# Final Year Project Report

**Full Unit - Interim Report**

# Resourceful Robots

Cougar Tasker

A report submitted in part fulfilment of the degree of

**MSci (Hons) in Computer Science (Artificial Intelligence)**

**Supervisor:** Dr. Anand Subramoney

Department of Computer Science
Royal Holloway, University of London

November 6, 2023

# Declaration

This report has been prepared on the basis of my own work. Where other published and unpublished source materials have been used, these have been acknowledged.

Word Count:

Student Name: Cougar Tasker

Date of Submission:

Signature:

# Table of Contents

# Abstract

This document serves as a layout and formatting template for your project report. It does not tell you how to write it, or what it should contain. It explains how it should be formatted and typeset. Please refer to your project booklet for information about report sizes, contents and rules.

***NOTE: in your report, you should replace this with an appropriate Abstract for your project report.***

# Project Specification

Your project specification goes here.

# Chapter 1: **Introduction**

The project report is a very important part of your project and its preparation and presentation should be of extremely high quality. Remember that a significant portion of the marks for your project are awarded for this report.

The format of the final report is fixed by the template of this document and the Department of Computer Science suggests its usage.

While this may sound like a rather prescriptive approach to report writing, it is introduced for the following reasons:

1. The template allows students to focus on the critical task of producing clear and concise content, instead of being distracted by font settings and paragraph spacing.
2. By providing a comprehensive template the Department benefits from a consistent and professional look to its internal project reports.

The remainder of this document briefly outlines the main components and their usage.

A **final project report** is approximately 15,000 words and must include a word count. It is acceptable to have other material in appendixes. Your **interim report** for the December Review meeting, even if it is a collection of reports, should have a total word count of about 5,000 words. This should summarise the work you have done so far, with sections on the theory you have learnt and the code that you have written.

Also remember that any details of report content and submission rules, as well as other deliverables, are defined in the project booklet

## 1.1 How to use this template

The simplest way to get started with your report is to save a copy of this document. First change the values for the initial document definitions such as `studentname` and `reportyear` to match your details. Delete the unneeded sections and start adding your own sections using the styles provided. Before submission, remember to fill in the Declaration section fields.

# Chapter 2: **Markov Decision Processes**

Markov Decision Processes (MDP) provide a mathematical formalisation of decision-making problems. Markov Decision Processes provide the foundation for reinforcement learning (RL). This is because MDPs distil the fundamental parts of decision-making, allowing RL techniques built upon MDPs to generalise to learning in the real world and across different domains such as finance and robotics.

As a formal mathematical framework, MDPs allow us to derive and prove statements about our RL methods built upon them. An important example of this is that we can prove that Q-learning (an RL technique explained in chapter 4) will converge to the true Q-values as long as each Action-State pair is visited infinitely often. [1]. Furthermore, MDPs allow us to reason about problems with uncertainty allowing RL agents to account for randomness in their environment.

The standardisation of decision-making problems as MDPs allows for a uniform definition of optimality with the value functions. MDPs give a basis for assessing the performance of RL algorithms, facilitating like-for-like comparisons for different RL approaches.

## 2.1 Markov Property

The Markov property is that the future state of a Markov system only depends on the current state of the system. In other words, if we have a system that follows the Markov property, then the history preceding the current configuration of the system will not influence the following state.

To put the Markov property formally $S_t$ represents the state at some time $t$. $S_t$ represents the outcome of some random variable. Then the Markov property would hold if and only if:

$$\Pr(S_{c+1} \mid S_c, S_{c-1}, \dots, S_0) = \Pr(S_{c+1} \mid S_c)$$

This definition demonstrates how the Markov property can hold in non-deterministic, stochastic processes. It also shows that predictions that are only based on the current state are just as good as those that record the history in a Markov process. The sequence of events in this definition, $S_t$, is called a Markov Chain[2].

## 2.2 Extending Markov Chains

Markov Decision Processes extend Markov Chain's in two important ways. Firstly MDPs introduce decision-making through actions. Each state in an MDP has a set of available actions in that state. In each state, an action is required to transition to the next state; this action with the current state can affect what the following state will be. Secondly, MDPs introduce a reward value. The reward is determined from the current state and action; it is produced simultaneously with the following state.

A formal definition of a Markov Decision Process is a tuple $(\mathcal{S}, \mathcal{A}_s, p)$ where:

- $\mathcal{S}$ defines the set of all states
- $\mathcal{A}_s$ defines the set of available actions in state $s$
- $p$ defines the relationship between states, actions and rewards:
  $p(s', r \mid s, a) \doteq \Pr(S_{t+1} = s', R_{t+1} = r | S_t = s, A_t = a)$[3]
  - $s, s' \in \mathcal{S}$, $a \in \mathcal{A}_s$ and $r \in \mathbb{R}$
  - $p : \mathcal{S} \times \mathbb{R} \times \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$

The function $p$ is an integral part of this definition; it fully describes how the system will evolve. We call this function the dynamics of the MDP. What this definition does not describe is how actions are chosen. This decision-making is done by an entity called an agent. For our purposes, the agent will have complete visibility as to the current state of the MDP. However, like most real-world situations, our agent will not have any a priori knowledge of the dynamics.

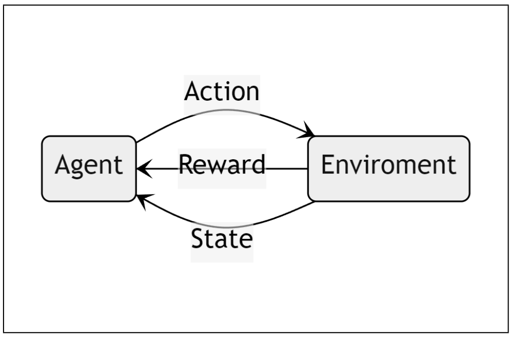


Figure 2.1: The agent-environment interface

The agent comprises the entire decision-making entity in an MDP; anything unknown or not wholly under the agent's control is called the agent's environment. In the context of reinforcement learning, the environment is essentially the dynamics of the MDP. Figure 2.1 demonstrates how the agent and environments affect each other in an MDP.

