Routing Choices in Intelligent Transport Systems

by

Charlotte Daisy Roman

Thesis

Submitted to the University of Warwick

for the degree of

Doctor of Philosophy

Mathematics of Systems

Contents

Acknowledgments iv

Declarations v

Abstract vi

Abbreviations vii

Symbols ix

Chapter 1 Introduction 1

1.1 Motivation . . . 1
1.2 Research Questions . . . 4
1.3 Publications . . . 6
1.4 Thesis Outline . . . 7

Chapter 2 Preliminaries 9

2.1 Basic Game Definitions . . . 10
2.2 Social Dilemmas . . . 11
2.3 Congestion Games . . . 12
2.3.1 Immunity to Braess' Paradox . . . 16
2.3.2 Information Constrained Nonatomic Congestion Games . . . 18
2.4 Nonlinear Optimisation . . . 20
2.5 Reinforcement Learning . . . 21
2.5.1 Multi-Agent Reinforcement Learning . . . 23

Chapter 3 Safe Cooperation in Social Dilemmas 29

3.1 Introduction . . . 29
3.2 Contributions . . . 30
3.3 Literature Review . . . 31
3.4 Motivating Example . . . 32
3.5 Safe Beliefs . . . 34

3.6 Cooperation Inducing Beliefs . . . 37
3.7 Trade-Off Between Cooperation and Safety . . . 40
3.8 Investing in Cooperation . . . 44
3.8.1 Matrix Games . . . 46
3.9 Multi-Agent Reinforcement Learning . . . 53
3.9.1 Prisoner's Dilemma . . . 54
3.9.2 Stag Hunt . . . 56
3.9.3 Route Choice . . . 57
3.10 Discussion . . . 58

Chapter 4 Network Control **61**
4.1 Introduction . . . 61
4.2 Contributions . . . 62
4.3 Literature Review . . . 62
4.4 Network Control Games . . . 65
4.5 Inefficiency of Route Controllers . . . 68
4.6 Multi-Agent Learning Example . . . 76
4.7 Choosing Route Planners . . . 78
4.8 Discussion . . . 81

Chapter 5 Information in Nonatomic Congestion Games **83**
5.1 Introduction . . . 83
5.2 Contributions . . . 84
5.3 Literature Review . . . 84
5.4 Known Results on Immunity to IBP . . . 86
5.5 Matroid Games and IBP . . . 87
5.6 Circuit Games and IBP . . . 90
5.7 IBP for Social Cost . . . 105
5.8 Discussion . . . 107

Chapter 6 Nonatomic Congestion Games with Traffic Lights **109**
6.1 Introduction . . . 109
6.2 Contributions . . . 110
6.3 Literature Review . . . 111
6.4 Traffic Light Game . . . 113
6.5 Braess' Paradox and Traffic Lights . . . 118
6.6 Price of Anarchy . . . 126
6.7 Adaptive Traffic Lights . . . 130

6.8 Biased Adaptive Traffic Lights 133
6.8.1 Traffic Lights in SUMO 136
6.8.2 Simulation Results 138
6.8.3 Fairness of Adaptive Traffic Lights 140
6.8.4 Fairness of Reward Functions 143
6.9 Discussion 148

Chapter 7 Conclusions **150**

Appendix A Reinforcement Learning Traffic Lights in SUMO **154**

Appendix B Multi-agent Reinforcement Learning Traffic Lights Simulation **160**

Acknowledgments

I am deeply grateful for my supervisor Paolo for his invaluable advice and continual support throughout my PhD. A huge thanks to Michael Dennis for mentoring in my CHAI research collaboration. I would also like to thank my Mum for providing her excellent proof-reading skills whenever I needed them and my partner Joe for his patience at being my 'rubber duck' on countless occasions. Many thanks to my co-authors Long Tran-Thanh, Andrew Critch, and Stuart Russell. Finally, I would like to extend my thanks to everyone in Mathsys, friends, and family.

Declarations

This work has been composed by myself and has not been submitted for any other degree or professional qualification.

- Chapters 4, 5, and 6 present theory for congestion games, and Chapters 3 and 6 present theory for multi-agent reinforcement learning.
- Part of the work in Chapter 5 has been published in IJCAI 2019 conference proceedings.
- Work from Chapter 3 has been published in the conference proceedings for AAMAS 2021.
- Work from Chapter 4 was presented at the IJCAI 2021 workshop on Reinforcement Learning for Intelligent Transport Systems.

Abstract

Road congestion is a phenomenon that can often be avoided; roads become popular, travel times increase, which could be mitigated with better coordination mechanisms. The choice of route, mode of transport, and departure time all play a crucial part in controlling congestion levels. Technology, such as navigation applications, have the ability to influence these decisions and play an essential role in congestion reduction. To predict vehicles' routing behaviours, we model the system as a game with rational players. Players choose a path between origin and destination nodes in a network. Each player seeks to minimise their own journey time, often leading to inefficient equilibria with poor social welfare. Traffic congestion motivates the results in this thesis. However, the results also hold true for many other applications where congestion occurs, e.g. power grid demand.

Coordinating route selection to reduce congestion constitutes a social dilemma for vehicles. In sequential social dilemmas, players' strategies need to balance their vulnerability to exploitation from their opponents and to learn to cooperate to achieve maximal payoffs. We address this trade-off between mathematical safety and cooperation of strategies in social dilemmas to motivate our proposed algorithm, a safe method of achieving cooperation in social dilemmas, including route choice games.

Many vehicles use navigation applications to help plan their journeys, but these provide only partial information about the routes available to them. We find a class of networks for which route information distribution cannot harm the receiver's expected travel times. Additionally, we consider a game where players always follow the route chosen by an application or where vehicle route selection is controlled by a route planner, such as autonomous vehicles. We show that having multiple route planners controlling vehicle routing leads to inefficient equilibria. We calculate the Price of Anarchy (PoA) for polynomial function travel times and show that multi-agent reinforcement learning algorithms suffer from the predicted Price of Anarchy when controlling vehicle routing.

Finally, we equip congestion games with waiting times at junctions to model the properties of traffic lights at intersections. Here, we show that Braess' paradox can be avoided by implementing traffic light cycles and establish the PoA for realistic waiting times. By employing intelligent traffic lights that use myopic learning, such as multi-agent reinforcement learning, we prove a natural reward function guarantees convergence to equilibrium. Moreover, we highlight the impact of multi-agent reinforcement learning traffic lights on the fairness of journey times to vehicles.

Abbreviations

A3C: Asynchronous Advantage Actor-Critic (algorithm)

Adv: Adversarial agent

AI: Artificial Intelligence

ARCTIC: Accumulating Risk Capital Through Investing in Cooperation (algorithm)

BP: Braess' Paradox

BR: Best Response

Coord Q: Coordinated Q-Learning (algorithm)

DQN: Deep Q Network (algorithm)

DUE: Dynamic User Equilibrium

IBP: Informational Braess' Paradox

IBPSC: Informational Braess' Paradox for Social Cost

ICUE: Information Constrained User Equilibrium

Ind Q: Independent Q-Learning (algorithm)

Ind Q: Independent Deep Q Network (algorithm)

GPS: Global Positioning System

LI: Linearly Independent

LSTM: Long Short Term Memory (neural network architecture)

MARL: Multi-Agent Reinforcement Learning

MDP: Markov Decision Process

Multi Q: Multi-Agent Q-Learning (algorithm)

Multi DQN: Multi-Agent Deep Q Network (algorithm)

NE: Nash Equilibrium

OD: Origin-Destination (pair)

PoA: Price of Anarchy

RL: Reinforcement Learning

RL-CD: Reinforcement Learning with Context Detection

SC: Social Cost

SLI: Series Linearly Independent

SSD: Sequential Social Dilemma

SO: Social Optimum

SUMO: Simulation of Urban Mobility (software)

SUE: Stochastic User Equilibrium

T4T: Tit-For-Tat (strategy)

TL: Traffic Light

TLUE: Traffic Light User Equilibrium

UE: User Equilibrium

UK: United Kingdom

VE: Variable Elimination

Symbols

α: learning rate, or cooperation in Chapter 3

β: opponent cooperation (Chapter 3)

γ: discount factor

ϵ: exploration rate, or safety level in Chapter 3

θ: Neural network parameters

κ: the set of all possible irredundant information types in a congestion game

λ: Lagrange multipliers

π: policy

σ_i: strategy of player i

Σ_i: mixed strategy space of player i

Φ: potential function

a: action

c_e: cost functions of edge e

C_i: cost function of population i

C_r: cost function of a route planner r

$\boldsymbol{\mathcal{C}}$: circuit

d_i: demand of population i

d_i^r: demand of a population i controlled by a route planner r

D_i: destination of population i

D_{κ_r}: strategy space for route planners in the network control game

E: edge set, resource set

G: a network

$\mathcal{I}$: set of independent sets

K_i: set of information types of population i

M: matroid

N: set of players or populations

NE: set of Nash equilibria

O_i: origin node of population i

P_0: responsive control policy

p: traffic light parameter or polynomial degree in Chapter 4

p_i: belief of player i

p^A: adversarial policy-conditioned belief

p^C: cooperation inducing policy-conditioned belief

Q: quality function, or state-action value

Q_j: quality function for edge j in a coordination graph

R: set of route planners

R, P, S, T: payoffs for a two-player matrix game social dilemma in Chapter 3

t_g: length of green light phase of traffic light cycle

t_r: length of red light phase of traffic light cycle

T: total simulation time

V: node set

$V(\pi)$: expected value function

s: state

S_i: strategy set of population i

SC: social cost function

x: level of cooperation in belief p^C

$\boldsymbol{x}$: feasible strategy distribution

x_i^s: amount of players in population i choosing strategy s

$\mathbb{1}$: indicator function

CHAPTER 1

Introduction

1.1 Motivation

Congestion is one of the most significant issues in developed cities across the world. Road traffic is considered a major threat to clean air due to the release of harmful gases that can affect people's health and contribute to global warming. The benefits of reducing congestion include saving time, reducing economic costs, lowering greenhouse emissions, and cleaner air. The 2019 INRIX Global Traffic Scorecard reports calculated that congestion cost the UK economy £6.9 billion in 2019. On average, UK road users lost 115 hours and £894 a year due to congestion. London was the most congested UK city with drivers losing an average of 149 hours and is the 8^{th} most congested city of the 975 cities in the study.

Many methods have been implemented to combat urban congestion and pollution. The UK government has supported the move towards electric vehicles by banning the sale of new petrol and diesel cars by 2030. In London, congestion tolls have been introduced to create 'Ultra Low Emission Zones'. This combination of approaches to minimise road usage can be described as an intelligent transport system.

An intelligent transport system is an interconnected system of technologies applied to increase safety and reduce congestion in traffic networks. The primary goals of intelligent transport systems are traffic management, data collection, and data analysis. For example, this spans smart motorways, navigation planners, traffic forecasting, and variable speed limits. A recent study showed to a large extent that intelligent transport systems have a greater effect on improving congestion than building new roads Cheng *et al.* (2020). Harnessing the power of existing technology to improve congestion is also more cost-effective than introducing new infrastructure.

