DEEP LEARNING FOR MAP-LESS NAVIGATION: SMOOTH VELOCITY CHANGE

by

WALTER TAY ANN LEE

A THESIS SUBMITTED FOR THE DEGREE OF

BACHELOR OF ENGINEERING

in

MECHANICAL ENGINEERING

in the

UNDERGRADUATE DIVISION

of the

NATIONAL UNIVERSITY OF SINGAPORE

2020

Supervisor:
Associate Professor ZHANG Yunfeng

Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Walter TAY Ann Lee

10 March 2020

To my friends from ME and USP

Acknowledgments

I would like to thank my parents, Tay Boon Chit and Lim Swee Yin, whom without I would not have been in NUS. I thank my brother, David Tay Ang Shun, who has been supportive of me in many ways. I thank all my friends in Cinnamon College and in the Mechanical Engineering Department, who have made my undergraduate life a rich and fulfilling one.

Contents

Acknowledgments **iii**

Summary **ix**

List of Figures **xi**

List of Tables **xiii**

1 Introduction **1**

1.1 Brief Description of Project 1

1.2 Problem Definition 1

2 Background **5**

2.1 Deep Reinforcement Learning 5

2.1.1 Markov Decision Process 6

2.1.2 Policies 6

2.1.3 Reward and Return 7

2.1.4 The RL Problem 7

2.1.5 Value Functions 8

2.1.6 The Optimal Q-Function and the Optimal Action 8

2.1.7 Bellman Equations 9

2.1.8 Advantage Functions 10

2.2 Deep Deterministic Policy Gradient (DDPG) 10

2.2.1 Quick Facts 11

2.2.2 The Q-Learning Side of DDPG 11

2.2.3 The Policy Learning Side of DDPG 13

3 Main Text **15**

3.1 Smooth Motion Profile 15

3.2 Proposed Approach . . . 16
3.2.1 Three Different Models . . . 16
3.2.2 State and Action . . . 17
3.2.3 Simulation Setup . . . 18
3.2.4 Reward Function . . . 18
3.2.5 Network Architecture . . . 19
3.2.6 Training Methodology . . . 21
3.3 Experiment Results . . . 21
3.4 Discussion . . . 21

4 Like White on Rice 23
4.1 Preliminaries . . . 24
4.2 Methodology . . . 25
4.3 Evaluation . . . 26
4.4 Summary . . . 28

5 Use Your Noodle 29
5.1 Preliminaries . . . 30
5.2 Methodology . . . 32
5.3 Evaluation . . . 33
5.4 Summary . . . 35

6 Conclusion and Future Work 37
6.1 Conclusion . . . 37
6.2 Future Research Directions . . . 38
6.2.1 Eat good, Feel good, Look Good . . . 38
6.2.2 An Apple a Day Keeps the Doctor Away . . . 38
6.2.3 A Healthy Food for a Wealthy Mood . . . 39
6.2.4 Eat Your Veggies, Have Less Wedgies . . . 39
6.2.5 Salad and Beets Are Some Healthy Treats . . . 39
6.2.6 Eat Only When Hungry . . . 40

Bibliography 41

A Stage Simulation Worlds **43**
A.1 Blank World 43
A.2 Simple World 44
A.3 Complex World 45

B Recipes **47**
B.1 Mustard Green Pork Rice 47
B.2 Hakka Pancake Roll 48
B.3 Pumpkin Tang Yuan 49

C Ingredients **51**
C.1 Hokkien Mee: Stir-fry Noodles with Shrimps and Squids 51
C.2 Nasi Lemak: Rice Boiled with Coconut Milk and Accompanied with Chilli Paste and Anchovies 51
C.3 Prata: a Dough Served with Curry 52
C.4 Fried Carrot Cake: Stir-fry Rice Cakes with Eggs 52

Summary

Deep learning for map-less navigation: Smooth Velocity Change

by

Walter TAY Ann Lee

Bachelor of Engineering in Mechanical Engineering

National University of Singapore

Summary

List of Figures

1.1 Histogram shows the angular velocity of a well-trained robot in 100 episodes of simulated map-less navigation in Complex World A.1 . . 2

1.2 Histogram shows the angular jerk of a well-trained robot in 100 episodes of simulated map-less navigation in Complex World A.1 . . 3

3.1 Comparison of Trapezoidal and S-Cruve Motion Profiles. Image from Meckl and Arestides, 1998 [5] 16

3.2 Network architecture. Network layers are demonstrated by rectangles. Orange arrows indicate the connectivity between network layers and some other components, e.g. input state and the output of the simple controller. Blue arrows indicate the operation of concatenation (i.e. tf.concat). The final action is selected based on the Q-value predicted by the critic-DQN. 20

4.1 A bowl of rice. 25

4.2 Taste with meal repetition. 28

4.3 Taste with meal freshness. 28

5.1 A bowl of noodles. 31

List of Tables

C.1 Table to test captions and labels 51

Chapter 1

Introduction

1.1 Brief Description of Project

This work presents a case study of a learning-based approach to smoothing the movement of a real robotic platform. The robotic platform is operated by an end-to-end neural network which is trained using deep reinforcement learning (DRL) to perform target driven map-less navigation. Previous works [6], [7], [9] have focused more on improving the path-finding or trajectory-generating aspect of map-less navigation while this work focuses on the motion profile planning of map-less navigation. That is, the main idea of this work is to show that **smoothing the motion profile generated by the DRL model can be achieved through a learning-based approach** and **show the potential improvements to the motion profile that such an approach may involve**. A smooth motion profile is taken here to mean a motion profile that has minimal jerk (time derivative of acceleration).

1.2 Problem Definition

Consider the following histogram of the robot's angular velocity:

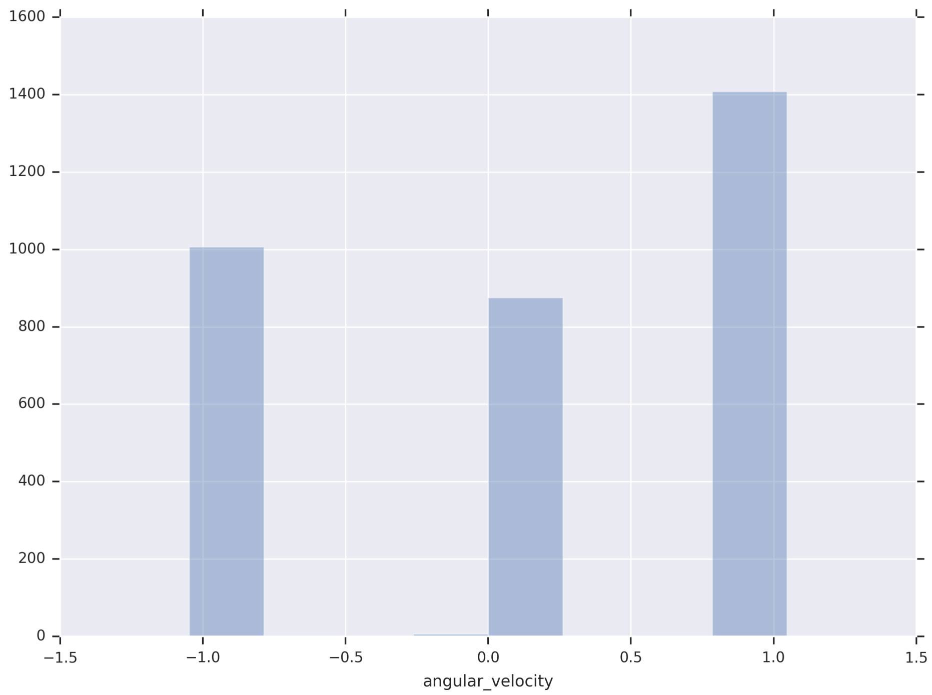


Figure 1.1: Histogram shows the angular velocity of a well-trained robot in 100 episodes of simulated map-less navigation in Complex World A.1

