction (PwoF),Hard Finger (HF),Soft Finger (SF). Each one more complex than the previous but with more components transmitted.

Point with-out Friction

This contact type its based on the assumption that the contact is produced in a infinitely small single point so there is no friction between the surface of the object and the finger. Where only the normal component (aligned with $\{n_i\}$) is transmitted.

Hard Finger

The contact allows for tangential friction μ,so the three linear components are passed $\{n_i, o_i, t_i\}$

Soft Finger

This contact is the same as Hard Finger with the addition of torsional friction γ allowing for the transmission of momentum along the contact normal.

We use the following table to reference and construct H.

Model	li	H_{iF}	H_{iM}
PwoF	1	(1 0 0)	none
HF	3	I_{3x3}	none
SF	4	I_{3x3}	(1 0 0)

Table 1: Selection Matrices for Contact Models

Now we have a comprehensive grasp matrix $G \in R^{6xl}$,where $l = \sum_{i=1}^{nc} li$. Which allows us to transmit the selected velocities and momentums at each contact point.
But now we have to know if the grasp that our matrix describes is correct, simply putted, if it accomplished our desired goal. That is why we need to classify the grasps and rank them, topics discussed in the following sections.

2.3 Classification

The chapter of grasping presents 2 different classifications for a grasp system that depend only on G.

Indeterminate

A grasping system is indeterminate if $N(G^T)$ is nonTrivial.
If a system is indeterminate it means that for an object with 6DoF there are some velocities and momentums that cannot be imprinted by the manipulator (there are internal object *twists* that cannot be controlled).

Graspable

A grasping system is graspable if $N(G)$ is nonTrivial.
All internal object *forces* are controlled, this means that we can augment the contact forces and the object will not move but instead improve the grasp if it is based on friction.

2.4 Quality Metrics

We need to determine how good is G, that is why we have a quality metric that test for Frictional Form Closure (FFC) this metric is change in the basic Form Closure. Form Closure can exist with any contact type, but in its most basic form, it uses PwoF and uses one contact to restrict each DoF of the object plus an additional contact to obtain full form closure. Now as this approach requires a lot of contacts and uses a very primitive contact model, that is why FFC was developed.

Frictional Form Closure

FFC reduces the amount of contacts by using the friction to restrict some of the DoF of the object. But this approach encompasses more challenges, now it is necessary a friction model, the book recommends the Coulomb friction Cone, that needs to be faceted fig.2 to work with a Linear Programming (LP) Problem as shown below.

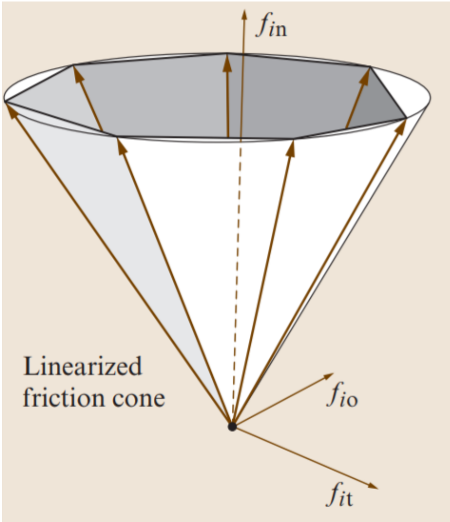

Figure 2: Faceted Friction Cone

$$
\begin{array}{lrl}
\max & d & \\
\text{s.t.} & G\lambda = & 0 \\
& F\lambda - \;\; Id \geq & 0 \\
& d \geq & 0 \\
& e\lambda \leq & nc
\end{array}
\tag{5}
$$

In this LP (5) we are trying to *maximize* d, a distance that can be interpreted as how far is the grasp from losing FFC, and rank the grasps by which one has a bigger d.

To solve 5 we are missing some definitions.

$$F = Blockdiag(F_1, \ldots, F_{nc})_{i:1,\ldots,nc}$$

$$
F_i = \begin{pmatrix} S_i[col_1] \times S_i[col_2] \\ \vdots \\ S_i[col_k] \times S_i[col_{k+1}] \\ \vdots \\ S_i[col_{ng}] \times S_i[col_1] \end{pmatrix}
\qquad
S_i = \begin{pmatrix} \ldots & 1 & \ldots \\ \ldots & \mu_i \cos(2k\pi/ng) & \ldots \\ \ldots & \mu_i \sin(2k\pi/ng) & \ldots \end{pmatrix}
$$

S_i is the friction cone generator matrix for contact point i, F_i is the faceted friction cone and F is the recollection of all friction cones for the nc points.