In particular, congestion can arise due to a lack of coordination between vehicles. The most popular fast routes, such as motorways, become congested since

each vehicle makes their route choice from their individual perspective, rather than considering how their choices affect others. These instances are referred to as social dilemmas; the dilemma is whether to behave cooperatively or selfishly. Deciding on a route to use, choosing the mode of transport, or planning a departure time can all create social dilemmas associated with congestion. As the faster routes options become popular, they no longer give the best journey times.

To achieve high utility when social dilemmas are played sequentially, strategies should be able to reciprocate their opponent's cooperative behaviour and punish their defections. Another mathematically desirable property is safety. A safe strategy maximises the minimum reward and cannot be exploited. Throughout this thesis, we will use the term safety to refer to this mathematical property. The properties of safety and cooperation are necessarily in tension here. However, achieving them both could incentivise better coordination of routes, thus, reducing congestion levels.

With many drivers choosing to use GPS navigation, intelligent transport systems have the potential improve congestion significantly. The algorithms used in these route-planning software could have a massive impact on congestion. As autonomous vehicles become increasingly popular, so should research to improve automatic route planners to choose more socially preferable routing. Otherwise, congestion will continue to cause a multitude of problems in urban areas.

Traffic lights also play an important role in inner-city congestion. In a town or city, traffic lights at intersections could optimise the flow of traffic through their cooperation and coordination. This is called the traffic light control problem and is often an application for reinforcement learning research; the state of the environment is the location of cars, and the actions are the possible light sequences. Traffic light technology is are becoming increasingly advanced, including sensor-based vehicle detection and speed monitoring capabilities. When considering the traffic light control problem for a whole city, the state-action space is much too large (increasing exponentially with the number of traffic lights) to solve analytically for the best solutions, so multi-agent reinforcement learning (MARL) or heuristics can be applied to estimate the solution. For an overview of MARL, refer to this survey paper Buşoniu *et al.* (2008).

In this thesis, we address the problem of reducing congestion using the following methods: optimal information allocation, such as navigation apps; encouraging safe cooperation in the social dilemma of route choice; and, optimising traffic light cycles. In each of these variants, we apply game theoretic solution concepts to analyse traffic problems.

The analysis is mainly based on congestion games Rosenthal (1973), a standard framework of algorithmic game theory used to study the equilibria of traffic flows. They are non-cooperative games of perfect information where self-interested actors choose sets of available resources, e.g. roads, and where the cost of each resource depends on its overall usage. The cost of a route can be thought of as an expected journey time, although it could represent other preferences as well. We restrict our analysis to games where players have homogeneous preferences. The congestion game model has many underlying assumptions to simplify the complex dynamics of real-world traffic networks, such as assuming all vehicles are homogeneous. Thus, the results are not directly applicable to real-world networks and instead use traffic as a motivating example for the results. However, results from congestion games have an impact beyond the scope of traffic congestion. Congestion games have been applied to numerous other domains such as power grid demand Ibars *et al.* (2010), cryptocurrency mining Altman *et al.* (2019), wireless communication networks Liu & Wu (2008), peer-to-peer computing Suri *et al.* (2004), virtual drug screening Nikitina *et al.* (2018), and cloud computing Anselmi *et al.* (2014). We focus on nonatomic congestion games that are played on a network.

Equilibria occur in congestion games when travellers have minimal and equal costs. The outcome of selfish routing is that equilibrium costs are often much higher than the social optimum, the ratio of these costs being the Price of Anarchy. An efficiency-related phenomenon occurring in these games is Braess' paradox Braess (1968), i.e., the existence of traffic networks that suffer from the increase of total cost when the cost of an available resource strictly decreases. We consider a variant where players do not have full information about the routes available to them and one where we equip edge-costs with waiting times to model traffic lights at junctions.

Although these results are influenced by the behaviour of traffic networks, it should be noted that this is merely a motivating scenario rather than a model that can be directly applied to traffic analysis. There are many limiting assumptions made that reduce the applicability of these theorems to real-world traffic, such as fixed origin and destinations, identical size and mass of vehicles, traffic routing exists at an equilibrium state, and drivers have full network information. Any data used throughout this thesis is simulated game data and the applications of these results to other areas, such as power grid demand modelling or wireless communication networks, may be more realistic.

1.2 Research Questions

The chapters of this thesis can be summarised by the following question.

How can inefficiency in network routing be reduced?

Each chapter considers a different variant of route selection including cooperation in route choice, restricted information, information design, traffic light cycles, and intelligent traffic lights. The specific research questions for each chapter are outlined below.

Chapter 3 Safety in Sequential Social Dilemmas

- *How do we equip agents with a belief system that provides the desirable properties of safety and cooperation?* To achieve safety, we want to earn the value of the game on average, requiring cautious play. Cooperation leaves you open to exploitation since you must trust your opponent to cooperate as well. We consider what beliefs about your opponents' play will achieve these two properties.
- *Which beliefs about opponent strategies are safe to best respond to?* Rational players best respond to their beliefs. Believing that your opponent is adversarial (minimises your utility) is safe since your best respond is to play the strategy that guarantees the value of the game. What other beliefs are safe? How can we guarantee ϵ-safe beliefs for some $\epsilon > 0$?
- *How do we quantify the trade-off between safety and cooperation?* The concepts of safety and cooperation are necessarily in tension. However, when the game is played repeatedly, this tension could reduce over time. If the cooperative strategy is not safe, then how can we quantify the difference in the expected utility of a policy against playing safely?
- *How well do safety and cooperation inducing beliefs perform in tournaments?* Safety and cooperation are both sensible properties. However, what we really want is that these strategies perform well against other strategies. We shall compare our player with the well-known tit-for-tat strategy. We want to optimise joint payoff within the space of safe cooperation-inducing beliefs.

Chapter 4 Bounding the Inefficiencies of Route Control

- *How efficient is route control with multiple route planners controlling flow on the same network?* When only one route planner is controlling the flow, they

are able to achieve socially optimal routing. Does this property still hold if there are multiple planners? We also want to show that there is an equilibrium and prove whether or not it is unique.

- *Is it possible to achieve socially optimal routing with multiple route planners?* If we take instances with multiple route planners, we hope to find out whether there are any properties that mean socially optimal routing can be achieved. For example, by changing the proportions of flow controlled by each router.

- *How does the Price of Anarchy change with the number of route planners?* The Price of Anarchy refers to the ratio of social cost between the best outcome and the worst Nash equilibrium. Thus, we will address how the equilibrium reached by the route planners compares with the best possible outcome.

- *Is there an equilibrium if we allow vehicles to choose their route planners? Is this system more efficient?* We will address what happens when drivers have control over which route planner they choose and discover the equilibrium properties of this game.

Chapter 5 Distribution of Information in Nonatomic Congestion Games

- *More specifically, what traffic network structures are immune to informational Braess' paradox?* When we say a nonatomic congestion game has heterogeneous information, we mean that players have different knowledge about the routes available to them. We will consider different road network structures and find ones that guarantee receiving any information (about the edge set) will not increase your journey time.

- *How does the distribution of information affect the social cost?* Informational Braess' paradox considers only the cost of the receiver of information, yet, the information could have an impact on all drivers in the network. We will address the impact of network structures on whether distribution of information will not increase the sum of journey times for all players.

Chapter 6 Nonatomic Congestion Games with Traffic Lights

- *How can traffic lights be represented in a nonatomic congestion game?* Normally, congestion games do not consider the properties of junctions. We aim to formulate traffic lights in a nonatomic congestion game whilst preserving

their real-world properties. We will use traffic simulation software to confirm that any simplifying assumptions allow for a realistic traffic light model.

- *How does the presence of traffic lights affect equilibria?* When implementing traffic lights, we should consider whether the equilibrium properties of nonatomic congestion games are preserved.

- *Can the implementation of traffic lights make a network immune to Braess' paradox?* We look at which traffic light phases for a traffic network mean that Braess' paradox cannot occur.

- *How do reinforcement learning traffic lights effect routing behaviours?* We consider what happens when traffic lights are able to choose their own cycles in a nonatomic congestion game. We then find out whether the equilibrium reached is optimal or inefficient.

- *Are reinforcement learning traffic lights fair?* We consider the problem of quantifying fairness in the context of traffic lights. We use simulation to see whether reinforcement learning traffic lights treat some vehicles unfairly by increasing their journey times, in order to optimise aggregate travel times. We also address some properties that vehicles would use to decide whether or not they were treated fairly.

1.3 Publications

Work from this thesis is published, or to be published, as declared below.

- "Multi-Population Congestion Games with Incomplete Information" is published as a full paper in the IJCAI conference proceedings for 2019 Roman & Turrini (2019). This work appears in Chapter 5.

- "Accumulating Risk Capital Through Investing in Cooperation" is published as a full paper in the AAMAS 2021 conferences proceedings Roman *et al.* (2021). This work appears in Chapter 3.

- Manuscripts under review are "The Inefficiency of Multiple Route Controllers in Transportation Networks" and "The Cost of Traffic Lights: Equilibria with Waiting Times in Nonatomic Congestion Games". Work from these papers is included in Chapters 4 and 6 respectively.

1.4 Thesis Outline

This thesis starts with a chapter of the relevant mathematical preliminaries required for understanding, Chapter 2.

It follows with Chapter 3, on the topic of sequential social dilemmas. Using policy-conditioned beliefs, we find the conditions for ϵ-safety of beliefs. We then continue by finding a cooperation-inducing policy-conditioned belief for two-player matrix games. We show that the trade-off between safety and cooperation diminishes over time. We then equip the cooperation-inducing belief with the safety property to formulate the algorithm called 'Accumulating Risk Capital Through Investing in Cooperation'. The performance of this algorithm is then shown in Iterated Prisoner's Dilemma, Stag Hunt, and Route Choice games. Furthermore, we demonstrate its applicability to multi-agent reinforcement learning.

The contents of Chapter 3 are relevant for all social dilemmas, including those that exist outside of traffic. The work on social dilemmas is continued in Chapter 4 but now focuses on the existence of a social dilemma in routing, using route control of autonomous vehicles as the motivating example.

Chapter 4 considers the problem of intelligent agents distributing routes of players in a nonatomic information constrained congestion game. We call this the network control game and show that it is an exact potential game with essentially unique equilibria. We bound the Price of Anarchy for network control games with polynomial cost functions as a function of the number of active route planners. Furthermore, we consider a game in which drivers choose their own route planner, and show that there exists a unique equilibrium with maximum social cost.

In Chapter 5, equilibria in information constrained congestion games is analysed further. Specifically, looking at a variant of a well-known phenomena called Braess' paradox. We study traffic networks with multiple origin-destination pairs, relaxing the simplifying assumption of agents having complete knowledge of the network structure. Firstly, we show that a game with matroid base strategy spaces can safely increase the agents' knowledge without affecting their own overall performance. We then identify a ubiquitous class of networks, i.e., rings, which have the same property, known as immunity to informational Braess' paradox. By extension of this performance measure to include the welfare of all agents, i.e., minimisation of the social cost, we show that informational Braess' paradox is a widespread phenomenon and no network is immune to it.

