School of Engineering, Discipline of Mechanical Engineering and Mechatronics, pp. 1–15

Final Year Project Part A

May 2022

Darcy Byrne [1]

[1] *Student of Mechatronics Engineering,*
The University of Newcastle, Callaghan, NSW 2308, AUSTRALIA
Student Number: 3256634
E-mail: D.Byrne@uon.edu.au

Abstract

Remember that your abstract may include the following information:

- Defines the intention of the report.
- Places the report in context so the reader knows why it is important to read it.
- Why is it important?
- What problem is addressed?
- Briefly states the results
- Briefly presents the implications and recommendations

Ensure that your abstract is less than 200 words.

Acknowledgements

You may like to say thank you to someone that helped you with your project.
(Kajita et al. 2002) - A realtime pattern generator for biped walking
(Kajita et al. 2003) - ZMP
(KATAYAMA et al. 1985) - Preview Control Discrete
(Perez 2014) - Fundamentals of Mechanical Systems Vec Calc etc.
(Renton 2021) - MCHA 4100

Contents

1. Introduction

To organise your introduction section you can use the following structure:

- **Position**: Show there is a problem and that it is important to solve it.
- **Problem**: Describe the specifics of the problem you are trying to address
- **Proposal**: Discuss how you are going to address this problem. Use the literature to back-up your approach to the problem, or to highlight that what you are doing has not been done before

Here you need to sell why what you are doing is important, and what benefits will it bring if you are successful and solve the problem?

1.1. Subsection title

You can use subsections within any section of the report.

1.2. Subsection title

Recall that you need at least two subsections per section.

1.2.1. Subsubsection 1

Do not use more than 2 levels of sub-sectioning.

1.2.2. Subsubsection 2

Do not use more than 2 levels of sub-sectioning.

The rest of the report is organised as follows. Section 2 describes items related to the core content. Section 7 concludes the report. Appendix A shows an example of how to make a Table.

2. Background

2.1. Mathematical Notation

In this section, examples of equations, figures, lists and code are provided.

2.1.1. Mathematics

LaTeX is very good for writing equations. Equations can be included in-line by using the $ symbols $y = mx + h$. Alternatively, equations can be written separately by using the `equation` environment—see (2.1) below.

$$y = mx + h. \tag{2.1}$$

When you use the `equaiton`, the expressions will be automatically numbered. If you do not want a number to be assigned, include an asterisk * when beginning the equation environment:

$$z = m_z x^2 + h_z.$$

The split command can be used within an equation environment to separate equations over multiple lines. Use an ampersand & on each line to specify how the equations should be aligned. Using split only provides a single equation number for multiple lines.

$$\begin{aligned} \dot{x} &= Ax + Bu, \\ y &= Cx + Du. \end{aligned} \tag{2.2}$$

If you want to have each line of a multi-line equation numbered, you can make used of the `align` environment,

$$\dot{x} = Ax + Bu, \tag{2.3}$$
$$y = Cx + Du, \tag{2.4}$$

where (2.3) is the first equation and (2.4) is the second. If you want to distinguish vectors from scalars you can use **bold** for vectors and matrices:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}u, \\ y &= \mathbf{C}\mathbf{x} + \mathbf{D}u, \end{aligned}$$

where u and y are scalar variables and $\mathbf{x}$ is a vector variable.

Matrices can be written by using the `bmatrix` command:

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \dots & A_{mn} \end{bmatrix}.$$

Greek letters can be written within equations by using appropriate commands. A few useful examples are:

- α, β

- γ, Γ
- θ, Θ
- ϕ, Φ, φ

You can also write Greek letters in bold using the `boldsymbol` command: $\boldsymbol{\alpha}$.

2.1.2. Figures

Figures can be included in your document by using the `figure` environment. Note that the position of the figure is decided by the latex compiler when creating the final document. Don't be surprised if it does not appear exactly where you expected it. Figure 1 below shows a plot of the function $\sin(x)/x$.

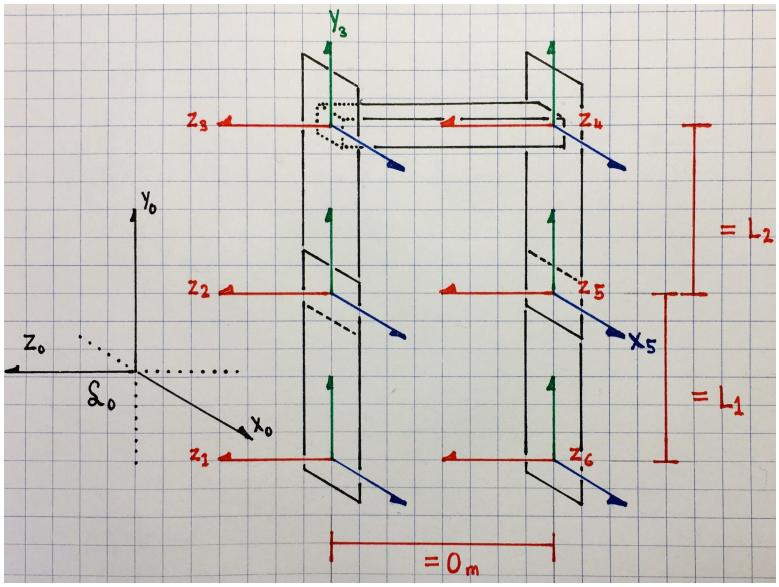

Figure 1: Here goes the caption.