Lastly, $e = (e_1, \ldots, e_{nc})$, with e_i being the first row of H_i.

The definition of this quality metric was important for the development of a more useful metric, the following metric was proposed by ***Mathieu GROSSARD*** for a previous similar project that was object oriented[2], fortunately, the metric is task oriented because it gives the maximum force or momentum for a given direction $d_{W_{ext}}$.

Task Oriented Quality Metric

As previously stated g is the sum of all external efforts, that means that g can be described as $g = \alpha d_{W_{ext}}$, with this form of g we can analyze any desired direction to obtain the maximum force or moment that the grasp can handle.

The resulting LP problem is the following:

$$
\begin{array}{lrl}
\max & \alpha & \\
\text{s.t.} & Gf_c + \alpha d_{W_{ext}} = & 0 \\
& Ff_c \geq & 0 \\
& \alpha \geq & 0 \\
& f_{c_n} \geq & 0 \\
& f_c \leq & f_{c_{max}}
\end{array}
\tag{6}
$$

The LP can be interpreted as assuring a static equilibrium, ensuring all contact forces are inside their friction cones, alpha to remain positive to maintain the direction, ensuring that the force in the normal direction of any contact point goes towards the object to maintain contact, and taking into account the actuators limitations to be able to correctly size the motors.

This quality metric is our holy grail, with it we can extract real life technical specifications for a determined grasp.

2.5 Jacobian

Throughout this entire section we have neglected the jacobian, if you are familiar with robotics you would know this is a very common concept, and is tightly linked to a robot architecture description. The jacobian matrix links each robot joint velocity to the end-effector velocities.

But in the case of Grasping we have a modified Jacobian, because we are not analysing the end-effector in it-self but rather and object being contacted by the end-effector, and more importantly, we have more than one end-effector, because for the manipulation we have to consider multiple fingers of a single manipulator.

Because of this we have the following mathematical description for the Grasp Jacobian, called inside this report just as jacobian $J \in R^{6 \times nq}$.

$$J = H\widetilde{J}$$

$$\widetilde{J} = \begin{pmatrix} \widetilde{J}_1 \\ \vdots \\ \widetilde{J}_{nc} \end{pmatrix}$$

$$\widetilde{J}_i = \overline{R}_i^T Z_i$$

I will not enter in the mathematical complexity of the Jacobian or the more in-depth definition of each element, the project is currently in an only grasp analysis state. If this changes i will include in the final report the theory missing from the grasp jacobian.

3 Overall Progress

As starters I had to learn all theory pertaining the grasping concept, after this was done I created a code base in python (but using Object Oriented Logic) to calculate all the necessary matrices and operate with them. It was created as classes to be able to call more than one of the same element and change one attribute (for instance, to have the same object but two different grasps). The proposed work flow is to analysed thousands of different grasps and manipulator configurations to determine the best hand for each task and object. If this workflow is maintained and accomplished the objective of giving the technical specifications of the final manipulator will be achieved.

3.1 Current Work

In this moment I'm working on the force analysis for different grasp configurations using the task oriented metric to create a table were we can extract the necessary force for the contact normals to maintain the object in the air with out slipping, a work flow of the current work can be found in fig(3), for the project in general the HF type was chosen for its simplicity but the amount of information it saves.

