Department of Mechanical and Mechatronics Engineering

# Title

*Author:*
James Graham-Hu

*Student Number:*
20555690

Date: August 22, 2019

August 22, 2019

William Melek, Director
Mechatronics Engineering
University of Waterloo
Waterloo, Ontario
N2L 3G1

Dear Professor Melek,

This report entitled "Title", was prepared as my Work Report 400 for the Department of Mechanical and Mechatronics Engineering at the University of Waterloo for the 4A term. Purpose of report.

Description of Amii.

Description of project and motivation behind project.

Thank yous. This report was written entirely by me and has not received any previous academic credit at this or any other institution.

Sincerely,

James Graham-Hu
ID 20555690
4A Mechatronics Engineering

# Contents

# List of Figures

# List of Tables

# 1 Summary

Summary

# 2 Introduction

## 2.1 Automatic Levelling Wrist Background

Powered wrist movement is rare in commercial systems, and many powered protheses have only one degree of freedom (DoF), usually rotation [2]. These limitations in ease of wrist movement in many upper limb prostheses force people with major upper limb loss to use compensatory movements [1]. Compensation occurs in trunk, shoulder, and elbow movements has been associated with causing musculoskeletal pain in the neck, upper back, shoulder, and remaining arm [3].
A two DoF automatic levelling wrist was developed by Dylan J. A. Brenneis in 2019 that addresses the issues with ease of wrist movement in wrist prosthetics [**?**]. Why it is useful. Uses a PID controller with manually tuned gains to level the wrist to make tasks easier [Dylan's thesis]. Users reported reliability issues [Dylan's thesis]. A more reliable control method should be implemented to improve the user experience.

### 2.1.1 Automatic Levelling Method

Angle definition Angle correction PID controller

## 2.2 Reinforcement Learning Background

Agent-Environment-Reward diagram

### 2.2.1 Q-Learning

Table action-values Update

### 2.2.2 Function Approximation

Update equation with function approx

## 2.3 Neural Network Background

Weights and forward prop, neurons, activation function

### 2.3.1 Backpropagation

Chain rule

### 2.3.2 Levenberg-Marquardt Algorithm

Show equation

# 3 Project Outline

## 3.1 Problem Definition

Amii needs demos that demonstrate the capabilities of ml Amii needs a hardware demo specifically The automatic levelling wrist provides is a good candidate for improvement using ml

## 3.2 Objective

Design an ML system to improve the performance of the automatic levelling wrist

## 3.3 Constraints

- The solution must improve the performance of the automatic levelling wrist.
- The solution must implement machine learning in some way.

## 3.4 Criteria

The following criteria considered in choosing an appropriate solution are chosen such that the automatic levelling wrist reliably improves the user's ability to use the prosthetic.

- The solution should minimize steady state error.
- The solution should minimize response time.
- The solution should minimize settling time.
- The solution should maximize reliability and consistency.

Due to the nature of the problem, training time and data are difficult to acquire. Without a sophisticated simulation, data and training can only be collected and run in real time. Therefore, the solution should require minimal training time and data.

# 4 Proposed Solutions

## 4.1 Reinforcement Learning Methods

Tabular Q-learning Q-learning with function approximation

## 4.2 Neural Network Methods

Neural Network Controller PID auto-tuning trained with a model, using the Levenberg-Marquardt algorithm PID auto-tuning trained model-free, using the Levenberg-Marquardt algorithm PID auto-tuning trained model-free, using backpropagation

# 5 Solution Evaluation

## 5.1 Q-learning - Tabular

Tabular control wasn't granular enough to enable precise control of the servo Even with a minimized state space (only using the IMU angle), it was still difficult to visit all states and learn making it unreliable.

## 5.2 Q-learning - Function Approximation

Not enough data to train the neural network, difficult to visit all states, no simulation (must train real-time which is very slow especially if testing different hyperparameters) Deadly triad (bootstrapping, function approximation and off-policy learning) - not guaranteed to converge

## 5.3 Neural Network Controller

Not enough data to generalize, better to make use of PID controller

## 5.4 PID Auto-Tuning - Trained with Model, using Levenberg-Marquardt

Requires a simulation for the servo, however the level of sophistication for the simulation does not need to be extremely high as in an RL setting. Uses a PID controller (so not starting from scratch) which makes it more reliable, as the PID controller on its own will achieve the objective of automatic levelling to a certain level. Therefore a simple simulation can be a starting point. Quick training for the neural network, as a simulation can tune the neural network many times faster than real time.

## 5.5 PID Auto-Tuning - Trained Model-Free, using Levenberg-Marquardt

Quick training for the neural network, as a simulation can tune the neural network many times faster than real time. However, another neural network must be trained to emulate the real system which requires real data. It was found that there was not enough real data to generalize to all situations. (generalizes well to what it saw, but not anything else like the APRBS used to find the jacobian)

## 5.6 PID Auto-Tuning - Trained Model-Free, using backpropagation

Same problems as above, as well as being difficult to implement, due to the statefulness of the PID controller.

### 5.7 Chosen Solution

The chosen solution was PID auto-tuning trained using a model, using the Levenberg-Marquardt algorithm.

## 6 PID Auto-Tuning Implementation

Required components: a transfer function or ode for the servo, a simulation, a neural network, a numerical implementation of the levenberg marquardt algorithm, a numerical jacobian calculator, an APRBS generator, a neural network implemented in C#

### 6.1 Servo Transfer Function

### 6.2 Single DoF Simulation

Best parameters for simulation

### 6.3 Neural Network Structure

5 inputs (error, angle, velocity), 4 nodes in hidden layer, 3 output nodes, leaky relu with alpha=0.3 activation function. Absolute value of output to keep positive (can't have negative gains)

### 6.4 Levenberg-Marquardt Algorithm Implementation

### 6.5 Numerical Jacobian Calculation

### 6.6 Amplitude Modulated Pseudo-Random Binary Signal (APRBS) Generation

Table 1: Sample table2

| Requests/Second | Required Size | Cores | Memory (GiB) | Cost ($/month) |
|---|---|---|---|---|
| 10 | t2.small | 1 | 2 | 16.56 |
| 100 | t2.medium | 2 | 4 | 33.41 |
| 1000 | t2.large | 2 | 8 | 66.82 |
| 10000 | t2.xlarge | 4 | 16 | 133.63 |

$$Sample equation \tag{1}$$

Figure 1: Sample figure

# 7 Results

# 8 Conclusion

# 9 Recommendations

Better Simulation (2 DoF, better approximation of moment of inertia for different servo positions, take torque due to gravity into account)

## References

[1] S. L. Carey, M. J. Highsmith, M. E. Maitland, and R. V. Dubey, “Compensatory movements of transradial prosthesis users during common tasks”, Clinical Biomechanics, vol. 23, no. 9, pp. 1128 –1135, 2008.

[2] N. M. Bajaj, A. J. Spiers, and A. M. Dollar, “State of the art in prosthetic wrists: Commercial and research devices”, in 2015 IEEE International Conference on Rehabilitation Robotics (ICORR), Aug. 2015, pp. 331–338.

[3] K. Østlie, R. J. Franklin, O. H. Skjeldal, A. Skrondal, and P. Magnus, “Musculoskeletal pain and overuse syndromes in adult acquired major upper-limb amputees”, Archives of Physical Medicine and Rehabilitation, vol. 92, no. 12, pp. 1967 –1973, 2011.

d.j.a.brenneis file:///C:/Users/James/Documents/Bypass$_{P}rothesis/Brenneis_{D}ylan_{J}A_{2}01903_{M}Sc.pd$

# Appendices

## A Appendix