From Figure 1.1, we observe that after sufficient training, the DRL model only chooses extreme actions for the robot to take (i.e. maximum velocity at 1.0 and -1.0, and minimum velocity at 0). Furthermore, to perform these actions, the robot has to accelerate instantaneously from 0 to 1.0 and vice versa. In practice, this is not possible for the robot due to acceleration limits; however, conventional DRL models for map-less navigation *implicitly assume* that arbitrary motion in the configuration space is allowed as long as obstacles are avoided. Moreover, even if this motion profile were possible it would not be ideal as it creates unnecessary jerk and jerk-induced vibrations for the robot.

Figure 1.2 shows that the robot does indeed experience jerk (and at specific magnitudes):

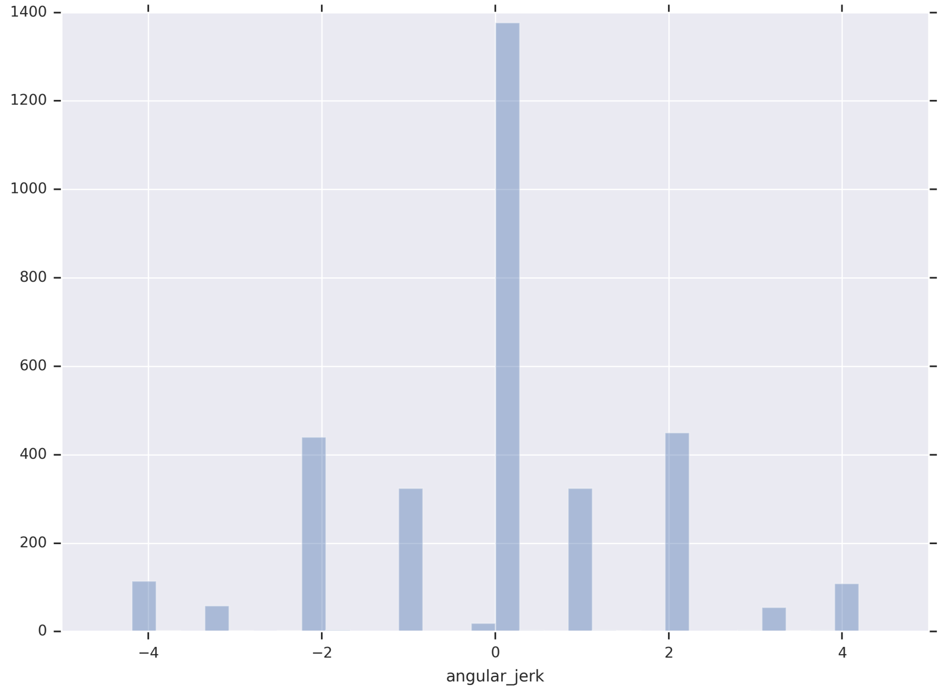


Figure 1.2: Histogram shows the angular jerk of a well-trained robot in 100 episodes of simulated map-less navigation in Complex World A.1

Motion control methods that achieve fast motions without residual vibrations are already well-known [5]. Thus, applying an s-curve velocity profile to velocities generated by the DRL model should be able to reduce jerk and vibrations.

However, the main contribution of this paper is to show that **jerk reduction can be learnt by the DRL model itself and the outcomes of that learning**. That is, instead of splitting the navigation task into multiple submodules (e.g. map-less navigation and motion profile planning), this work represents a case study of an end-to-end approach which uses a direct mapping from sensor data to robot motion commands that achieves both map-less navigation and smooth motion. This is achieved by including jerk as an additional feature in the reward function of the DRL model.

Chapter 2

Background

This section provides the background required to understand this work. It is *not intended* for the reader who already has an understanding of Deep Reinforcement Learning and Deep Deterministic Policy Gradient (DDPG).

We would like to note that a large portion of this section has been reproduced from OpenAI's Spinning Up in Deep RL [1]. What is different in this reproduction, however, is that the background information on DRL has been adapted to fit the ideas and concepts related to map-less navigation found in this work.

2.1 Deep Reinforcement Learning

Deep reinforcement learning (DRL) is the combination of reinforcement learning and deep learning. This idea of using neural networks for reinforcement learning, however, is not new and can be dated all the way back to Teasauro's TD-Gammon [8]. In the early 2010s, however, the field of deep learning began to find groundbreaking success, particularly in speech recognition [2] and computer vision [4]. This success, combined with the advances in computing power, allowed the revival in interest of using deep neural networks as universal function approximators for reinforcement learning - leading to deep reinforcement learning.

Deep reinforcement learning is useful when [1]:

- we have a sequential decision-making problem (which we can represent as a Markov Decision Process)
- we do not know the optimal behaviour (e.g. multi-modal problem)

- but we can still evaluate whether behaviours are good or bad

2.1.1 Markov Decision Process

To train our agent with deep reinforcement learning, we must be able to model the relationship between the agent and its environment. A Markov Decision Process (MDP) is a mathematical object that describes our agent interacting with a stochastic environment. It is defined by the following components:

- S: **state space**, a set of states of the environment. This is the set of inputs for our robot which contain a view of the world (i.e. a stack of laser scans, the current speed of the robot, and the target position with respect to the robot's local frame)
- A: **action space**, a set of actions, which the agent selects from each timestep. The actions taken by our robot to reach its target are its linear velocity and angular velocity, both of which are continuous-valued variables
- $P(r, s'|s, a)$: **transition probability distribution**. For each state s and action a, P specifies the probability that the environment will emit reward r and transition to state s'. This transition function describes the relationship between states, actions, next states, and rewards in an environment.

2.1.2 Policies

The end goal is to find a policy π, which tells the agent what actions to take given a state. In DRL, we use parameterized policies: policies whose outputs are computable functions that depend on a set of parameters (e.g. the weights and biases of a neural network) which we can adjust to change the behavior via some optimization algorithm [1].