For learning agents, we wish to improve the agent's behaviour over time. For this purpose, we introduce a policy $\pi$. This policy defines the action chosen by an agent under a particular state. The policy can be represented with a lookup table like in Q-learning4 or a more complex process such as deep Q-learning. A Policy like this is not hard-coded, allowing the agent to update the policy based on the information the agent learns from the environment.

# Chapter 3: **Policy and Value functions**

At each time step a reward is received, it follows that the goal of an agent should be to maximise these reward signals received. following from the Markov principle and the definition of a MDP this reward only depends on the current state and the action chosen. The consequence is that being in certain states and performing certain actions are more valuable to the agent than others. We can define value functions:

- $v(s)$ function determines the value of being in a given state
- $q(s,a)$ function determines the value of being in a given state and performing a specific action

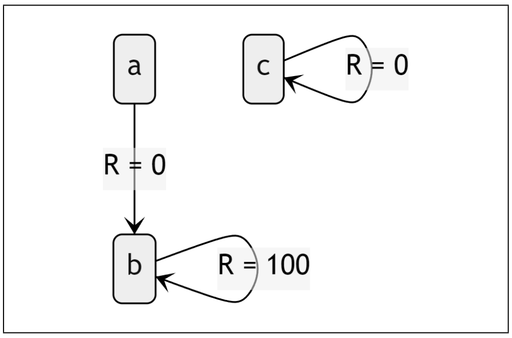


Figure 3.1: An example of transitive value

The value of being in a state is more than just its immediate reward that might be found from performing actions in that state, it is also related to the potential future reward that might be achieved in the reachable subsequent states. It can be thought about with two states $a$, and $b$ if there is a large reward at $b$ but the only way to reach $b$ is through $a$ then being at $a$ is also valuable regardless of the reward available at $a$. However if like figure 3.1 despite the potential reward at $a$ being the same as $b$ it requires more actions. In this sense, $a$ is less valuable than state $b$. To account for this we want to discount future value with a parameter $\gamma$.

— give formal defition for policy

These value functions go hand in hand with the policy of an agent, A good policy maximises being in valuable states and performing valuable actions. On the other side of the coin, the value is determined by the subsequent states and rewards which are determined by the actions selected and the dynamics of the MDP. In on-policy learning the policy and value functions can compound on each other making the effect of the discount rate ($\gamma$) especially important. With high discount rates $\gamma \approx 1$ the agent can be far-sighted and ignore short-term high-reward actions available to it and take longer to learn. With low discount rates $\gamma \approx 0$, the agent can be short-sighted ignoring the potential long-term benefits of certain actions.

## 3.1 optimal policy/value function via the Bellman equation

## 3.2 Finding optimal policies by iteration

### 3.2.1 Value iteration

### 3.2.2 Policy iteration

# Chapter 4: **Q-learning**

# Chapter 5: **Page Layout & Size**

The page size and margins have been set in this document. These should not be changed or adjusted.

In addition, page headers and footers have been included. They will be automatically filled in, so do not attempt to change their contents.

# Chapter 6: **Headings**

Your report will be structured as a collection of numbered sections at different levels of detail. For example, the heading to this section is a first-level heading and has been defined with a particular set of font and spacing characteristics. At the start of a new section, you need to select the appropriate LaTeX command, `\chapter` in this case.

## 6.1 Second Level Headings

Second level headings, like this one, are created by using the command `\section`.

### 6.1.1 Third Level Headings

The heading for this subsection is a third level heading, which is obtained by using command `\subsection`. In general, it is unlikely that fourth of fifth level headings will be required in your final report. Indeed it is more likely that if you do find yourself needing them, then your document structure is probably not ideal. So, try to stick to these three levels.

## 6.2 A Word on Numbering

You will notice that the main section headings in this document are all numbered in a hierarchical fashion. You don't have to worry about the numbering. It is all automatic as it has been built into the heading styles. Each time you create a new heading by selecting the appropriate style, the correct number will be assigned.

# Chapter 7: **Presentation Issues**

## 7.1 Figures, Charts and Tables

Most final reports will contain a mixture of figures and charts along with the main body of text. The figure caption should appear directly after the figure as seen in Figure 7.1 whereas a table caption should appear directly above the table. Figures, charts and tables should always be centered horizontally.

Figure 7.1: Logo of RHUL.

## 7.2 Source Code

If you wish to print a short excerpt of your source code, ensure that you are using a fixed-width sans-serif font such as the Courier font. By using the `verbatim` environment your code will be properly indented and will appear as follows:

```
static public void main(String[] args) {
  try  {
    UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
  }
  catch(Exception e) {
    e.printStackTrace();
  }
  new WelcomeApp();
}
```

# Chapter 8: **References**

Use one consistent system for citing works in the body of your report. Several such systems are in common use in textbooks and in conference and journal papers. Ensure that any works you cite are listed in the references section, and vice versa.

# Chapter 9: **Project Information and Rules**

The details about how your project will be assessed, as well as the rules you must follow for this final project report, are detailed in the project booklet.

**You must read that document and strictly follow it.**

# Bibliography

[1] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, pp. 279–292, 1992.

[2] S. P. Meyn and R. L. Tweedie, *Markov chains and stochastic stability.* Springer Science & Business Media, 2012.

[3] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* MIT press, 2018.