Figures can be created using your favourite graphics editor before being included in your LaTeX document. If I need to make a simple diagram, I use powerpoint and select the drawing and save it as a pdf. For example, look at Figure 2.

To import the figures from Matlab, follow the following procedure:

1. Add labels and legends (don't forget to include units in the labels of each axis.)
2. From the file menu tag on the figure select export set up
3. Change the font size to 14 and click apply to figure
4. Export the figure as eps
5. Import it in LaTeX using the include graphics within a `figure` environment.

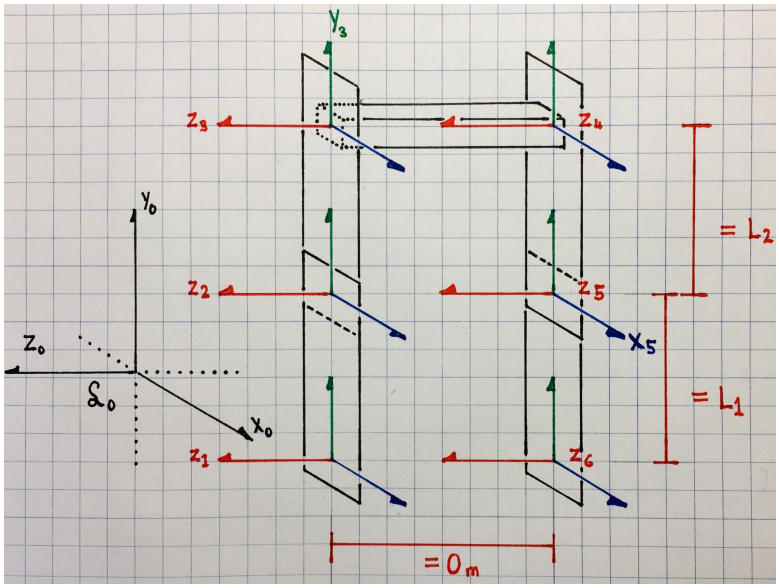

Figure 2: Here goes the caption.

2.1.3. Lists

To create lists use the environments `itemize`, `enumerate`, or `description`

The following is generated using *itemize*

- This is item 1
- This is item 2

The following is generated using *enumerate*

1) This is item 1
   a) Subitem a
   b) Subitem b
      i) Subsubitem i
      ii) Subsubitem ii
2) This is item 2

The following is generated using *description*

foo) This is item 1

bar) This is item 2

2.1.4. Code listings

To include a syntax-highlighted code listing, you can use the *listings* package. The default options are specified by the `\lstset` command. The default settings for this document have been configured

for Matlab code. There are 3 main commands, all of which can include options to override the defaults:

1. `\lstinline`: Command for including code fragments inline with the text, as an alternative to `\verb`. For example, we might describe function prototypes such as `int main(int argc, char *argv[])`.

2. `\begin{lstlisting}`,...,`\end{lstlisting}`: Environment for including a source code listing—embedded in the LaTeX source—in a box or floating environment.

```
figure(1)
hold on
grid on
% Plot the input voltage
plot(time,voltage)
% Plot the recovered voltage
plot(time,Ra*current + Kw*velocity,'+')
```

   The default style can be changed using the `style` command. A `CStyle` option has been defined within this temple for `.c` code.

```c
int add_function(int x, int y)
{
    /* Add inputs and return value */
    return x+y;
}

```

3. `\lstinputlisting`: Command for including a source code listing—loaded from an external file—in a box or floating environment. An example is shown in Listing **??**.

2.2. Numerical Optimisation

2.3. Kinematics

2.4. Kinematic Chains

2.5. Walking

3. Models

3.1. 2D Model

3.2. 3D Model

4. Quasi Static Locomotion

4.1. Quasi Static concept

4.2. 2D model implementation

4.3. 3D model implementation

5. Zero Moment Point Locomotion

5.1. Zero Moment Point concept

5.2. 3D Linear Inverted Pendulum

5.3. Preview Control

5.4. 3D Model Implementation

6. Results

6.1. 2D Model

6.2. 3D Model

7. Conclusion

This is one of the most important parts of the report. In the conclusion section, you should

- briefly summarise the results,
- reflect on the work presented,
- make recommendations,
- suggest future work or improvements.

References

Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K. & Hirukawa, H. (2003), 'Biped walking pattern generation by using preview control of zero-moment point', *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)* **3**.

Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Yokoi, K. & Hirukawa, H. (2002), 'A realtime pattern generator for biped walking', *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)* **4**.

KATAYAMA, T., OHKI, T., INOUE, T. & KATO, T. (1985), 'Design of an optimal controller for a discrete-time system subject to previewable demand', *International Journal of Control* **41**, 677–699.

Perez, T. (2014), *Fundamentals of Mechanical Systems*, The University of Newcastle.

Renton, C. (2021), 'Lecture notes: Mcha4100 - mechatronics systems'.

A. Example of a Table