We denote the parameters of such a policy by θ, and then write this as a subscript on the policy symbol to highlight the connection:

$$a_t \sim \pi_\theta(\cdot|s_t) \tag{2.1}$$

2.1.3 Reward and Return

The reward function R is critically important in reinforcement learning (and specifically for this work). It depends on the current state of the world, the action just taken, and the next state of the world:

$$r_t = R(s_t, a_t, s_{t+1}) \tag{2.2}$$

The goal of an agent is to maximise the cumulative reward over a trajectory. In this work, the type of return used is called an **infinite-horizon discounted return**, which is the sum of all rewards ever obtained by the agent, but discounted by how far off in the future they're obtained. This formulation of reward includes a discount factor $\gamma \in (0, 1)$

Specifically (prior to adding the reward feature proposed in this work), this work uses the reward function for map-less navigation proposed in Xie et. al's AsDDPG network [9]:

$$Reward, r_t = \begin{cases} R_{crash} & \text{if robot crashes,} \\ R_{reach} & \text{if robot reaches the goal,} \\ \gamma((d_{t-1} - d_t)\Delta t - C) & \text{otherwise.} \end{cases}$$

2.1.4 The RL Problem

The goal of reinforcement learning is to select a policy which maximises **expected return** when the agent acts according to it. To talk about expected return, we first have to talk about probability distributions over trajectories.

Let's suppose that both the environment transitions and the policy are stochastic. In this case, the probability of a T-step trajectory is:

$$P(\tau|\pi) = p_0(s_0) \prod_{t=0}^{T-1} P(s_{t-1}|s_t, a_t)\pi(a_t|s_t) \tag{2.3}$$

The expected return, denoted by $J(\pi)$, is then:

$$J(\pi) = \int_\tau P(\tau|\pi)R(\tau) = \tau \sim \pi R(\tau) \tag{2.4}$$

The central optimization problem in RL can then be expressed by:

$$\pi^* = arg\underset{\pi}{max} J(\pi) \tag{2.5}$$

with π being the optimal policy.

2.1.5 Value Functions

It's often useful to know the **value** of a state, or state-action pair. By value, we mean the expected return if you start in that state or state-action pair, and then act according to a particular policy forever after. Value functions are used, one way or another, in almost every RL algorithm.

There are four main functions of note:

1. The **On-Policy Value Function**, $V^{\pi}(s)$, which gives the expected return if you start in a state s and always act according to policy π:

$$V^{\pi}(s) = \tau \sim \pi R(\tau) | s_0 = s \tag{2.6}$$

2. The **On-Policy Action-Value Function**, $Q^{\pi}(s,a)$, which gives the gives the expected return if you start in state s, take an arbitrary action a (which may not have come from the policy), and then forever after act according to policy π:

$$Q^{\pi}(s,a) = \tau \sim \pi R(\tau) | s_0 = s, a_0 = a \tag{2.7}$$

3. The **Optimal Value Function**, $V^{*}(s)$, which gives the expected return if you start in state s and always act according to the *optimal* policy in the environment

$$V^{*}(s) = \underset{\pi}{max} \tau \sim \pi R(\tau) | s_0 = s \tag{2.8}$$

4. The **Optimal Action-Value Function**, $Q^{*}(s,a)$, which gives the expected return if you start in state s, take an arbitrary action a, and then forever after act according to the *optimal* policy in the environment:

$$Q^{*}(s,a) = \underset{\pi}{max} \tau \sim \pi R(\tau) | s_0 = s, a_0 = a \tag{2.9}$$

2.1.6 The Optimal Q-Function and the Optimal Action

There is an important connection between the optimal action-value function $Q^{*}(s,a)$ and the action selected by the optimal policy. By definition, $Q^{*}(s,a)$ gives the expected return for starting in state s, taking (arbitrary) action a, and then acting according to the optimal policy forever after.

The optimal policy in s will select whichever action maximises the expected return from starting in s. As a result, if we have Q^*, we can directly obtain the optimal action, $a^*(s)$, via:

$$a^*(s) = \arg\max_a Q^*(s,a) \tag{2.10}$$

This connection is important; some RL algorithms learn by improving their policy (thereby directly obtaining "good" actions), whereas others learn by improving their Q-Function (which allows them to determine the best action from a set of actions, indirectly obtaining "good" actions).

2.1.7 Bellman Equations

All four of the value functions obey special self-consistency equations called **Bellman equations**. The basic idea behind the Bellman equations is this:

The value of your starting point is the reward you expect to get from being there, plus the value of wherever you land next.

The Bellman equations for the on-policy value functions are:

$$V^{\pi}(s) = \underset{\substack{a\sim\pi \\ s'\sim P}}{E}[r(s,a)+\gamma V^{\pi}(s')] Q^{\pi}(s,a) = \underset{s'\sim P}{E}[r(s,a)+\gamma \underset{a'\sim\pi}{E}[Q^{\pi}(s',a')]] \tag{2.11}$$

where $s' \sim P$ is shorthand for $s' \sim P(\cdot|s,a)$, indicating that the next state s' is sampled from the environment's transition rules; $a \sim \pi$ is shorthand for $a \sim \pi(\cdot|s)$; and $a' \sim \pi$ is shorthand for $a' \sim \pi(\cdot|s')$.

The Bellman equations for the optimal value functions are:

$$V^*(s) = \underset{a}{max} \underset{s'\sim P}{E}[r(s,a) + \gamma V^*(s')] Q^*(s,a) = \underset{s'\sim P}{E}[r(s,a) + \gamma \underset{a'}{max}[Q^*(s',a')]] \tag{2.12}$$

The crucial difference between the Bellman equations for the on-policy value functions and the optimal value functions, is the absence or presence of the *max* over actions. Its inclusion reflects the fact that whenever the agent gets to choose its action, in order to act optimally, it has to pick whichever action leads to the highest value.

2.1.8 Advantage Functions

Sometimes in RL, we don't need to describe how good an action is in an absolute sense, but only how much better it is than others on average. That is to say, we want to know the relative **advantage** of that action. We make this concept precise with the **advantage function**.

The advantage function $A^{\pi}(s, a)$ corresponding to a policy π describes how much better it is to take a specific action a in state s, over randomly selecting an action according to $\pi(\cdot|s)$, assuming you act according to π forever after. Mathematically, the advantage function is defined by

$$A^{\pi}(s, a) = Q^{\pi}(s, a) - V^{\pi}(s) \tag{2.13}$$

2.2 Deep Deterministic Policy Gradient (DDPG)

The network presented in this work is a variant of DDPG [9]. As such, it is crucial that we also provide a background for what DDPG is, and how it works.

Deep Deterministic Policy Gradient (DDPG) is an algorithm which concurrently learns a Q-function and a policy. It uses off-policy data and the Bellman equation to learn the Q-function, and uses the Q-function to learn the policy.

This approach is closely connected to Q-learning, and is motivated the same way: if you know the optimal action-value function $Q^*(s, a)$, then in any given state, the optimal action $a^*(s)$ can be found by solving:

$$a^*(s) = \arg\max_a Q^*(s, a) \tag{2.14}$$

DDPG interleaves learning an approximator to $Q^*(s, a)$ with learning an approximator to $a^*(s)$, and it does so in a way which is specifically adapted for environments with continuous action spaces. But what does it mean that DDPG is adapted specifically for environments with continuous action spaces? It relates to how we compute the max over actions in $\max_a Q^*(s, a)$.