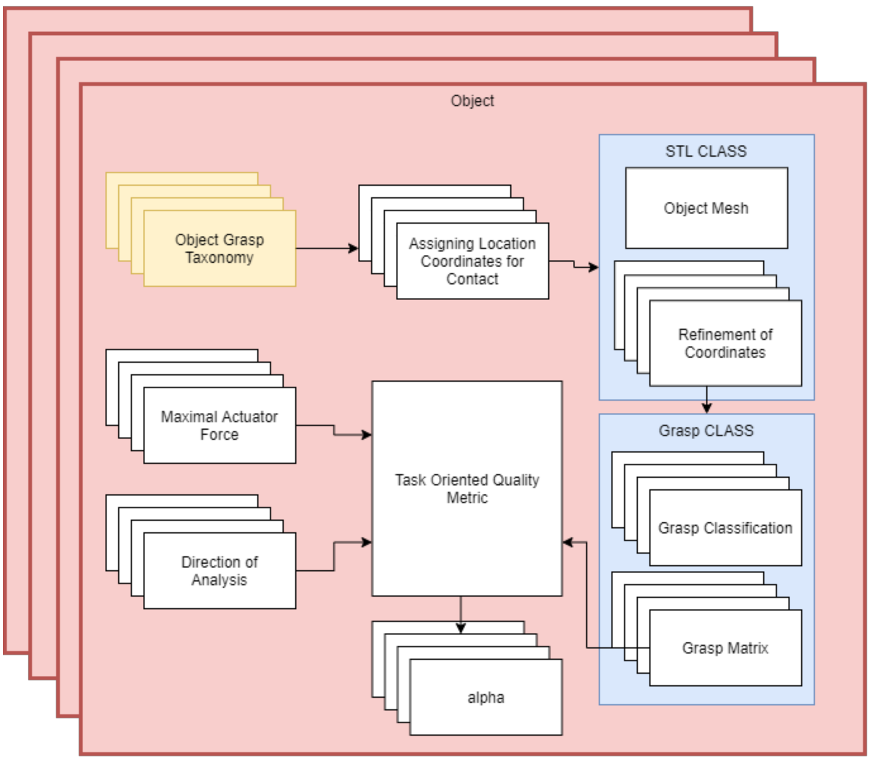

Figure 3: Flow of information to fill a table with Alphas

3.2 Future Work

For the moment the next steps aren't clear, We will be analysing each grasp taxonomy to determine the required force for the actuators, this information will be pertinent if we stay with a human hand architecture for the manipulator, if not, the next steps would be the automatic generation of different manipulators architectures to find the most appropriate ones.

4 Code

The code base to solve this project is being written in Python, all mathematical calculations are based in the equations presented in the *Grasping* section.

For ease of use and understanding the code is divided in different categories. Classes are their own file, mathematical tools used by more than one class have their own file, the quality metrics have their own file, and at last the data structures created are joined together.

To test the code I solved some of the examples proposed in the book[3].

4.1 Architecture

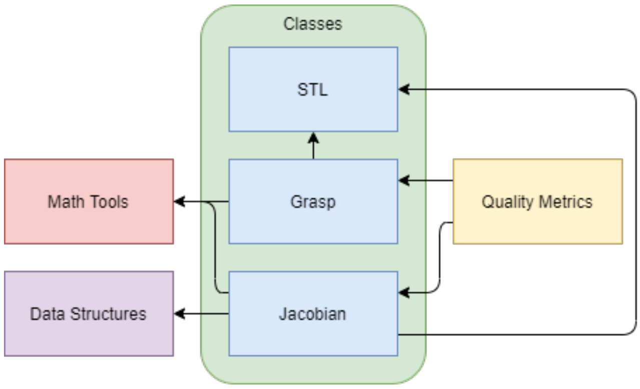

Figure 4: Code Architecture (the arrow points to the used file)

References

[1] CORDIS. *Traceable Robotic Handling of Sterile Medical Products*. Apr. 2021. URL: https://cordis.europa.eu/project/id/101017089.

[2] Caroline Pascale. *Development of an object-centered grasp analysis and synthesis framework*. Nov. 2020.

[3] Bruno Siciliano and Oussama Khatib. *Springer Handbook of Robotics*. 8th ed. Springer Science & Business Media, May 2008. ISBN: 9783540239574.

Acronyms & Notation

C_i Contact Point i. 4, 5

G Grasp Matrix. 4, 6

H Component Selection Matrix. 5

N Nullspace of a Matrix. 6

R_i Rotation Matrix of Contact Point i. 5

γ Torsional friction. 5

μ Tangential friction. 5

$\widetilde{G}$ *Complete* Grasp Matrix. 5

$\{B\}$ Object Frame. 4

$\{N\}$ World Frame. 4, 5, 12

g Sum of External Forces at p expressed in $\{N\}$. 4, 7

p Origin of Object Frame. 4, 12

CEA Atomic Energy Commissary. 3

DoF Degrees of Freedom. 6

FFC Frictional Form Closure. 6, 7

HF Hard Finger. 5

LP Linear Programming. 6, 7

nc Number of Contact Points. 4–7

ng Number of faces. 7

nq Number of joints. 8

PwoF Point with-out Friction. 5, 6

SF Soft Finger. 5

Glossary

Trivial A trivial system is one that has the vector zero as an answer. 6