The thesis continues by considering how another form of intelligent transportation system technology, the traffic light, effects congestion game equilibria and

the existence of Braess' paradox. In Chapter 6, we equip congestion games with traffic lights, modelled as junction-based waiting cycles, therefore enabling individuals with more realistic path planning strategies. Using the SUMO simulator, we show that our modelling choices coincide with simulated routing behaviours. In particular, drivers' decisions about routes are based on the proportion of red light time for their direction of travel. Drawing upon the experimental results, we show that the effects of the notorious Braess' paradox can be outright avoided in theory and at least significantly reduced in the games, by allocating the appropriate traffic light phases in a transport network. Furthermore, intelligent traffic lights in congestion games using learning algorithms will converge to an essentially unique equilibrium, independent of the algorithm chosen. A further simulation study of implementing intelligent traffic lights in traffic simulators is included. The results from these simulations suggest a bias in journey times is caused by the adaptive traffic lights, with these effects quantified. We consider changing the reward function as a method of mitigating bias.

Finally, Chapter 7 discusses the interconnectivity of ideas brought forth in this thesis. Moreover, the implications for future research are considered.

CHAPTER 2

Preliminaries

Here we outline the required mathematical preliminaries and notation required for understanding Chapters 3, 4, 5, and 6. In section 2.1, we introduce the notation and relevant definitions for finite games including the concepts of best-response and safety which are core to the theory introduced in later chapters. We then explain a specific type of game called a "social dilemma" in section 2.2. Congestion games is a model that occurs throughout the thesis and we include a review of essential theory central to congestion games in section 2.3, including illustrating an important motivating example of the inefficiency of selfish routing models called "Braess' Paradox" and the required graph theory definitions to be able to characterise network immunity to Braess' paradox. In section 2.4, we explain a method of solving nonlinear optimisation problems which will be used in later examples. Finally, we include a summary of reinforcement learning theory in section 2.5 including multi-agent reinforcement learning and the pseudocode of algorithms later used throughout the thesis.

Below is a table that summarises which sections of this chapter are required for which of the following chapters.

Table 2.1: Table mapping required preliminaries to chapters.

		Required for
Section 2.1	Basic Game Definitions	Chapters 3, 4, 5, and 6
Section 2.2	Social Dilemmas	Chapters 3 and 4
Section 2.3	Congestion Games	Chapters 4, 5, and 6
Section 2.4	Nonlinear Optimisation	Chapters 4 and 6
Section 2.5	Reinforcement Learning	Chapters 3, 4 and 6

2.1 Basic Game Definitions

A *finite game* is a tuple (N, A, u) where: $N = \{1, 2, ..., n\}$ is the set of agents[1]; $A = A_1 \times ... \times A_n$ is the action space of all players, $n \geq 2$; and, $u = (u_1, ..., u_n)$ is a vector of utility functions, where $u_i : A \to \mathbb{R}$ is a convex utility function for player i.

The *mixed strategy space* of player i is denoted Σ_i, where $\sigma_i \in \Sigma_i$ is a probability distribution over A_i. A *strategy profile* $\boldsymbol{\sigma} := (\sigma_i)_{i \in N}$ is the joint mixed strategy of the players, where $\sigma_i \in \Sigma$. The notation $-i$ corresponds to all players in $N \backslash \{i\}$, i.e., $\boldsymbol{\sigma}$ and (σ_i, σ_{-i}) are interchangeable. Thus, the strategy space of the opponents of i is denoted $\Sigma_{-i} = \Pi_{j \neq i} \Sigma_j$. The *expected utility* of player i is denoted $E[u_i(\sigma_i, \sigma_{-i})]$.

Then the *minimax value* of the game v_i is the highest value player i can guarantee without knowing their opponents' actions,

$$v_i = \max_{\sigma_i \in \Sigma_i} \min_{\sigma_{-i} \in \Sigma_{-i}} E[u_i(\sigma_i, \sigma_{-i})].$$

A *best response* (BR) to a strategy profile σ_{-i} is defined as $BR(\sigma_{-i}) := \arg\max_{\sigma_i \in \Sigma_i} E[u_i(\sigma_i, \sigma_{-i})]$. It is rational for players to choose a best response if they know their opponents' strategies, since this maximises their payoff. If all players play a best response to their opponents, then we have an equilibrium. A *Nash equilibrium* (NE) is a strategy profile $\boldsymbol{\sigma} \in \prod_{i \in N} \Sigma_i$ such that $\forall i \in N$, $\sigma_i \in BR(\sigma_{-i})$. The definitions of best response and Nash equilibrium are core to game theory and are referenced throughout this thesis. For a more comprehensive introduction to finite games, see Maschler *et al.* (2013).

A strategy σ_i is *safe*[2] if it guarantees at least the minimax value on average:

$$E[u_i(\sigma_i, \sigma_{-i})] \geq v_i \text{ for any } \sigma_{-i} \in \Sigma_{-i}.$$

A strategy σ_i is ϵ-*safe* if, and only if, for all $\sigma_{-i} \in \Sigma_{-i}$ we have: $v_i - \epsilon \leq E[u_i(\sigma_i, \sigma_{-i})]$. The definition of safety is a key concept in Chapter 3.

The Risk What You've Won in Expectation algorithm plays an ϵ-safe best response to a model of an opponent's strategy M and time horizon $T < \infty$, achieving safety Ganzfried & Sandholm (2012). Pseudocode is shown in Algorithm 1, where the set of ϵ-safe strategies is denoted $SAFE(\epsilon)$. Our proposed ARCTIC algorithm

[1] We use the terms agent and player interchangeably throughout.

[2] Throughout this thesis we will refer to safety in this context and not the context of danger arising from vehicle movement.

in Chapter 3 follows the same structure, thus, achieves safety.

Algorithm 1: Risk What You've Won in Expectation Ganzfried & Sandholm (2012)

Initialize $\epsilon_0 \leftarrow 0$, $v_i \leftarrow$minimax value;
for $t = 1$ *to* T **do**
  $\pi_i \leftarrow \arg\max_{\sigma_i \in SAFE(\epsilon_t)} E[u_i(\sigma_i, M)]$;
  i plays σ_i from π_i;
  $-i$ plays σ_{-i} from unknown π_{-i};
  $\epsilon_{t+1} \leftarrow \epsilon_t + E[u_i(\pi_i, \sigma_{-i})] - v_i$
end

2.2 Social Dilemmas

A social dilemma is a collective action problem in which individuals would be better off by cooperating yet conflicting interests exist on the individual level. In a social dilemma game, each player must choose to either cooperate or defect with their opponents. Mutual cooperation (or coordination) gains the highest total rewards, but there is an incentive to deviate from this. We will explore the conditions for beliefs about opponent strategies to incentivise cooperation in sequential social dilemmas (SSDs) in Chapter 3.

Here, we introduce the format of two-player matrix game SSDs. The possible payoffs for each stage game are: reward R for mutual cooperation, punishment P for mutual defection, sucker S if exploited, and temptation T for exploiting. Table 2.2 shows the form of a two-player matrix game to be social dilemma, where we must have $R > P$, $R > S$, $2R > T + S$, and either $T > R$, and/or $P > S$ Macy & Flache (2002).

Table 2.2: The format of social dilemma payoffs in a two-player matrix game.

	C	D
C	(R, R)	(S, T)
D	(T, S)	(P, P)

Example games of Prisoner's Dilemma, Stag Hunt, and Route Choice are shown in Table 2.3. Prisoner's Dilemma is the most well studied social dilemma, since defect is a dominant strategy and so mutual defection is the Nash equilibrium. In Stag Hunt, mutual cooperation is also a Nash equilibrium, but it is not stable

since if either player chooses to defect, the best response is to also defect. Route Choice was later introduced by Helbing *et al.* (2005) as an example (outside of the set defined by Macy & Flache (2002)) of a social dilemma where coordination is required. To maximise joint payoff, players must play a mixed strategy, taking turns to cooperate and defect.

Table 2.3: Examples of payoff matrices for two-player SSDs (payoffs normalised on $[0, 1]$): Prisoner's Dilemma (left), Stag Hunt (centre), and Route Choice (right).

	C	D
C	$(3/4, 3/4)$	$(0, 1)$
D	$(1, 0)$	$(1/4, 1/4)$

	C	D
C	$(1, 1)$	$(0, 3/4)$
D	$(3/4, 0)$	$(1/4, 1/4)$

	C	D
C	$(0, 0)$	$(1/4, 1)$
D	$(1, 1/4)$	$(1/2, 1/2)$

Desirable properties of strategies in SSDs include that they are not exploitable, i.e., safe; achieve mutual cooperation with cooperative strategies, including itself; forgiving, such that if either player accidentally defects, cooperation can be reached again. The well-known strategy 'tit-for-tat', whereby one cooperates initially and then continues by imitating their opponent's previous move, attributes these properties to its success Axelrod & Hamilton (1981).

2.3 Congestion Games

Congestion games are the standard framework of algorithmic game theory to study the equilibria of traffic flows. They are non-cooperative games of perfect information where self-interested actors choose sets of available resources, where the cost of each resource depends on its overall usage. Later, we will see that congestion games are similar to social dilemmas since there is an individual incentive to use suboptimal routing, whilst cooperation between players yields the best utility.

The idea of traffic equilibria was first posed by Wardrop (1952) as an equilibrium that exists when a population of selfish drivers choose routes such that their journey times are equal and less than any unused routes. This equilibrium is the same "user equilibrium" that we address in congestion games, in spite of being developed outside of the field of game theory. Congestion game were first formalised by Rosenthal (1973) as a class of games that always have a pure strategy Nash equilibrium. We use congestion games to model traffic flows when the set of resources are edges on a network. Chapters 4, 5, and 6 all address theory related to congestion games on a network.

In the atomic instance, there are a discrete number of players. Often in traffic

models, it is easier and is more appropriate to use the nonatomic form that assumes an infinite number of players exist each controlling a negligible amount of "flow". These players are then grouped into populations such that all members have the same origin and destination nodes. Congestion games make the assumption that players have predetermined and fixed origin and destination nodes. This assumption can be limiting to real-world application of congestion games to traffic but is necessary to define the strategy sets of players.

Let $N = \{1, ..., n\}$ be a nonempty finite set of agents populations such that players in the same population have the same strategy set. The demand for a population, i.e., the traffic rate associated with that population, is $d_i \geq 0$.

Each population has a nonempty finite resource set E_i made up of *relevant* resources, i.e., those which are used in at least one strategy, $S_i \subseteq 2^{E_i}$. Denote E as the *irredundant* resource set $E = \bigcup_{i \in N} E_i$. Finally, resource cost functions, $c_e : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ such that $e \in E$, are assumed to be continuous, nondecreasing and nonnegative. Formally, a *nonatomic congestion game* is defined as a tuple $\mathcal{M} = (N, (E_i)_{i \in N}, (S_i)_{i \in N}, (c_e)_{e \in E}, (d_i)_{i \in N})$.