When there are a finite number of discrete actions, the max poses no problem, because we can just compute the Q-values for each action separately and directly compare them. (This also immediately gives us the action which maximizes the Q-value.) But when the action space is continuous, we can't exhaustively evaluate the space, and solving the optimization problem is highly non-trivial. Using a normal optimization algorithm would make calculating $\max_a Q^*(s,a)$ a painfully expensive subroutine. And since it would need to be run every time the agent wants to take an action in the environment, this is unacceptable.

Because the action space is continuous, the function $Q^*(s,a)$ is presumed to be differentiable with respect to the action argument. This allows us to set up an efficient, gradient-based learning rule for a policy $\mu(s)$ which exploits that fact. Then, instead of running an expensive optimization subroutine each time we wish to compute $\max_a Q(s,a)$, we can approximate it with $\max_a Q(s,a) \approx Q(s,\mu(s))$.

2.2.1 Quick Facts

- DDPG is an off-policy algorithm
- DDPG can only be used for environments with continuous action spaces
- DDPG can be thought of as being deep Q-learning for continuous action spaces

Next, we'll explain the math behind the two parts of DDPG: learning a Q function, and learning a policy.

2.2.2 The Q-Learning Side of DDPG

First, let's recap the Bellman equation describing the optimal action-value function, $Q^*(s,a)$. It's given by

$$Q^*(s,a) = \underset{s' \sim P}{E}[r(s,a) + \gamma \max_{a'} Q^*(s',a')] \tag{2.15}$$

where $s' \sim P$ is shorthand for saying that the next state, s', is sampled by the environment from a distribution $P(\cdot|s,a)$.

This Bellman equation is the starting point for learning an approximator to $Q^*(s,a)$. Suppose the approximator is a neural network $Q_\phi(s,a)$, with parameters ϕ, and that we have collected a set $\mathcal{D}$ of transitions (s,a,r,s',d) (where d indicates whether state s' is terminal). We can set up a mean-squared

Bellman error (MSBE) function, which tells us roughly how closely Q_ϕ comes to satisfying the Bellman equation:

$$L(\phi, \mathcal{D}) = \underset{(s,a,r,s',d)\sim\mathcal{D}}{\mathrm{E}} \left[\left(Q_\phi(s,a) - \left(r + \gamma(1-d) \max_{a'} Q_\phi(s',a') \right) \right)^2 \right] \tag{2.16}$$

Here, in evaluating $(1-d)$, we've used a Python convention of evaluating True to 1 and False to zero. Thus, when d==True—which is to say, when s' is a terminal state—the Q-function should show that the agent gets no additional rewards after the current state.

Q-learning algorithms for function approximators, such as DQN (and all its variants) and DDPG, are largely based on minimizing this MSBE loss function. There are two main tricks employed by all of them which are worth describing, and then a specific detail for DDPG.

Trick One: Replay Buffers. All standard algorithms for training a deep neural network to approximate $Q^*(s,a)$ make use of an experience replay buffer. This is the set $\mathcal{D}$ of previous experiences. In order for the algorithm to have stable behavior, the replay buffer should be large enough to contain a wide range of experiences, but it may not always be good to keep everything. If you only use the very-most recent data, you will overfit to that and things will break; if you use too much experience, you may slow down your learning. This may take some tuning to get right.

Trick Two: Target Networks. Q-learning algorithms make use of target networks. The term

$$r + \gamma(1-d) \max_{a'} Q_\phi(s',a') \tag{2.17}$$

is called the target, because when we minimize the MSBE loss, we are trying to make the Q-function be more like this target. Problematically, the target depends on the same parameters we are trying to train: ϕ. This makes MSBE minimization unstable. The solution is to use a set of parameters which comes close to ϕ, but with a time delay—that is to say, a second network, called the target network, which lags the first. The parameters of the target network are denoted ϕ_{targ}.

In DQN-based algorithms, the target network is just copied over from the main network every some-fixed-number of steps. In DDPG-style algorithms, the target network is updated once per main network update by polyak averaging:

$$\phi_{\text{targ}} \leftarrow \rho\phi_{\text{targ}} + (1-\rho)\phi, \tag{2.18}$$

where ρ is a hyperparameter between 0 and 1 (usually close to 1).

DDPG Detail: Calculating the Max Over Actions in the Target. As mentioned earlier: computing the maximum over actions in the target is a challenge in continuous action spaces. DDPG deals with this by using a target policy network to compute an action which approximately maximizes $Q_{\phi_{\text{targ}}}$. The target policy network is found the same way as the target Q-function: by polyak averaging the policy parameters over the course of training.

Putting it all together, Q-learning in DDPG is performed by minimizing the following MSBE loss with stochastic gradient descent:

$$L(\phi, \mathcal{D}) = \underset{(s,a,r,s',d)\sim\mathcal{D}}{\mathrm{E}}\left[\left(Q_\phi(s,a) - \left(r + \gamma(1-d)Q_{\phi_{\text{targ}}}(s', \mu_{\theta_{\text{targ}}}(s'))\right)\right)^2\right], \tag{2.19}$$

where $\mu_{\theta_{\text{targ}}}$ is the target policy.

2.2.3 The Policy Learning Side of DDPG

Policy learning in DDPG is fairly simple. We want to learn a deterministic policy $\mu_\theta(s)$ which gives the action that maximizes $Q_\phi(s,a)$. Because the action space is continuous, and we assume the Q-function is differentiable with respect to action, we can just perform gradient ascent (with respect to policy parameters only) to solve

$$\max_\theta \underset{s\sim\mathcal{D}}{\mathrm{E}}\left[Q_\phi(s, \mu_\theta(s))\right]. \tag{2.20}$$

Note that the Q-function parameters are treated as constants here. Exploration vs. Exploitation

DDPG trains a deterministic policy in an off-policy way. Because the policy is deterministic, if the agent were to explore on-policy, in the beginning it would probably not try a wide enough variety of actions to find useful learning signals. To make DDPG policies explore better, we add noise to their actions at training time.

At test time, to see how well the policy exploits what it has learned, we do not add noise to the actions.

Chapter 3

Main Text

We would like to reiterate that the main contribution of this work as a case study is to **demonstrate a learning-based approach to smoothing the motion profile of a DRL model**. We do this in the following sequence:

1. Introduce the idea of "smooth" motion profiles

2. Present the network architecture used to apply this learning-based approach

3. Discuss the empirical findings of this learning-based approach as well as comparing it with a non-learning based approach

3.1 Smooth Motion Profile

In map-less navigation, we deal with point-to-point motion. That is, the goal of the robot is to move from its starting position and reach some target end position while avoiding obstacles. In terms of motion, point-to-point means that from a stop (zero velocity), the load is accelerated to a constant velocity, and then decelerated such that the final acceleration, and velocity, are zero at the moment the load arrives at the programmed destination. However, achieveing fast motions without residual vibration is a challenge that pervades many applications of point-to-point motion [5], including deep reinforcement learning for map-less navigation.

To perform point-to-point motion with minimal vibration, previous works have focused on optimization of the motion profile itself [5]. For example,

Figure 3.1 shows a comparison between a typical trapezoidal motion profile and an (improved) S-curve motion profile.

