



**FACULTY
OF MATHEMATICS
AND PHYSICS**
Charles University

MASTER THESIS

Name Surname

Thesis title

Name of the department

Supervisor of the master thesis: Supervisor's Name

Study programme: study programme

Study branch: study branch

Prague YEAR

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In date
Author's signature

Dedication.

Title: Thesis title

Author: Name Surname

Department: Name of the department

Supervisor: Supervisor's Name, department

Abstract: Abstract.

Keywords: key words

Contents

Introduction	2
1 Gentle summarization of Reinforcement learning	3
1.1 Defining mathematical common ground for environment	3
2 Policy gradient methods	4
2.1 Idea, motivation and brief technical description of algorithm . . .	4
2.2 Variants of policy theorem	4
3 Multi agent environments for RL	5
3.1 Definitions	5
3.2 Possible mention of MAPPO success	5
3.3 Zero sum vs cooperation	5
4 Overcooked environment	6
4.1 Overcooked game	6
4.2 Basic layouts	6
4.3 Problem of robustness	7
4.3.1 Human cooperation	7
5 Lack of robustness problem of AI agents	8
5.1 Related work	8
6 Our contribution	9
6.1 Utilized framework	9
6.2 Robustness metric	9
6.3 Population build up	9
6.4 Population policies difference rewards augmentation	9
6.5 Population policies difference loss	9
Conclusion	10
Bibliography	11
List of Figures	12
List of Tables	13
List of Abbreviations	14
A Attachments	15
A.1 First Attachment	15

Introduction

1. Gentle summarization of Reinforcement learning

An example citation: Anděl [2007]

1.1 Defining mathematical common ground for environment

2. Policy gradient methods

2.1 Idea, motivation and brief technical description of algorithm

2.2 Variants of policy theorem

Vanilla

PPO

3. Multi agent environments for RL

3.1 Definitions

3.2 Possible mention of MAPPO success

3.3 Zero sum vs cooperation

4. Overcooked environment

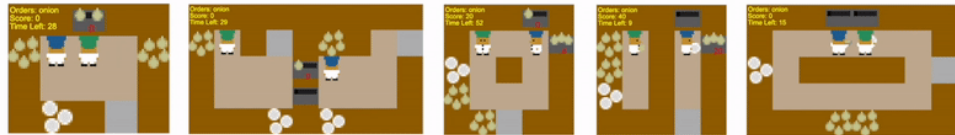
4.1 Overcooked game

Before we get into our problems with cooperation let us first examine the environment. We will be working with environment based on popular cooking video game <https://ghosttowntgames.com/overcooked/>. Overcooked is multiplayer cooperative game where the goal is to work in a kitchen as a team with partner cooks and prepare together various dishes within limited time. However, the game is dynamic to a great extent. In many maps the kitchen itself is not static and may be changing on a run. Moreover, random events such as pots catching fire add to the chaos. The challenge lies in coordination with rest of the team and dividing subtasks efficiently.

The aforementioned game was simplified and reimplemented to simpler environment https://github.com/HumanCompatibleAI/overcooked_ai to serve a purpose of scientific common ground for studying multi agent cooperation in somewhat complex settings. Lot of additional features of original game were removed and remained only essential coordination aspects. In its simplest form, environment is taking place in small static kitchen layout where only available recipe is onion soup which can be prepared by putting three onions in a pot and waiting for given time period. Somewhere in the kitchen there is unlimited source of onions and dish dispenser, where player can grab a dish to carry cooked onion soup in to the counter. Team of cooks is rewarded as team by abstract reward of value 20 every time cooked soup is delivered to the counter. It may seem that the task is quite straightforward. However, players face problems on multiple levels.

4.2 Basic layouts

Although the Overcooked implementation has its own generator that can be used to generate new random kitchen layouts, the majority of the related scientific work has so far experimented with a fixed set of predefined layouts, where each of them capture some important aspect of coordination.



(From left to right: Cramped room, Assymmetric advantages, Coordination ring, Forced coordination, Counter circuit)

Cramped room as a name suggests represents cramped kitchen layout where all important places are relatively easy to reach. Challenge lies in low level coordination of movement with the other partner as there is no spare room.

In Assymmetric advantages both players are located in separated regions where each region is fully self-sustaining. However, each region has better potential for specific subtask. And it is only when both players make the most of their own region's potential that the maximal shared efficacy is reached.

The Coordination ring is another example of a layout where clever coordination is required as the only possible movement around the kitchen is along a narrow circular path that can be used in a given direction. For example, if one player decides to move in clockwise direction, the other player would automatically get stuck if pursuing counter-clockwise movement.

Forced coordination kitchen layout is significantly different from others. In this layout, each player is located in a separate region where neither player has all the resources necessary to prepare a complete onion soup. Thus, players are forced to cooperate with each other with the resources they have.

In the last layout, the situation may look similar to the coordination ring. However, in this case, carrying onions around the entire kitchen is highly suboptimal no matter which direction the players choose. To deliver onions efficiently, players must pass them over the counter to shorten the distance. However, the cooks still need to decide who will be responsible for bringing the plates.

4.3 Problem of robustness

4.3.1 Human cooperation

Most of the previous work in this area has focused on one of two types of coordination. The first being coordination between a human and an AI partner. And second, focusing solely on the fully AI-driven pair.

While perfect AI-human coordination is generally a more desirable goal to achieve in all sorts of domains, it will not be our main focus. Several previous scientific papers have addressed this issue. A particularly noteworthy contribution is the article On the Utility of Learning about Humans for Human-AI Coordination <https://arxiv.org/abs/1910.05789>, whose authors are also responsible for creating the overcooked environment implementation. They collected many human-human episodes and incorporated these experiences in the form of human behavior clone models into the training. Their main conclusions were that AI models often rely heavily on the optimal behavior of their partner. However, when such a model is paired with generally suboptimal human behavior, it often fails to cooperate at all, regardless of the approach used for training of the AI model.

One of the other important conclusions they came to was that even AI-AI coordination often fails when models are paired with another AI model trained using a different approach. We will discuss some of these popular approaches in the next section.

Definition of robustness problem

Ad hoc agent playing? Trivial states failure?

5. Lack of robustness problem of AI agents

5.1 Related work

Common approaches fail when paired with foreign

6. Our contribution

6.1 Utilized framework

Comparision of rllib and StableBaselines3

Rllib framework was used in original paper, however for our usages stable baselines seemed sufficient and reasonably easy to extend. Stable baselines has no explicit support of multi-agent environments.

Essential modifications of stable baselines

By default SB3 comes with various wrappers to support most of common environment settings, including

Essential modifications of stable baselines

Initializing methods that resets environment did not yield correct initial positions for some maps.

6.2 Robustness metric

Defining our own metric for robust cooperation of two sets of agents. Probably just average of pair results (non diagonal in case of same sets). Maybe number of pairs who surpassed some threshold?

6.3 Population build up

6.4 Population policies difference rewards augmentation

6.5 Population policies difference loss

Conclusion

Bibliography

J. Anděl. *Základy matematické statistiky*. Druhé opravené vydání. Matfyzpress, Praha, 2007. ISBN 80-7378-001-1.

List of Figures

List of Tables

List of Abbreviations

A. Attachments

A.1 First Attachment