The outcome of all players of population i choosing their strategy leads to a strategy distribution $\boldsymbol{x}^i$ satisfying $\sum_{s \in S_i} x_s^i = d_i$ and $x_s^i \geq 0$, $\forall s \in S_i$. A strategy distribution or outcome $\boldsymbol{x} = (\boldsymbol{x}^i)_{i \in N}$ is *feasible* if $\sum_{s \in S_i} x_s^i = d_i$, $\forall i \in N$. Denote then the load on e in an outcome $\boldsymbol{x}$ to be $f_e(\boldsymbol{x}) = \sum_{i \in N} \sum_{s \in S_i} x_s^i \mathbb{1}_s(e)$, where $\mathbb{1}$ is the indicator function. Moreover, in $\boldsymbol{x}$, let a player from population i be charged a cost function $C_i(s, \boldsymbol{x}) := \sum_{e \in s} c_e(f_e(\boldsymbol{x}))$ when selecting strategy $s \in S_i$. For only one population, we can write the cost function as C instead of C_i.

A *user equilibrium* (UE), also known as Wardrop equilibrium, is a strategy distribution $\boldsymbol{x}$, such that every player of every population chooses a strategy with minimum cost. More formally, a UE is a strategy distribution $\boldsymbol{x}$ such that the following inequality holds true $\sum_{e \in s_i} c_e(f_e(\boldsymbol{x})) \leq \sum_{e \in s_i'} c_e(f_e(\boldsymbol{x}))$ for all $s_i, s_i' \in S_i$ such that $\boldsymbol{x}_{s_i}^i > 0$, $\forall i \in N$. Since every player in a population i has the same cost at a UE $\boldsymbol{x}$, we denote this as $C_i(\boldsymbol{x})$.

The *social cost* of $\boldsymbol{x}$, is the total cost incurred by all players $SC(\boldsymbol{x}) = \sum_{i \in N} C_i(\boldsymbol{x}) d_i = \sum_{e \in E} f_e(\boldsymbol{x}) c_e(f_e(\boldsymbol{x}))$. We say that a user equilibrium is *essentially unique* if all user equilibrium have the same social cost. For any nonatomic congestion game, there exists a UE and it is essentially unique Smith (1979). Congestion games are an important class of games since they have a pure strategy Nash equilibrium for both atomic Monderer & Shapley (1996) and nonatomic Schmeidler (1973) variants.

The strategy distribution $\boldsymbol{x}$ is a *social optimum* (SO) if it solves the following

minimisation problem:

$$\begin{aligned} &\text{minimise } SC(\boldsymbol{x}) \\ &\text{subject to } \sum_{s_i \in S_i} x_i^{s_i} = d_i \qquad \forall i \in N \\ &\qquad\qquad x_i^{s_i} \geq 0 \qquad\qquad \forall i \in N \end{aligned}$$

In most cases, the SO solution is different to the UE solution, since players only maximise individual utility. Thus, the term selfish routing is used to describe the strategies in congestion games. The contrast between selfish routing and social optima gives rise to interesting phenomena.

For a thorough introduction to atomic and nonatomic congestion games, we refer the reader to Shoham & Leyton-Brown (2008).

Price of Anarchy

The efficiency of the UE when compared with the SO is measured by the *Price of Anarchy* (PoA) Koutsoupias & Papadimitriou (1999). It is defined as the ratio between the worst social cost of a UE and the social cost of a SO outcome. For any UE $\boldsymbol{y}$, and feasible strategy distribution $\boldsymbol{x}$,

$$PoA = \frac{\max_{\boldsymbol{y}} SC(\boldsymbol{y})}{\min_{\boldsymbol{x}} SC(\boldsymbol{x})}.$$

Note, that a social dilemma necessarily has a Price of Anarchy strictly greater than 1. A cost function C is called (λ, μ)-*smooth* if $C(\boldsymbol{x}) \leq \lambda C(\boldsymbol{x}^*) + \mu C(\boldsymbol{x})$ for all strategy profiles $\boldsymbol{x}, \boldsymbol{x}^*$. If all edge-cost functions are (λ, μ)-smooth for $\mu < 1$, then $PoA = \frac{\lambda}{1-\mu}$ Roughgarden (2003).

We will later use the concept of the Price of Anarchy as a measure of inefficiency in Chapters 4 and 6.

Braess' Paradox

Braess' paradox is a phenomenon that arises when the cost of a resource is strictly decreased, yet results in a strict increase in the social cost of the equilibria Braess (1968). This can be observed in the Wheatstone network in Figure 2.1.

Assume that there is a population of unit size that wish to travel between nodes O and D. In the first instance of cost functions, as shown in left side network from Figure 2.1, any UE requires that $\frac{1}{2}$ choose the path $\{O, 1, D\}$ and $\frac{1}{2}$ choose

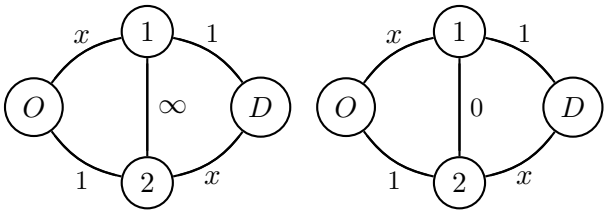

Figure 2.1: Braess' paradox on the Wheatstone network with one population of players, where O and D are the respective origin and destination nodes. When the demand of the population is 1, the equilibrium social cost of travel is $\frac{3}{2}$ before and 2 after reducing the costs of the middle edge from ∞ to 0.

the path $\{O, 2, D\}$ generating a social cost of $\frac{3}{2}$. By reducing the cost of the edge 12 to be 0, the resulting UE requires that all players now choose to travel the path $\{O, 1, 2, D\}$. This increases the social cost of the equilibrium to 2, despite the strict reduction of edge costs.

More formally, a set of systems $(E, S_i)_{i\in N}$ admits *Braess' paradox* (BP) if there are two nonatomic congestion games $\mathcal{M} = (N, E, (S_i)_{i\in N}, (c_e)_{e\in E}, (d_i)_{i\in N})$ and $\mathcal{M}' = (N, E, (S_i)_{i\in N}, (c'_e)_{e\in E}, (d'_i)_{i\in N})$ where $c'_e(t) \leq c_e(t)$, $\forall t \geq 0$ and $d'_i \leq d_i$, $\forall i \in N$, and two UE $\boldsymbol{x}$ and $\boldsymbol{x}'$, such that $SC(\boldsymbol{x}) < SC(\boldsymbol{x}')$. If no such $\mathcal{M}$ and $\mathcal{M}'$ exist, then we say that the set system is *immune* to Braess' paradox.

In Chapter 6, we consider how the implementation of traffic lights in a network could create and remove Braess' paradox.

Potential Games

The concept of potential games was first posed by Monderer & Shapley (1996) for atomic games, and later extended to nonatomic games Sandholm (2001); Cheung & Lahkar (2018). An *exact potential game* is one that can be expressed using a single global payoff function called the potential function. More formally, a game is an exact potential game if, and only if, it has an exact potential function $\Phi : A \to \mathbb{R}$ such that $\forall a_{-i} \in A_{-i}$, $\forall a_i, a'_i \in A_i$,

$$\Phi(a_i, a_{-i}) - \Phi(a'_i, a_{-i}) = u_i(a_i, a_{-i}) - u_i(a'_i, a_{-i})$$

Potential games and congestion games are equivalent, where a player's utility is their negative cost. Another notable property of atomic potential games is that a Nash equilibrium is reached through myopic best response Monderer & Shapley

(1996). The potential function for nonatomic congestion games is

$$\Phi(\boldsymbol{x}) := \sum_{e \in E} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}) d\boldsymbol{z}, \tag{2.1}$$

where $\boldsymbol{x}$ is the strategy distribution of players, also referred to as the Beckmann function Beckmann *et al.* (1956).

A strategy distribution is an ICUE if, and only if, it minimises the potential function Acemoglu *et al.* (2018) (an extension of results in Beckmann *et al.* (1956); Smith (1979). All local maximisers of potential are equilibria Sandholm (2001).

2.3.1 Immunity to Braess' Paradox

Before we introduce some known results on immunity to Braess' paradox, we must first explain the following terms from graph theory, where further details can be found in Bondy *et al.* (1976).

A *simple network* $G = (V, E)$ is an undirected graph with at most one edge between any pair of nodes and no self-loops. A *path* is an ordered collection of edges such that adjacent pairs of edges share a node. If a path visits no node more than once then it is called *acyclic*. A network is *connected* if it has at least one vertex and there is a path between every pair of vertices. A *tree* is a connected simple network that has only acyclic paths. A *spanning tree* of a undirected network is a connected acyclic subnetwork, connecting all nodes of the network. A *directed spanning tree* of a directed network is a connected acyclic subnetwork such that there exists a path to all nodes from a source node. A *ring* (or cycle) is a connected network such that every node connects to exactly two others, forming a single continuous loop.

Definition 2.3.1 (Network Nonatomic Congestion Game)**.** *A* ***network nonatomic congestion game*** *is played on an undirected network $G = (V, E)$, where the resources are edges and players move between the distinct origin and destination terminal nodes $O_i, D_i \in V$ for any $i \in N$. The strategies of players are choices of paths such that no vertex is visited more than once and such that the start and end nodes are the associated origin and destination.*

If a network is *two-terminal*, then there is a single origin and destination pair for players to travel between. An *asymmetric* (or multi-population) game is one in which there are multiple OD pairs.

A two-terminal network is *series-parallel* if it is either a single edge, or composed recursively by joining two series-parallel networks in series or in parallel. We say that a two-terminal network is *linearly independent* (LI) if each path has at

least one edge that does not belong to any other path. A network is *series linearly independent* (SLI) if, and only if, (i) it comprises a single LI network, or (ii) it is constructed by connecting two SLI networks in series.

An *embedding* is a collection of injective maps from the sets of relevant resources to the irredundant resources. For example, Figure 2.2 shows how the Wheatstone network is embedded in a grid road system. More formally, an embedding is a collection of injective maps $\tau := (\tau_i)_{i \in N}$, for $\tau_i : E_i \to \hat{E}$ where $|\hat{E}| = \sum_{i \in N} |E_i|$.

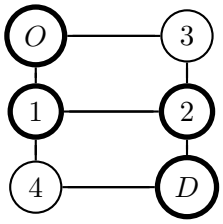

Figure 2.2: A Wheatstone network is embedded in any lattice.

The exact network properties of immunity to Braess' paradox is known. Theorem 2.3.2 characterises immunity in the two-terminal case.

Theorem 2.3.2. *Milchtaich (2006) Braess' paradox does not occur in nonatomic congestion game played on a two-terminal network G if, and only if, G is series-parallel.*

Suppose that nonatomic congestion games allow for multiple origin-destination pairs, hence $|N| > 1$. Before we can characterise networks immune to BP, we must state a few more definitions. A *relevant* network $G_i = (V_i, E_i)$ for a player i as the set of edges and nodes used in every possible path given a player's information set and OD pair. For two SLI networks G_i and G_j, a *coincident block* is a common LI subgraph of G_i and G_j with the same set of terminal nodes.