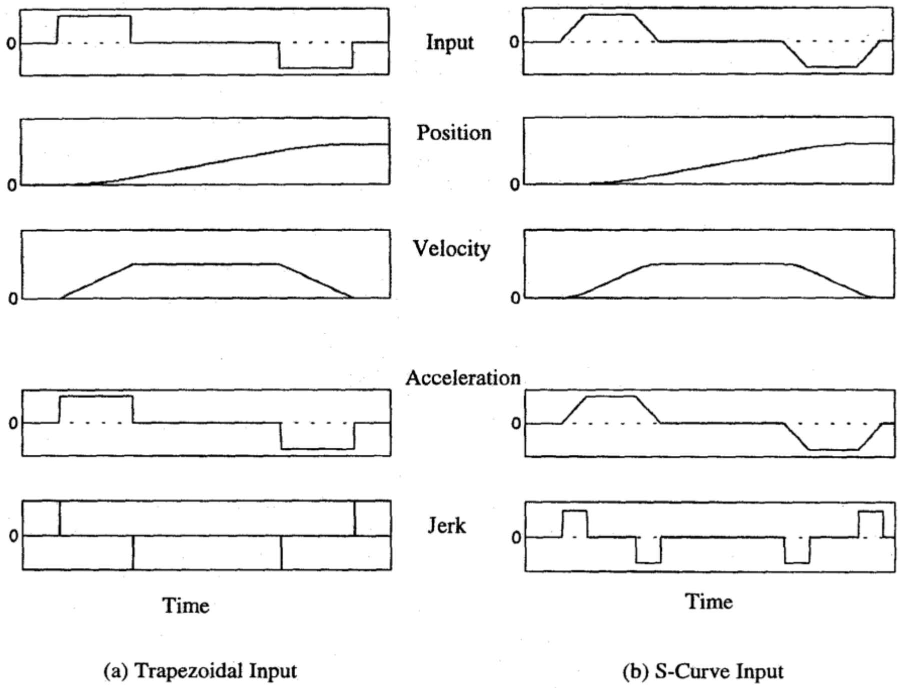


Figure 3.1: Comparison of Trapezoidal and S-Cruve Motion Profiles. Image from Meckl and Arestides, 1998 [5]

As can be seen from Figure 3.1, smoothing a motion profile involves jerk reduction (or some manipulation of the jerk profile). Jerk is the time-derivative of acceleration, that is:

$$j(t) = \frac{da(t)}{dt} = \frac{d^2v(t)}{dt^2} \tag{3.1}$$

3.2 Proposed Approach

3.2.1 Three Different Models

In this work, we compare the motion profiles of the following three models:

1. **Normal Model**: The default end-to-end neural network trained with deep reinforcement learning to perform map-less navigation
2. **Normal Model with Velocity Smoother**: The same network as Normal Model but trained with a simple velocity smoother

3. **Model with Jerk-Learning**: A network very similar to Normal Model but with an added reward function that penalizes the robot for the jerk generated. The inputs of this model are also slightly different (for the reward function to work properly).

3.2.2 State and Action

Normal Model and *Model with Jerk-Learning* have different features included in their inputs (states) but they both have the same output (action).

At a certain time step t, the state of the agent (robot) trained with *Normal Model* is described by:

- a stack of three 270-degree sparse laser scans y (y_t, y_{t-1}, y_{t-2})
- the current linear and angular velocity of the robot (v_t, ω_t)
- x and y components of the robot's displacement from its target position $(d_{x,t}, d_{y,t})$

In the *Model with Jerk-Learning*, the acceleration and jerk of the robot are included in the input layer. The idea is to allow the robot to use information about its motion profile to reduce the jerk generated (in combination with the reward function to incentivize this behaviour). The inputs for this model are:

- a stack of three 270-degree sparse laser scans y (y_t, y_{t-1}, y_{t-2})
- a stack of linear and angular velocities of the robot $(v_t, \omega_t, v_{t-1}, \omega_{t-1}, v_{t-2}, \omega_{t-2})$
- a stack of linear and angular accelerations of the robot $(a_t, \alpha_t, a_{t-1}, \alpha_{t-1}, a_{t-2}, \alpha_{t-2})$
- a stack of linear and angular jerks of the robot $(j_t, \zeta_t, j_{t-1}, \zeta_{t-1}, j_{t-2}, \zeta_{t-2})$
- x and y components of the robot's displacement from its target position $(d_{x,t}, d_{y,t})$

For their actions, both models output a linear and angular velocity for the robot/agent to perform.

3.2.3 Simulation Setup

To enable the DRL agent to learn from trial and error, we use a simple robot simulation environment called Stage. Stage provides a virtual world populated by mobile robots and sensors, along with various objects for the robots to sense and manipulate. We use Stage in ROS (Robot Operating System) to train our DRL agent [3]

3.2.4 Reward Function

The reward function, $r(s, a)$, serves as a training signal to encourage or discourage behaviours in the context of a desired task [10]. Thus, the features chosen for the reward function must capture the relevant objectives of the task. In this work, the following reward function was used for the *Normal Model*:

$$Reward, r_t = \begin{cases} -5 & \text{if robot crashes,} \\ 5 & \text{if robot reaches the goal,} \\ \gamma\Big((d_{t-1} - d_t)\Delta t - C\Big) & \text{otherwise.} \end{cases}$$

Where:

- γ acts as a discount factor (for infinite-horizon discounted return)
- d_{t-1} and d_t indicate the distance between the robot and its target at two consecutive time stamps
- Δt represents the time for each step
- C is a constant used as a time penalty.

Note that the policy learnt with this reward function may be *sub-optimal*. This is due to the fact that the agent is driven by something else other than reaching the target in the shortest time possible (i.e. the dense reward function with a discount factor γ). Ideally, a sparse reward function would be better as it would only reward the robot when it actually reaches the target; however, a sparse reward function is challenging for random exploration.

For the *Model with Jerk-Learning*, a new feature which penalizes the robot for jerky movements is added to the reward function. We have chosen to use a dense penalty function for the new feature as it would alleviate the challenges of learning the appropriate motion profile to minimize jerk:

$$Reward, r_t = \begin{cases} -5 & \text{if robot crashes,} \\ 5 & \text{if robot reaches the} \\ \gamma\Big((d_{t-1} - d_t)\Delta t - C\Big) - \gamma j_t - \frac{\gamma}{25}\left(\left(\frac{j_t}{\max j_t}\right)^2 + \left(\frac{\zeta_t}{\max \zeta t}\right)^2\right) & \text{otherwise.} \end{cases}$$

Where:

- j_t is linear jerk at time t
- $\max j_t$ is the theoretical maximum linear jerk for each state transition
- ζ_t is angular jerk at time t
- $\max \zeta t$ is the theoretical maximum angular jerk for each state transition

3.2.5 Network Architecture

The network architecture we use is largely similar to Xie et. al's AsDDPG network [9]. The difference, however, is that we have discarded the convolutional neural network (CNN) layers and used a sparser laser beam to allow the agent to learn the policy faster. We reiterate: the goal here is not to produce a state-of-the-art model for map-less navigation, but to empirically study a learning-based approach to jerk reduction.