Theorem 2.3.3. *Chen* et al. *(2015) Braess' paradox cannot occur in a network nonatomic congestion game played on G if, and only if, every relevant network G_i $\forall i \in N$ is series-parallel and $\forall i, j \in N$, $i \neq j$, either $E_i \cap E_j = \emptyset$ or the network induced by $E_i \cap E_j$ consists of all coincident blocks of G_i and G_j.*

Another way to characterise network immunity is through matroids. Thus, we define the following combinatorial theory that will help us define strategy sets that have immunity to congestion paradoxes.

A *matroid* is a tuple $M = (E, \mathcal{I})$, where E is a finite set called the ground set, and $\mathcal{I} \subseteq 2^E$ is a nonempty family of subsets of E called the *independent sets*

that hold the following properties: (i) $\emptyset \in \mathcal{I}$, (ii) *hereditary property*: if $X \in \mathcal{I}$ and $Y \subseteq X$, then $Y \in \mathcal{I}$; and (iii) *augmentation property*: if $X, Y \in \mathcal{I}$ with $|X| < |Y|$, then $\exists e \in Y \backslash X$ such that $X \cup \{e\} \in \mathcal{I}$. The inclusion-wise maximal independent sets of $\mathcal{I}$ are called the *bases* of the matroid and are denoted $\mathcal{B}$. The *rank* of a matroid M, denoted $rk(M)$, is the cardinality of a basis of M. For more information on matroids see Oxley (2006).

Definition 2.3.4 (Matroid Nonatomic Congestion Game)**.** *A* ***matroid nonatomic congestion game*** *is a nonatomic congestion game $\mathcal{M}$ such that $\forall i \in N$ there is a matroid $M_i = (E_i, \mathcal{I}_i)$ such that the strategy set S_i is equivalent to the base set of the matroid $\mathcal{B}_i$.*

Since the independent sets $\mathcal{I}_i$ are generated by the base set $\mathcal{B}_i$, we simplify notation by writing such matroids as $M_i = (E_i, S_i)$. A tuple $(E, (S_i)_{i \in N})$ is said to be *universally immune* to Braess' paradox if it is immune for all embeddings τ in $\hat{E}$. We know that a matroid structure is universally immune to Braess' paradox.

Theorem 2.3.5. *Fujishige* et al. *(2017) If $(E, (S_i)_{i \in N})$ forms the base set of a matroid $M_i = (E, S_i) \forall i \in N$, then the associated nonatomic congestion game is universally immune to Braess' paradox.*

2.3.2 Information Constrained Nonatomic Congestion Games

The assumption that players have full information sets in congestion games is unrealistic. To improve upon this, we can relax the assumption by modelling a variant of the game where players have restricted knowledge[3] about the edges available to them in a network. We address nonatomic congestion games with heterogenous information sets in Chapters 4 and 6.

Let $N = \{1, ..., n\}$ be a nonempty finite set of agent populations. In each population, we suppose that there exists heterogeneity among knowledge of the resources due to previous experience, use of GPS systems etc. For each population i, there are $K_i \geq 1$ information types of players. We refer to a player from population i of type k as (i, k), which we abbreviate ik. The demand for a information type, i.e., the traffic rate associated with that population, is $d_{ik} \geq 0$.

Each population has a nonempty finite resource set E_i, where information types can restrict such knowledge, i.e., each population-type pair is associated with a *known* set $E_{ik} \subseteq E_i$. We assume that each E_i is made of *relevant* resources, i.e., those which are used in at least one strategy, and that strategy sets $S_{ik} \subseteq 2^{E_{ik}}$ only

[3]It is still assumed players' information about congestion and knowledge of edges is without noise.

contain resources from their information set and are disjoint for distinct populations. Denote E as the *irredundant* resource set $E = \bigcup_{i \in N} E_i$. Finally, resource cost functions $c_e : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ such that $e \in E$ are assumed to be continuous, nondecreasing, and nonnegative. Formally, a *nonatomic information constrained congestion game* is defined as a tuple $\mathcal{M} = (N, (K_i), (E_{ik}), (S_{ik}), (c_e)_{e \in E}, (d_{ik}))$, with $i \in N$ and $k \in K_i$.

The outcome of all players of type (i, k) choosing strategies leads to a strategy distribution $\boldsymbol{x}^{ik}$, satisfying $\sum_{s_{ik} \in S_{ik}} x^{ik}_{s_{ik}} = d_{ik}$ and $x^{ik}_{s_{ik}} \geq 0$, $\forall s_{ik} \in S_{ik}$. A strategy distribution (or outcome) $\boldsymbol{x} = (\boldsymbol{x}^{ik})_{\{i \in N, k \in K_i\}}$ is *feasible* if $\sum_{s_{ik} \in S_{ik}} x^{ik}_{s_{ik}} = d_{ik}$, $\forall i \in N$, $k \in K_i$. Henceforth, we focus, without loss of generality, on feasible strategies.

Denote the *load* on e in an outcome $\boldsymbol{x}$ to be $f_e(\boldsymbol{x}) = \sum_{i \in N} \sum_{s_i \in S_i} x^i_{s_i} \mathbb{1}_{s_i}(e)$. At strategy distribution $\boldsymbol{x}$, a player from population i receives a cost function $C_{ik}(s_{ik}, \boldsymbol{x}) := \sum_{e \in s_{ik}} c_e(f_e(\boldsymbol{x}))$, when selecting strategy $s_{ik} \in S_{ik}$.

An *information constrained user equilibrium* (ICUE) is a strategy distribution $\boldsymbol{x}$ such that all players choose a strategy of minimum cost: $\forall i \in N$, $k \in K_i$ and strategies $s_{ik}, s'_{ik} \in S_{(i,k)}$ such that $x^i_{s_{ik}} > 0$ we have $C_{ik}(s_{ik}, \boldsymbol{x}) \leq C_{ik}(s'_{ik}, \boldsymbol{x})$. The *social cost* is the total cost incurred to all players $SC(\boldsymbol{x}) = \sum_{i \in N} \sum_{k \in K_i} C_{ik}(\boldsymbol{x}) d_{ik}$. For any nonatomic information constrained congestion game, there exists an essentially unique ICUE Acemoglu *et al.* (2018).

Informational Braess' Paradox

Informational Braess' paradox (IBP) Acemoglu *et al.* (2018) occurs when one player's type has its information set expanded, without loss of generality type $(1, 1)$, and this paradoxically increases their strategy cost. More formally, IBP occurs if there exist expanded information sets $(\tilde{E}_{(i,k)})_{\{i \in N, k \in K_i\}}$ with $E_{(1,1)} \subset \tilde{E}_{(1,1)}$ and $E_{(i,k)} = \tilde{E}_{(i,k)}$ for any $(i, k) \neq (1, 1)$ with associated ICUE $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$, where the costs increase for the expanded information player $C_{(1,1)}(\boldsymbol{x}) < C_{(1,1)}(\tilde{\boldsymbol{x}})$.

Immunity to IBP is characterised for two-terminal networks but is not yet fully understood for asymmetric networks Acemoglu *et al.* (2018).

Theorem 2.3.6. *Acemoglu* et al. *(2018) A two-terminal network nonatomic congestion game played on network G is immune to IBP if, and only if, G is an SLI network.*

Theorem 2.3.7. *Acemoglu* et al. *(2018) For any nonatomic congestion game on asymmetric network G, where $\forall i \in N$, $G_i = (V_i, E_i)$ is the relevant network, IBP does not occur if the following hold:*
(a) $\forall i \in N$, G_i is SLI

(b) For all distinct $i, j \in N$, either $E_i \cap E_j = \emptyset$, or $E_i \cap E_j$ consists of all coincident blocks of G_i and G_j.

In Chapter 5, we will find a new class of networks that have immunity to IBP in multi-population games.

2.4 Nonlinear Optimisation

In Chapters 4 and 6, when we address nonlinear optimisation problems in congestion games, we will use the Karush-Kuhn-Tucker conditions Karush (1939); Kuhn & Tucker (1951) to reach a solution.

Consider a minimisation problem of an objective function f over X (a convex subset of $\mathbb{R}^n$).

$$\begin{aligned} &\text{minimise } f(\boldsymbol{x}) \\ &\text{subject to } g_i(\boldsymbol{x}) \leq 0 \qquad \text{for } i = 1, .., m \end{aligned}$$

To solve this problem, we write a *Lagrangian function* of the form

$$L(\boldsymbol{x}, \lambda_1, .., \lambda_m) = f(\boldsymbol{x}) - \lambda_1 g_1(\boldsymbol{x}) - ... - \lambda_m g_m(\boldsymbol{x}) \tag{2.2}$$

where $\lambda_1, ..., \lambda_m$ are constants referred to as Lagrange multipliers.

The Karush-Kuhn-Tucker Theorem states that a saddle point of equation 2.2 finds the solution to the optimisation problem. The Karush-Kuhn-Tucker equations, also known as Kuhn-Tucker equations, are formulated to find the saddle point of equation 2.2. For a minimisation problem, the Karush-Kuhn-Tucker conditions are:

$$\begin{array}{lll} \frac{\delta L}{\delta x_1} = 0 & \frac{\delta L}{\delta \lambda_1} \leq 0 & \lambda_1 \frac{\delta L}{\delta \lambda_1} = 0 \\ \frac{\delta L}{\delta x_2} = 0 & \frac{\delta L}{\delta \lambda_2} \leq 0 & \lambda_2 \frac{\delta L}{\delta \lambda_2} = 0 \\ ... & ... & ... \\ \frac{\delta L}{\delta x_n} = 0 & \frac{\delta L}{\delta \lambda_m} \leq 0 & \lambda_m \frac{\delta L}{\delta \lambda_m} = 0 \end{array}$$

These equations are then solved analytically, although a closed-form solution does not always exist.

2.5 Reinforcement Learning

Reinforcement learning (RL) is a setting where an agent can earn rewards for taking actions in a given environment. The goal is to define a policy - a sequence of actions to take in each environmental state in order to maximise rewards. Value functions are used to estimate long-term rewards given that the agent observes a particular state and selects actions aligning with its policy. Equivalently, this can be formalised as a single-player stochastic game where the policy is the agent's strategy. For a more detailed overview of reinforcement learning see Sutton & Barto (2018).

The environment is represented by a state variable, $s \in S$, and the principle task of the agent is to select the best action, $a \in A$, given the current state. An optimal policy states the actions to be taken in a given state to achieve the highest cumulative rewards.

A *Markov decision process* (MDP) is a discrete-time stochastic process that provides a suitable mathematical framework for modelling an agent's reasoning and planning strategies in the face of uncertainty. It satisfies the Markov property - the probability distribution over the next set of states only depends on the current state and not its history, i.e., $P(s_{t+1}|s_t) = P(s_{t+1}|s_t, s_{t-1}, ..., s_1, s_0)$. Formally, we write an MDP as a tuple (S, A, P, R): state space S, action set A, Markovian transition model $P : S \times A \times S \mapsto [0, 1]$, and reward function $R : S \times A \times S \mapsto \mathbb{R}$.