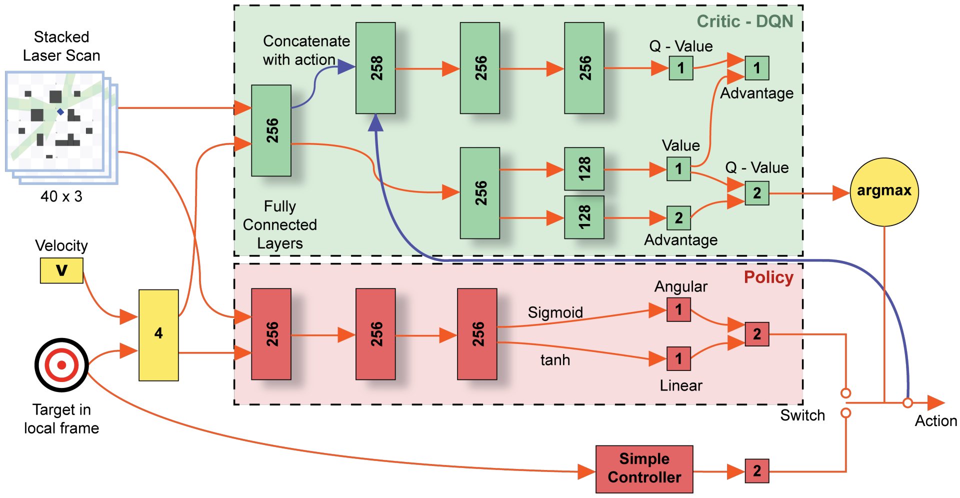


Figure 3.2: Network architecture. Network layers are demonstrated by rectangles. Orange arrows indicate the connectivity between network layers and some other components, e.g. input state and the output of the simple controller. Blue arrows indicate the operation of concatenation (i.e. tf.concat). The final action is selected based on the Q-value predicted by the critic-DQN.

The network from Figure 3.2 consists of two parts:

1. **Policy network and assistive controller (red)**: The policy network consists of a fully-connected layers which estimates the optimal linear and angular velocities for the robot based only on the input state (i.e., laser scans, current speed, and target position in the local frame). This is a very important feature of DDPG as sampling actions from a policy is computationally less expensive than exhaustively evaluating as many discrete actions as possible (since our action space is continuous). Note that the activation functions of the outputs of the policy are sigmoid (linear velocity) and tanh (angular velocity), respectively.

2. **Augmented critic network (green)**: The critic-DQN is also constructed with fully-connected layers. It has two branches: one is the critic branch where the action is concatenated into the second layer; the other is the DQN branch where (similar to Xie et. al [9]) we apply dueling and double network architecture to speed up training and avoid overestimation. Note that there is no nonlinear activation for its output layers.

3.2.6 Training Methodology

Our network from Figure 3.2 was trained in 3 different Stage simulation worlds A.1, A.2, A.3. In each simulation world, the network was trained until the average performance reaches a plateau (gradient of average performance with respect to time is roughly zero).

Here, average performance of the network is obtained according to Equation 3.2:

$$\text{Average Performance} = \mathop{\mathrm{E}}_{Ep\sim(100Eps)}\left[\mathop{\mathrm{E}}_{(s,a)\sim Ep}\Big(\max_{a\sim\pi} Q^{\pi}(s,a)\Big)\right] \tag{3.2}$$

3.3 Experiment Results

3.4 Discussion

Chapter 4

Like White on Rice

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

4.1 Preliminaries

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

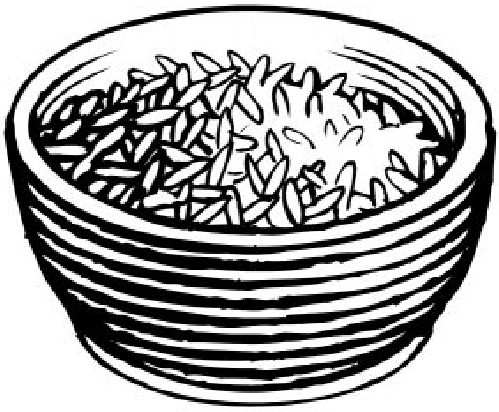


Figure 4.1: A bowl of rice.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

4.2 Methodology

Figure 4.1 shows a bowl of rice. Algorithm 1 demonstrates the formatting of pseudo code. Please carefully check the source files and learn how to use

Algorithm 1: Sample pseudo code of a dummy algorithm.

Global: max. calories of daily intake $\mathcal{C}$
Global: calories per bowl of rice $\mathcal{B}$

Input: number of bowls of rice n
Output: calories intake

Procedure `EatRice(`n`):`
- $cal \leftarrow n \times \mathcal{B}$
- **if** $cal \geq \mathcal{C}$ **then**
  - **return** $\mathcal{C}$
- **else**
  - **return** $cal -$ `DoExercise(`n`)`

Input: time duration (in minutes) of exercise t
Output: calories consumed

Function `DoExercise(`t`):`
- $cal \leftarrow 0$
- **for** $i \leftarrow 1$ **to** t **do** $cal \leftarrow cal + i$
- **return** cal

this style. Importantly:

- Always state your input.
- State the output if any.
- Always number your lines for quick referral.
- Always declare and initialize your local variables.
- Always use `\gets` (“←”) for assignments.

4.3 Evaluation

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam congue neque id dolor.

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

Figure 4.2 and Figure 4.3 show how the taste of rice is affected by meal repetition and freshness respectively. Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc ut metus. Ut metus justo, auctor

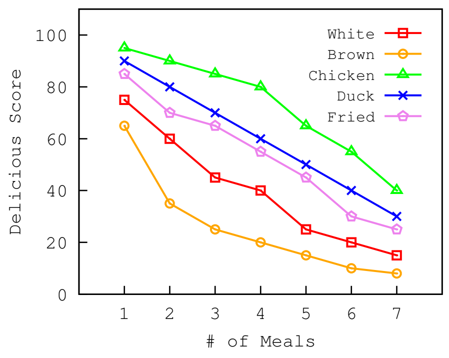


Figure 4.2: Taste with meal repetition.

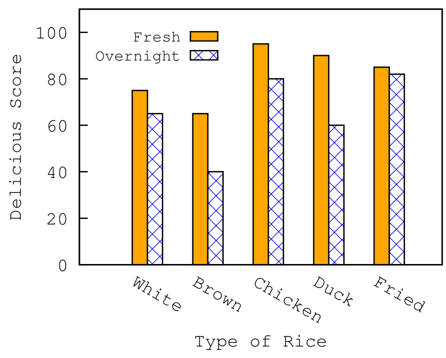


Figure 4.3: Taste with meal freshness.

at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

4.4 Summary

Etiam pede massa, dapibus vitae, rhoncus in, placerat posuere, odio. Vestibulum luctus commodo lacus. Morbi lacus dui, tempor sed, euismod eget, condimentum at, tortor. Phasellus aliquet odio ac lacus tempor faucibus. Praesent sed sem. Praesent iaculis. Cras rhoncus tellus sed justo ullamcorper sagittis. Donec quis orci. Sed ut tortor quis tellus euismod tincidunt. Suspendisse congue nisl eu elit. Aliquam tortor diam, tempus id, tristique eget, sodales vel, nulla. Praesent tellus mi, condimentum sed, viverra at, consectetuer quis, lectus. In auctor vehicula orci. Sed pede sapien, euismod in, suscipit in, pharetra placerat, metus. Vivamus commodo dui non odio. Donec et felis.

Chapter 5

Use Your Noodle

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

5.1 Preliminaries

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

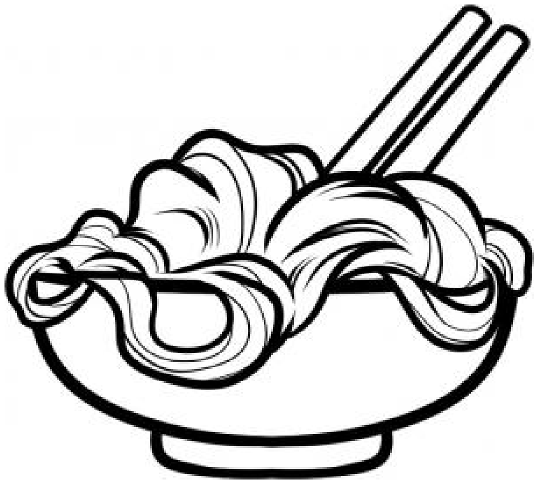


Figure 5.1: A bowl of noodles.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

5.2 Methodology

Figure 5.1 shows a bowl of noodles. Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetuer tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

5.3 Evaluation

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam congue neque id dolor.

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

5.4 Summary

Etiam pede massa, dapibus vitae, rhoncus in, placerat posuere, odio. Vestibulum luctus commodo lacus. Morbi lacus dui, tempor sed, euismod eget, condimentum at, tortor. Phasellus aliquet odio ac lacus tempor faucibus. Praesent sed sem. Praesent iaculis. Cras rhoncus tellus sed justo ullamcorper sagittis. Donec quis orci. Sed ut tortor quis tellus euismod tincidunt. Suspendisse congue nisl eu elit. Aliquam tortor diam, tempus id, tristique eget, sodales vel, nulla. Praesent tellus mi, condimentum sed, viverra at, consectetuer quis, lectus. In auctor vehicula orci. Sed pede sapien, euismod in, suscipit in, pharetra placerat, metus. Vivamus commodo dui non odio. Donec et felis.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

6.2 Future Research Directions

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

6.2.1 Eat good, Feel good, Look Good

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

6.2.2 An Apple a Day Keeps the Doctor Away

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

6.2.3 A Healthy Food for a Wealthy Mood

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

6.2.4 Eat Your Veggies, Have Less Wedgies

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

6.2.5 Salad and Beets Are Some Healthy Treats

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

6.2.6 Eat Only When Hungry

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.

We believe the above (but not limited to) future research directions will advance the technology presented in this thesis and contribute to academia and industry.

Bibliography

[1] J. Achiam, “Spinning Up in Deep Reinforcement Learning”,, 2018.

[2] G. E. Dahl, Dong Yu, Li Deng, and A. Acero, “Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition”, *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, Jan. 2012, ISSN: 1558-7916, 1558-7924. [Online]. Available: `http://ieeexplore.ieee.org/document/5740583/` (visited on 02/22/2020).

[3] B. P. Gerkey, R. T. Vaughan, and A. Howard, “The player/stage project: Tools for multi-robot and distributed sensor systems”,, p. 7,

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks”, *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 24, 2017, ISSN: 00010782. [Online]. Available: `http://dl.acm.org/citation.cfm?doid=3098997.3065386` (visited on 02/22/2020).

[5] P. Meckl and P. Arestides, “Optimized s-curve motion profiles for minimum residual vibration”, in *Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No.98CH36207)*, Philadelphia, PA, USA: IEEE, 1998, 2627–2631 vol.5, ISBN: 978-0-7803-4530-0. [Online]. Available: `http://ieeexplore.ieee.org/document/688324/` (visited on 01/06/2020).

[6] M. Pfeiffer, S. Shukla, M. Turchetta, C. Cadena, A. Krause, R. Siegwart, and J. Nieto, “Reinforced imitation: Sample efficient deep reinforcement learning for map-less navigation by leveraging prior demonstrations”, *arXiv:1805.07095 [cs]*, Aug. 31, 2018. arXiv: `1805.07095`. [Online]. Available: `http://arxiv.org/abs/1805.07095` (visited on 12/23/2019).

[7] L. Tai, G. Paolo, and M. Liu, "Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation", *arXiv:1703.00420 [cs]*, Jul. 21, 2017. arXiv: 1703.00420. [Online]. Available: http://arxiv.org/abs/1703.00420 (visited on 01/01/2020).

[8] G. Tesauro, "Temporal difference learning and TD-gammon",, p. 16,

[9] L. Xie, S. Wang, S. Rosa, A. Markham, and N. Trigoni, "Learning with training wheels: Speeding up training with a simple controller for deep reinforcement learning", in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD: IEEE, May 2018, pp. 6276–6283, ISBN: 978-1-5386-3081-5. [Online]. Available: https://ieeexplore.ieee.org/document/8461203/ (visited on 12/30/2019).

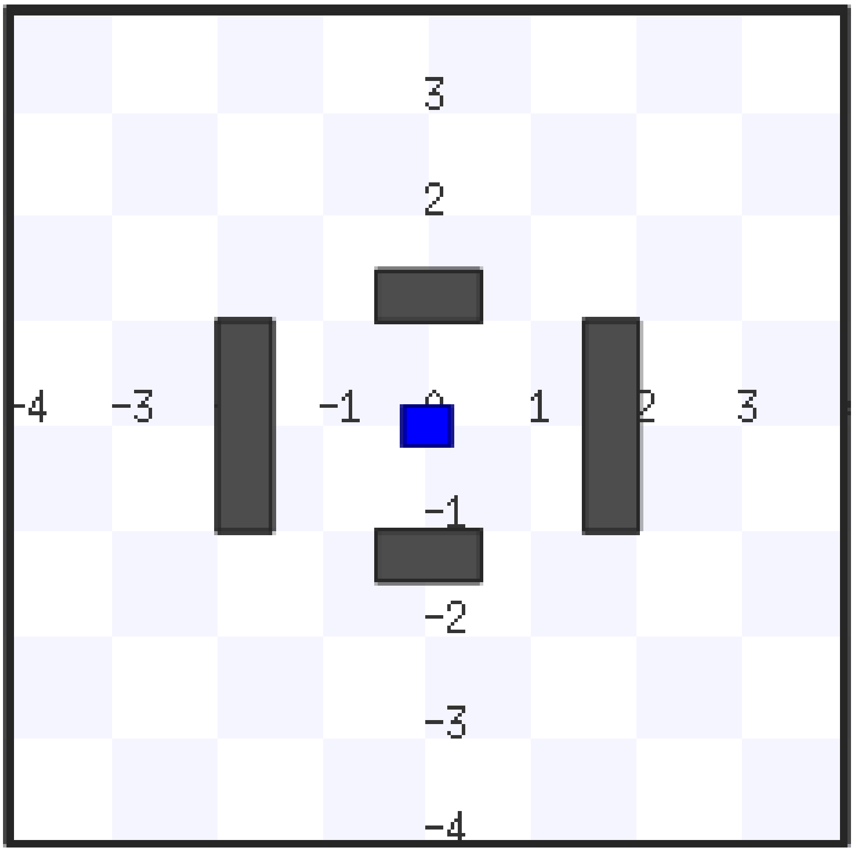
[10] M. Zhu, Y. Wang, Z. Pu, J. Hu, X. Wang, and R. Ke, "Safe, efficient, and comfortable velocity control based on reinforcement learning for autonomous driving", *arXiv:1902.00089 [cs, stat]*, Oct. 31, 2019. arXiv: 1902.00089. [Online]. Available: http://arxiv.org/abs/1902.00089 (visited on 12/23/2019).