The goal of the agent is to select a sequence of actions, or *policy* π. An *optimal policy* π^* maximises the cumulative discounted return, $R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$, where $\gamma \in [0, 1)$ is a discount factor and $r_i = R(s_i, a_i, s_{i+1})$ is the reward at step i. The state-value function, or *value function*, $V_\pi : S \to \mathbb{R}$ describes the expected value of following policy π from state s: $V_\pi(s) = E_\pi[R_t|s_t = s]$. This equation can be written iteratively to enable dynamic programming:

$$V_\pi(s) = \sum_{a \in A} \pi(s, a) \sum_{s' \in S} P(s, a, s')(R(s, a, s') + \gamma V_\pi(s')) \tag{2.3}$$

The action-value function, or *Q-function*, $Q_\pi : S \times A \to \mathbb{R}$ estimates the expected value of choosing an action a in state s then following policy π: $Q_\pi(s, a) = E_\pi[R_t|s_t = s, a_t = a]$. We can write the Q-function in terms of the value-function as follows:

$$Q_\pi(s, a) = R(s, a, s') + \gamma \sum_{s'} P(s, a, s')V_\pi(s') \tag{2.4}$$

Model-based reinforcement learning is a setting where an agent takes samples from the environment to estimate P and R, then uses planning algorithms to find

an optimal policy. In contrast, *model-free* learning directly estimates the Q-values from experience.

The *behaviour policy* is the policy that an agent uses to choose an action in any given state, and the *target policy* is the policy that the agent uses to learn from reward action pairs. An *off-policy* algorithm is one where the target policy is different that the behaviour policy, whereas in an *on-policy* algorithm the target and behaviour policies are the same. Throughout this thesis we use off-policy algorithms due to their increased capacity for exploration and better sample efficiency since we can use a replay buffer to reuse old data.

Algorithms must be able to balance taking the actions which maximise Q (exploitation), with taking suboptimal actions in order to improve the knowledge of the environment (exploration). A *greedy* policy chooses the actions in each state that maximises the value of Q in each state. An *epsilon greedy policy* is one that selects the greedy action with probability $1-\epsilon$ and chooses uniformly random actions otherwise, for some $\epsilon > 0$. Epsilon greedy is an off-policy exploration strategy.

For RL instances with small action-state space, solutions can be found through dynamic programming. The Bellman optimality equations form the basis for these solutions, including the most well-known (model-free) RL algorithms, such as temporal difference Sutton (1988) and Q-learning Watkins (1989). The value function is updated directly from interaction with the environment to assign rewards to possible state-action pairs. Model-based RL calculates average rewards associated with applying an action in a state by estimating a transitional model of the state-transition probabilities that occur from actions.

Q-Learning

Q-learning Watkins (1989) is a model-free RL algorithm for discrete action and state spaces. The Q-value for each state action pair (s, a) is learned using the update step

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t))$$

where t is the timestep, γ is the discount factor, and α is the learning rate. The pseudocode for Q-learning is shown in Algorithm 2.

In general, choosing a learning rate that is too small means the algorithm could converge to a suboptimal solution, whereas selecting a learning rate that is too small means that algorithm may never converge.

Algorithm 2: Q-Learning (epsilon greedy policy)

Initialise Q arbitrarily;
while Q *not converged* **do**
  Initialise $s \in S$;
  while s *not terminal* **do**
    $a \leftarrow EpsilonGreedy(\arg\max_a Q(s,a), \epsilon)$;
    take action a;
    receive r and observe s';
    $Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a') - Q(s,a)]$;
    $s \leftarrow s'$;
  end
end
return Q

Deep Q-Learning

Deep Q-learning Mnih *et al.* (2013) (or DQN) is an extension of Q-learning with a feed-forward neural network used to model the Q-function. The deep Q network is a neural network architecture that estimates the Q-function with the input encoding the state and outputs the value for each action. Deep Q-learning extends Q-learning by allowing for continuous state spaces.

To aid with convergence, we use two tools: experience replay, and a target Q network. Instead of using the last transition for the Q update, each transition is stored inside the experience replay memory. The update is then done from a batch of randomly sampled transitions from the same experience replay. The target Q network is to stabilise the training by making the fluctuations less severe between steps.

2.5.1 Multi-Agent Reinforcement Learning

Multi-agent reinforcement learning (MARL) is a term that has been used in many ways throughout the literature, since there are a variety of ways to extend single-agent RL to the domain with at least two agents. For example, learning can use centralised or decentralised algorithms; there could be prescriptive or descriptive agents; agents could be cooperative, competitive, or neither.

In Chapters 3 and 4, we consider cooperative MARL in a social dilemma environment. Then, in Chapter 6, we consider cooperative reinforcement learning algorithms that optimise traffic light sequences at junctions.

In a multi-agent setting, each agent must make assumptions about their op-

Algorithm 3: Deep Q-Learning with Experience Replay

Initialise: state s_0, time $t = 0$, replay memory D;
Initialise Q with random weights θ;
Initialise target $\hat{Q}$ with random weights $\hat{\theta}$;
for *epsiode* 1 *to* M **do**
  for $t \leq T$ **do**
    $a_t \leftarrow \text{EpsilonGreedy}(\arg\max_a Q(s_t, a, \theta), \epsilon)$;
    play action a_t ;
    receive r_t and observe s_{t+1};
    store transition (s_t, a_t, r_t, s_{t+1}) in memory D;
    sample random minibatch of transitions from D;
    if *episode terminates at step* $j + 1$ **then**
      $Y_j = r_j$;
    else
      $Y_j = r_j + \gamma \max_{a'} \hat{Q}(s_{j+1}, a', \theta)$;
    end
    perform gradient descent on $(Y_j - Q(s_j, a_j, \theta))^2$ w.r.t. θ;
    Every C steps, reset $\hat{Q} = Q$;
  end
end

ponents' strategies in order to optimise their own payoff. An optimal policy is no longer clearly defined as it depends on the policies of other agents. The MARL algorithms used in this thesis for multi-agent RL problems are Q-learning Hu & Wellman (1998), Asynchronous Advantageous Actor-Critic (A3C) Mnih *et al.* (2016), Deep Q Network (DQN) (or deep Q-learning) Mnih *et al.* (2013) and coordinated reinforcement learning Guestrin *et al.* (2002).

For Q-learning with multiple agents, we consider two types of implementation. The first assumes that agents are independent of one another and have only local state space information. The second is when there exists a single Q-agent with a joint state space that finds the joint optimal action. We refer to the first as *independent Q-learning* and the latter as *multi-agent Q-learning*. The joint action space grows exponentially with the number of agents, thus making it very inefficient for many agents. Similarly, DQN can be implemented with local independent agents or in a multi-agent format.

An MDP can be generalised to capture multiple agents through the use of Markov games. Formally, a *Markov game* is a tuple $(N, S, (A^i)_{i \in N}, P, (R^i)_{i \in N}, \gamma)$ where N is the set of agents, $P : S \times A^1 \times ... \times A^{|N|} \times S \to [0, 1]$ is the transition function, A^i is the action space of $i \in N$, $R^i : S \times ... \times A^{|N|} \times S \to \mathbb{R}$ is the reward

function, and γ is the discount factor.

Denote the action profile of agents at time t is $\boldsymbol{a}_t$, then we can define the *value function* for player i as

$$V^i_{\pi^i,\pi^{-i}}(s) := E[\sum_{t\geq 0} \gamma^t R^i(s_t, \boldsymbol{a}_t, s_{t+1}) | a^i_t \sim \pi^i(.|s_t),\ s_0 = s].$$

For Markov games, a *Nash equilibrium* is a joint policy $\boldsymbol{\pi} = (\pi^1, ..., \pi^{|N|})$ such that for all $i \in N$ and $s \in S$,

$$V^i_{\pi^i,\pi^{-i}}(s) \geq V^i_{\bar{\pi}^i,\pi^{-i}}(s) \text{ for all } \bar{\pi}^i.$$

If all players do not have access to the global state, then we can use partially observable MDPs.

Multi-Agent Q-Learning

This version of Q-learning is adapted for multiple agents. As discussed, we use the term multi-agent Q-learning to refer to an adaption of Q-learning to include multiple actions in the Q-function. Suppose we have a set of agents $\{1, 2, ..., n\}$. Then, Algorithm 4 describes the method of learning the Q-function with a greedy policy.

Algorithm 4: Multi-agent Q-Learning (greedy policy)

Initialise Q arbitrarily;
while Q *not converged* **do**
  Initialise $s \in S$;
  while s *not terminal* **do**
    $a_1, a_2, ..., a_n \leftarrow \arg\max_{a_1,a_2,...,a_n} Q(s, a_1, a_2, ..., a_n)$;
    take actions $a_1, a_2, ..., a_n$;
    receive $r_1, r_2, ..., r_n$ and observe s';
    $Q(s, a_1, a_2, ..., a_n) \leftarrow$
      $(1-\alpha)Q(s, a_1, a_2, ..., a_n) + \alpha[\sum_{i=1}^n r_i + \gamma \max_{\boldsymbol{a}'} Q(s', \boldsymbol{a}')]$;
    $s \leftarrow s'$;
  end
end
return Q

Independent Q-Learning

This is a version of Q-learning adapted for multiple agents where we assume that the agents learn independently from one another. This implementation is similar to Q-learning but with local state space information for each agent. Let $\boldsymbol{s}$ be a vector of the state space for a set of agents $\{1, 2, ..., n\}$. Algorithm 5 shows the pseudocode for this with a greedy policy.

Algorithm 5: Independent Q-Learning (greedy policy)

Initialise Q arbitrarily;
while Q *not converged* **do**
  Initialise $\boldsymbol{s} \in \prod_{i=1,2,...,n} S_i$;
  while $\boldsymbol{s}$ *not terminal* **do**
    for $i = 1, 2, ..., n$ **do**
      $a_i \leftarrow \arg\max_{a_i} Q_i(s_i, a_i)$;
    end
    take actions $a_1, a_2, ..., a_n$;
    receive $r_1, r_2, ..., r_n$ and observe $\boldsymbol{s}'$;
    for $i = 1, 2, ..., n$ **do**
      $Q_i(s_i, a_i) \leftarrow (1 - \alpha)Q_i(s_i, a_i) + \alpha[r_i + \gamma \max_{a_i'} Q_i(s_i', a_i')]$;
    end
    $\boldsymbol{s} \leftarrow \boldsymbol{s}'$;
  end
end
return Q

Multi-Agent/Independent Deep Q-Learning

The extension of DQN to multi-agent DQN and independent DQN uses the same methodology as Algorithms 4 and 5 respectively, combined with deep Q learning from Algorithm 3.

Coordinated Q-Learning

In order to estimate the action profile that gives maximum global utility, we can consider the easier problem of maximising the sum of local utilities. Coordinated reinforcement learning is a cooperative extension of Q-learning whereby agents are connected through a so-called coordination graph that allows for pairwise cooperation. Agents should have access to the states that influence them, i.e. can see their local state and any state of neighbouring agents. Coordination graphs offer a

framework for cooperative decision-making by decomposing a global payoff function into the sum of local rewards.

A coordination graph $G = (N, E)$ is a network that exists between agents. Two agents are connected by an edge if their behaviours directly impact each other. Coordinated reinforcement learning Guestrin *et al.* (2002) combines coordination graphs with MARL by assuming agents have knowledge of all of the state information affecting them but must coordinate their actions to achieve maximum utility. The factorised global Q-function is a linear combination of local Q_e-functions: $Q(\boldsymbol{s}, \boldsymbol{a}) = \sum_e Q_e(s_e, a_e)$ where each $e \in E$ is a pair of neighbouring agents in the coordination graph.

The variable elimination algorithm (VE) is an exact method in which a joint optimal action is found. The procedure works by computing an optimal action only for the last eliminated agent (assuming that the graph is connected) and conditional strategies for other agents. The execution time is exponential in the induced graph width (size of the largest clique) and therefore does not scale well. However, it does converge to an optimal solution. The outcome of VE is independent of elimination ordering, as it always results in the joint optimal action. We choose to follow the heuristic of eliminating agents in the coordination graph in ascending order of degree.

Algorithm 6: Coordinated Q-Learning with VE for two agents

Initialise: state s_0, time $t = 0$;
for $t \leq T$ **do**
  $t \leftarrow t + 1$;
  determine s;
  collect $Q(s, a_1, a_2)$;
  replace $Q(s, a_1, a_2)$ as $f = \max_{a_1} Q(s, a_1, a_2)$;
  $a_1 \leftarrow \text{EpsilonGreedy}(\arg\max_{a \in A_1} f, \epsilon)$;
  $a_2 \leftarrow \text{EpsilonGreedy}(\arg\max_{a \in A_2} f, \epsilon)$;
  receive r and observe s';
  $Q(s, a_1, a_2) \leftarrow Q(s, a_1, a_2) + \alpha[r + \gamma \max_{a'_1, a'_2} Q(s', a'_1, a'_2) - Q(s, a_1, a_2)]$;
  $s \leftarrow s'$;
end

The max-plus algorithm Kok & Vlassis (2005), analogous to belief propagation, is often preferred in real-time systems over variable elimination. However, we use variable elimination for its simplicity.

Asynchronous Advantageous Actor-Critic

Actor-critic methods Konda & Tsitsiklis (2000) are a class of algorithms where a 'critic' advises an 'actor' of the quality of each action. The actor and critic each learn separately. Here, the critic estimates the value function while the actor learns the policy. Asynchronous Advantageous Actor-Critic equips the actor-critic format with independent local agents (asynchronicity) whereby the critics' estimate the Q-function (equation 2.4) minus the value function (equation 2.3), which is called the *advantage function*.

Asynchronous Advantageous Actor-Critic (A3C) is a model-free policy optimisation based MARL algorithm. In policy optimisation, we learn the policy directly, rather than the Q-values. Here, the actor learns the policy and the critic learns the value function. The actor learns based on the feedback from the critic. In deep learning, we learn the parameters θ of the neural network that represents the policy/value function. The A3C implementation used in this thesis comes from the Ray library: `https://github.com/ray-project/ray`. For further details of the algorithm, see Mnih *et al.* (2016).

A3C is an on-policy algorithm since it uses the policy gradient theorem to find an estimate for the gradient of a given policy. Adding the entropy of the policy to the objective function improves exploration by reducing the likelihood of premature convergence. A high entropy coefficient would lead to more random actions whereas a lower entropy coefficient could lead to premature suboptimal convergence.

CHAPTER 3

Safe Cooperation in Social Dilemmas

3.1 Introduction

Sequential social dilemmas (SSDs) are iterated games where short term individual incentives conflict with the long-term social welfare. Social dilemmas exist in traffic networks in many ways, the three most common are route choice, departure time, and mode of transport Klein & Ben-Elia (2016). When choosing strategies in these scenarios, the incentive to be selfish and choose the outcome that has the best possible travel time quickly becomes congested and therefore makes it a worse outcome. Research into cooperation in social dilemmas can help to reduce congestion in roads, thereby improving air quality and journey times.

Countless methods have been proposed for training policies that would better optimise the social good, but many of these methods do so at a greater risk of being exploited. A policy that optimises social welfare could allow other policies to be selfish without long-term consequences, and so the welfare-maximizing policy may fare worse in these settings than if it had only optimised for its own self-interest. To some extent, this is inevitable, since every choice to cooperate leaves room to be exploited. However, though this trade-off is unavoidable, it is not as stark as it first appears.

To study this trade-off formally, we present two objectives. The first objective is the well-studied notion of ϵ-safety, that on average a strategy achieves at least the value of the game minus ϵ. Previous work shows that ϵ-safe policies in sequential settings risk what the policy won in expectation Ganzfried & Sandholm (2012), allowing them in the long run to take much larger risks. The second objective is to perform well with cooperation-promoting policies. We define cooperation-promoting policies to be such that they are more cooperative when faced with cooperative

policies. The existence of cooperation-promoting policies is the reason why a rational designer might consider making an agent that cooperates. Since perfect safety would require only defection and perfect performance with cooperation-promoting policies would require only cooperation, these two objectives are in clear conflict.

Throughout this chapter, we find it convenient to conceive of ϵ-safety using the concept of risk capital. Risk capital is the amount of capital an investor is willing to lose. In our case, we will be using risk capital to refer to the amount of utility we are willing to risk, which for ϵ-safe policies is ϵ. In this framing, we can think of prior work, showing that ϵ-safe policies risk what the policy won in expectation Ganzfried & Sandholm (2012), as suggesting reinvestment of unexpected winnings is a natural form of risk capital.

Since cooperation could always be met with defection, cooperating necessitates some amount of willingness to lose utility. However, like an investment, cooperation often results in returning more utility than was risked. This returned utility can then be reinvested without risking worse outcomes, leading to growing risk capital over time. Thus, more cooperation over time, even for a small initial amount of risk capital.

This argument shows an interesting fact about the trade-off between safety and cooperation – that though cooperation with total safety is impossible, giving away even a small amount of safety will lead to nearly optimal cooperation in the long-term. We formalise this argument as a trade-off between safe beliefs and cooperation promoting beliefs. We go on to propose a method to train policies that satisfy this objective, which we call Accumulating Risk Capital Through Investing in Cooperation (ARCTIC), based on the idea of investing risk capital to achieve long-term cooperation.

3.2 Contributions

In this chapter, we begin by analysing some conditions that guarantee that a strategy is ϵ-safe. We then introduce policy-conditioned beliefs as a mechanism to achieve ϵ-safety through best-response dynamics.

Next, we formulate an example of an ϵ-safe strategy that induces cooperative behaviour in social dilemmas and show how to use it to trade-off between cooperation and safety in SSDs. We then propose a new algorithm called Accumulating Risk Capital Through Investing in Cooperation (ARCTIC) that achieves safe play whilst having the capacity to cooperate with other cooperation capable agents.

Finally, we include some experimental results of ARCTIC and show that

introducing a small level of risk to the initial round of play allows faster convergence of cooperation against itself and other reciprocating or cooperative strategies.

3.3 Literature Review

The problem of cooperation in sequential settings has been extensively studied in game theory. One of the most famous strategies is Rapoport's tit-for-tat Rapoport *et al.* (1965), which achieved great success in Axelrod's tournament Axelrod & Hamilton (1981). In part, Axelrod attributed its success to its ability to promote cooperation, and to the fact that it is impossible to exploit after the first move.

Human experiments with sequential social dilemmas often find that cooperation is the most popular strategy. Gardner et al. Gardner *et al.* (1984) use bounded rationality to explain the high levels of cooperation in lab experiments, where participants played a common pool resource game and could communicate with each other. Cooperation seen in social dilemmas is often founded in indirect reciprocity Rand & Nowak (2013), such as social capital. Viedma Viedma (1999) discusses the role of social capital in social networks. Furthermore, so-called "silly rules" are useful in building social capital in populations Köster *et al.* (2020). Moreover, Capraro et al. introduced the iterated cooperative equilibrium as a solution concept design to capture human behaviour in SSDs Capraro *et al.* (2013).

One theoretical suggestion, *translucent game theory* Capraro & Halpern (2019), explains cooperative behaviours through the idea that others' could detect an intended defection and punish it. This serves as the inspiration for our use of policy-conditioned beliefs. There is evidence that transparency considerations, such as the ability to view opponent's facial expressions, may influence behaviour in social dilemmas de Melo *et al.* (2018); Hoegen *et al.* (2017).

In the reinforcement learning literature, temporal difference learning has been applied successfully to learn cooperation in Prisoner's Dilemma since it maximises future rewards Masuda & Ohtsuki (2009). Leibo et al. Leibo *et al.* (2017) extended classic sequential social dilemma examples into the domain of deep reinforcement learning. Following on from this work, Jaques et al. Jaques *et al.* (2019) used social influence as intrinsic motivation to achieve coordination and communication between agents in SSDs. Additionally, approximate Markov tit-for-tat Lerer & Peysakhovich (2017) maintains desirable properties from tit-for-tat, but applies to deep learning in general games. Reciprocity can be achieved in symmetric games through a system of innovators that maximise their own rewards and imitators that learn to mimic innovator behaviours by measuring the "niceness" of their actions, resulting in a

good performance in SSDs Eccles *et al.* (2019b,a).

The social dilemmas we consider are Prisoner's Dilemma, Stag Hunt and Route Choice. In Prisoner's Dilemma, Press and Dyson Press & Dyson (2012) showed that there exists evolutionary dominant strategies that can only be outperformed by player's that have a theory of mind about their opponent; and, when combining strategies with a memory of one and theory of mind, stable strategies have been classified Glynatsi & Knight (2020). For Stag Hunt, the effects of network topology are important for the emergence of cooperation in games played on a network Van Segbroeck *et al.* (2010). In Route Choice, the coordination of alternating strategies provides additional complexity in achieving cooperation between agents Stark *et al.* (2008).

Safe strategies are essential to restrict adversarial opponents Gleave *et al.* (2019). Ganzfried and Sandholm Ganzfried & Sandholm (2012) explore the necessary properties for a policy to allow for safe opponent exploitation. The Price of Anarchy can be used as an index of inefficiency in social dilemmas Mak & Rapoport (2013). Moreover, the beliefs required for trust-based cooperation in Prisoner's Dilemma were found to correlate with the Price of Anarchy Murphy & Ackermann (2015). Additionally, Chakraborty and Stone Chakraborty & Stone (2014) proposed a method for achieving safety against memory-bounded agents that converges to the Nash equilibrium in self-play.

The three main areas that social dilemmas are found in transportation are: route choice; mode of transport; and, departure times. Dafoe et al. Dafoe *et al.* (2020) discuss the open problems in cooperative AI, using autonomous vehicles and navigation applications as motivation to ground the discussion of why AI tools must consider cooperation as an essential goal. Rapoport et al. Rapoport *et al.* (2009) used experimental data to illustrate that selfish routing is akin to Prisoner's Dilemma. Klein and Ben-Elia Klein & Ben-Elia (2016) reviewed the important game-theoretical and experimental research in these areas, and highlighted the need for further research in the emergence of cooperation in road networks.

3.4 Motivating Example

Consider a two-player game of iterated Prisoner's Dilemma as motivation for the ideas put forward in this chapter[1].

[1]No new ideas are put forward in this example. It is simply included to aid the reader in understanding the ideas put forward later, which combine the safety property and policy-conditioned beliefs.