Appendix A

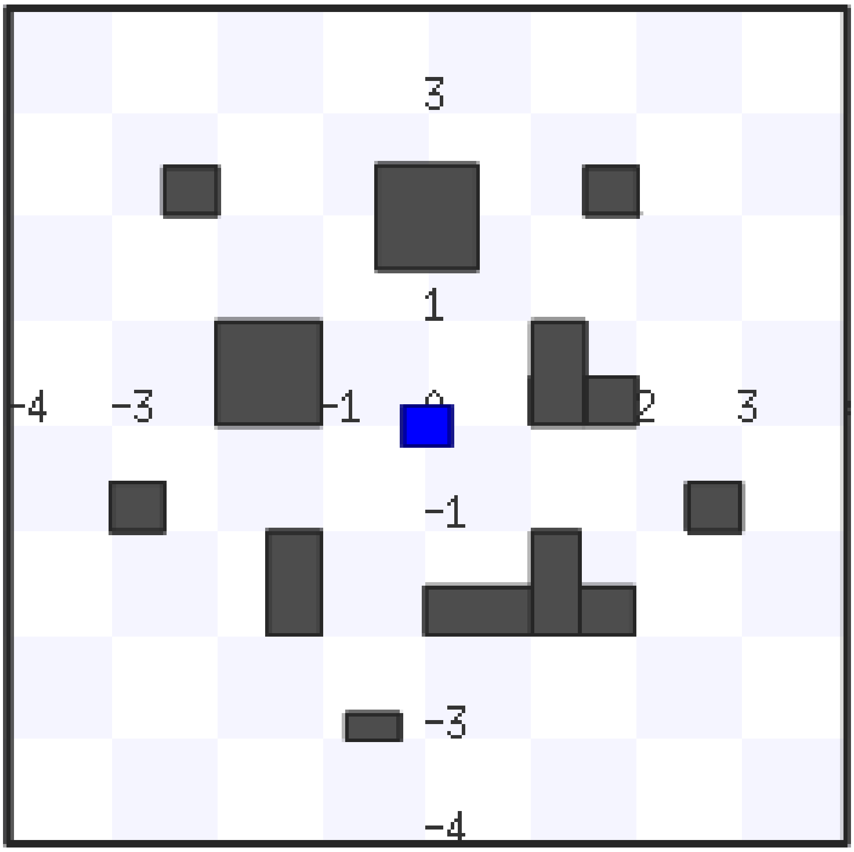
Stage Simulation Worlds

A.1 Blank World

A.2 Simple World



A.3 Complex World



Appendix B

Recipes

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

B.1 Mustard Green Pork Rice

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

B.2 Hakka Pancake Roll

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

B.3 Pumpkin Tang Yuan

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Appendix C

Ingredients

C.1 Hokkien Mee: Stir-fry Noodles with Shrimps and Squids

Table C.1: Table to test captions and labels

Col1	Col2	Col2	Col3
1	6	87837	787
2	7	78	5415
3	545	778	7507
4	545	18744	7560
5	88	788	6344

Maecenas dui. Aliquam volutpat auctor lorem. Cras placerat est vitae lectus. Curabitur massa lectus, rutrum euismod, dignissim ut, dapibus a, odio. Ut eros erat, vulputate ut, interdum non, porta eu, erat. Cras fermentum, felis in porta congue, velit leo facilisis odio, vitae consectetuer lorem quam vitae orci. Sed ultrices, pede eu placerat auctor, ante ligula rutrum tellus, vel posuere nibh lacus nec nibh. Maecenas laoreet dolor at enim. Donec molestie dolor nec metus. Vestibulum libero. Sed quis erat. Sed tristique. Duis pede leo, fermentum quis, consectetuer eget, vulputate sit amet, erat.

C.2 Nasi Lemak: Rice Boiled with Coconut Milk and Accompanied with Chilli Paste and Anchovies

Donec vitae velit. Suspendisse porta fermentum mauris. Ut vel nunc non mauris pharetra varius. Duis consequat libero quis urna. Maecenas at ante.

Vivamus varius, wisi sed egestas tristique, odio wisi luctus nulla, lobortis dictum dolor ligula in lacus. Vivamus aliquam, urna sed interdum porttitor, metus orci interdum odio, sit amet euismod lectus felis et leo. Praesent ac wisi. Nam suscipit vestibulum sem. Praesent eu ipsum vitae pede cursus venenatis. Duis sed odio. Vestibulum eleifend. Nulla ut massa. Proin rutrum mattis sapien. Curabitur dictum gravida ante.

C.3 Prata: a Dough Served with Curry

Phasellus placerat vulputate quam. Maecenas at tellus. Pellentesque neque diam, dignissim ac, venenatis vitae, consequat ut, lacus. Nam nibh. Vestibulum fringilla arcu mollis arcu. Sed et turpis. Donec sem tellus, volutpat et, varius eu, commodo sed, lectus. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Quisque enim arcu, suscipit nec, tempus at, imperdiet vel, metus. Morbi volutpat purus at erat. Donec dignissim, sem id semper tempus, nibh massa eleifend turpis, sed pellentesque wisi purus sed libero. Nullam lobortis tortor vel risus. Pellentesque consequat nulla eu tellus. Donec velit. Aliquam fermentum, wisi ac rhoncus iaculis, tellus nunc malesuada orci, quis volutpat dui magna id mi. Nunc vel ante. Duis vitae lacus. Cras nec ipsum.

C.4 Fried Carrot Cake: Stir-fry Rice Cakes with Eggs

Morbi nunc. Aliquam consectetuer varius nulla. Phasellus eros. Cras dapibus porttitor risus. Maecenas ultrices mi sed diam. Praesent gravida velit at elit vehicula porttitor. Phasellus nisl mi, sagittis ac, pulvinar id, gravida sit amet, erat. Vestibulum est. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Curabitur id sem elementum leo rutrum hendrerit. Ut at mi. Donec tincidunt faucibus massa. Sed turpis quam, sollicitudin a, hendrerit eget, pretium ut, nisl. Duis hendrerit ligula. Nunc pulvinar congue urna.