(1,2)	C	D
C	$(3/4, 3/4)$	$(0, 1)$
D	$(1, 0)$	$(1/4, 1/4)$

The value of the game is $1/4$, since any player can guarantee a payoff of at least this by choosing to defect. Examine the following gameplay where player 1 considers beliefs about her opponent's strategy to decide her own approach.

Suppose that player 1 believes that player 2 will choose to defect. Then her best response would also be to defect. By believing that her opponent will defect, her strategy is safe. She starts with this belief to avoid being exploited by an adversarial player, which would give her an average utility of at least the value of the game.

Round 1. Player 1 chooses D, Player 2 chooses C.

Player 2 chose to cooperate in the first round, earning player 1 a payoff of 1. Given this information, player 1 may wish to re-evaluate her belief. She knows that to get the best long-term payoff, both players must cooperate. If her opponent is willing to cooperate with her, then she can get a payoff of $3/4$. However, she must balance this with the risk that her opponent could exploit her cooperation which would earn her nothing. At the moment, she has an average payoff higher than the game's value. If she cooperates in round 2 and her opponent defects, she will still have earned higher than the value of the game per round. So, player 1 updates her belief so that she believes player 2 will cooperate in round 2, and plays her best response, with respect to the belief, which is to cooperate.

Round 2. Player 1 chooses C, Player 2 chooses D.

This time, player 2 chose to defect. Player 1 is disappointed that her opponent has exploited her. However, she knows that player 2 cooperated in the first round, so they are capable of cooperation. She may reason that player 2 only chose to defect in the last round in retaliation to her original defect. This is risky since there is a chance that the original cooperation was a mistake and player 2 is, in fact, her adversary. Player 1 decides to risk the additional payoff she has earned (since her average payoff is still above the game's value) and chooses to believe that player 2 will cooperate.

Round 3. Player 1 chooses C, Player 2 chooses C.

Player 1's risky strategy has paid off! Both players chose to cooperate. Player 1 continues to play with the same belief. In all subsequent rounds, both players choose to cooperate.

In this example, player 2 was playing tit-for-tat. If player 1 had not updated her original belief that her opponent was adversarial, her average payoff would have

been much lower than what she earned. Although the safe belief would have earned her the value of the game, she was able to achieve mutual cooperation with her opponent by risking her surplus utility. She recognised that her opponent was capable of cooperation and updated her strategy accordingly. The ideas put forward in the rest of this chapter aim to utilise similar reasoning about beliefs to achieve safety and cooperation.

3.5 Safe Beliefs

As a mathematical convenience to represent both our safety criteria and our beliefs about the distribution of opponent strategies, we introduce policy-conditioned beliefs. A *policy-conditioned belief*[2] is a function $p_i : \Sigma_i \to \Sigma_{-i}$, where Σ_i denotes a mixed strategy space for player i. This allows us to represent typical beliefs about an opponent. For example, our opponent could be drawn from some fixed distribution, or could be adaptive to our strategy. The set of *best responses* to a policy-conditioned belief p_i is defined as

$$BR(p_i) := \arg\max_{\sigma_i \in \Sigma_i} E[u_i(\sigma_i, p_i(\sigma_i))].$$

A Nash equilibrium is a strategy profile σ for policy-conditioned belief profile p such that $\sigma_i \in BR(p_i)$ for all $i \in N$. A policy-conditioned belief is *ϵ-safe* if, and only if, for all $\sigma_i \in BR(p_i)$, σ_i is ϵ-safe. An example of a safe belief is the adversarial policy-conditioned belief, we define this policy-conditioned belief as

$$p_i^A(\sigma_i) := \arg\min_{\sigma_{-i} \in \Sigma_{-i}} E[u_i(\sigma_i, \sigma_{-i})] \tag{3.1}$$

The safety property is desirable to minimise risk in environments where an adversarial opponent can exploit a policy. However, playing a safe strategy will end up with suboptimal outcomes. Thus, we shall classify ϵ-safe strategies that can safely cooperate with other agents in any multi-agent game. The payoff matrix for players is assumed to be normalised on $[0, 1]$ for the purpose of simplicity; otherwise, there is an additional coefficient of the range of payoffs[3].

[2]We intend policy-conditioned beliefs to only be a mathematical convenience which allows us to represent both the cooperativeness and safety constraints in one expression, and more easily construct policies which meet them both. Without policy-conditioning, we would not be able to construct a notion of "safe beliefs", and would have to separately analyse safety and cooperativeness, ultimately leading to the same results but with a more fragmented analysis.

[3]For bounded utilities where the greatest range in payoff a player can have is K, replace ϵ-safe with $K\epsilon$-safe in Propositions 3.5.1 and 3.5.3.

Proposition 3.5.1. *For any two-player game with a Nash equilibrium* σ^*, $\forall \epsilon \in [0,1]$ *and* $\forall i \in N$, $\exists \sigma_i \in \Sigma_i$ *such that* $||\sigma_i - \sigma_i^*||_\infty \leq \epsilon$ *and* σ_i *is* ϵ*-safe*[4].

Proof. Since we assumed the payoffs are normalised, we have

$$\max_{\sigma_i} E[u_i(\sigma_i, BR(\sigma_i))] = 1 \text{ and } \min_{\sigma_i} E[u_i(\sigma_i, BR(\sigma_i))] = 0.$$

Let σ_i^{min} be such that $E[u_i(\sigma_i^{min}, BR(\sigma_i^{min}))] = 0$. For any σ_i, such that $||\sigma_i - \sigma_i^*||_\infty \leq \epsilon$, we can bound the expected utility loss by the worst case:

$$E[u_i(\sigma_i^*, BR(\sigma_i^*))] - E[u_i(\sigma_i, BR(\sigma_i))] \leq E[u_i(\sigma_i^*, BR(\sigma_i^*))] - E[u_i(\sigma_i^\epsilon, BR(\sigma_i^\epsilon))],$$

where $\sigma_i^\epsilon = (1-\epsilon)\sigma_i^* + \epsilon\sigma_i^{min}$. We can bound the right-hand side of the inequality using $E[u_i(\sigma_i^*, BR(\sigma_i^*))] \leq 1$ and, since we have assumed convex utility functions, $E[u_i(\sigma_i^\epsilon, BR(\sigma_i^\epsilon))] \geq 1 - \epsilon$. Thus, we have ϵ-safety;

$$E[u_i(\sigma_i^*, BR(\sigma_i^*))] - E[u_i(\sigma_i, BR(\sigma_i))] \leq \epsilon.$$

□

Now that there exist a strategy in the neighbourhood of the Nash equilibrium strategy that is ϵ-safe, we will prove similar properties for policy-conditioned beliefs. Define a policy-conditioned belief p_i to be ϵ*-close* to a policy-conditioned belief p_i' if, and only if,

$$\max_{\sigma_i \in \Sigma_i} ||p_i(\sigma_i) - p_i'(\sigma_i)||_\infty \leq \epsilon.$$

Proposition 3.5.2. *In a two-player game, for any belief* p_i *that is* ϵ*-close to* p_i^A, $\forall \sigma_i \in \Sigma_i$,

$$E[u_i(\sigma_i, p_i(\sigma_i))] - E[u_i(\sigma_i, p_i^A(\sigma_i))] \leq \epsilon.$$

Proof. For all $\sigma_i \in \Sigma_i$ and any policy-conditioned belief p_i such that p_i is ϵ-close to p_i^A, can be bounded by

$$p_i(\sigma_i) = (1-\epsilon)p^A(\sigma_i) + \epsilon p_i^{max}(\sigma_i),$$

where $p_i^{max}(\sigma_i) = \arg\max_{\sigma_{-i}} E[u_i(\sigma_i, \sigma_{-i})]$. We can write the expected utility of belief p_i as $E[u_i(\sigma_i, p_i(\sigma_i))] = E[u_i(\sigma_i, (1-\epsilon)p_i^A(\sigma_i) + \epsilon p^{max}(\sigma_i))]$. Since the utility function is convex, we can find an upper bound:

$$E[u_i(\sigma_i, (1-\epsilon)p_i^A(\sigma_i) + \epsilon p^{max}(\sigma_i))] \leq (1-\epsilon)E[u_i(\sigma_i, p^A)] + \epsilon E[u_i(\sigma_i, p_i^{max}(\sigma_i))].$$

[4]For $\boldsymbol{x} = (x_1, .., x_n)$, the supremum norm is defined as $||\boldsymbol{x}||_\infty := \sup\{|x_1|, ..., |x_n|\}$.

Now, we rearrange the difference in expected utility of p_i and p_i^A:

$$\begin{aligned}
&E[u_i(\sigma_i, p_i(\sigma_i))] - E[u_i(\sigma_i, p_i^A(\sigma_i))] \\
&\quad \leq (1-\epsilon)E[u_i(\sigma_i, p^A)] + \epsilon E[u_i(\sigma_i, p_i^{max}(\sigma_i))] - E[u_i(\sigma_i, p_i^A(\sigma_i))] \\
&\quad \leq \epsilon(E[u_i(\sigma_i, p_i^{max}(\sigma_i))] - E[u_i(\sigma_i, p_i^A(\sigma_i))]) \\
&\quad \leq \epsilon(1-0) = \epsilon
\end{aligned}$$

Therefore, the inequality holds true. □

Thus, beliefs close to p^A have similar expected utilities. Now we can address the safety property in the context of safety.

Proposition 3.5.3. *In any two player game, if the policy-conditioned belief p_i is ϵ-close to the adversarial policy-conditioned belief p_i^A and utilities are bounded on $[0,1]$, then p_i is ϵ-safe.*

Proof. Since p_i is ϵ-close to the adversarial policy-conditioned belief p_i^A, then by Proposition 3.5.2, $\forall \sigma_i \in \Sigma_i$ we have

$$E[u_i(\sigma_i, p_i(\sigma_i))] - E[u_i(\sigma_i, p_i^A(\sigma_i))] \leq \epsilon.$$

Take $\sigma_i \in BR(p_i)$, then

$$\begin{aligned}
E[u_i(\sigma_i, p_i(\sigma_i))] &= \max_{\sigma_i \in \Sigma_i} E[u_i(\sigma_i, p_i(\sigma_i))] \\
&\geq \max_{\sigma_i \in \Sigma_i} E[u_i(\sigma_i, p_i^A(\sigma_i))] \\
&= v_i.
\end{aligned}$$

Thus, $v_i - \epsilon \leq E[u_i(\sigma_i, p_i^A(\sigma_i))]$. Hence, $\forall \sigma_i \in \Sigma_i$, σ_i is ϵ-safe. So, we have that p_i is ϵ-safe. □

Proposition 3.5.3 shows that we can append any policy-conditioned belief with some level of an adversarial policy-conditioned belief and it will be ϵ-safe for some ϵ. Consequently, we can take a policy-conditioned belief that naively cooperates with any opponent and bound its safety through creating an uncertainty; either this belief is true, or they face an adversarial opponent.

To understand these results more intuitively, consider the following example.

Example 1. *Let us readdress the motivating example from Section 3.4, Prisoner's Dilemma, but this time consider the safety of player 1's policy-conditioned beliefs.*

(1, 2)	*C*	*D*
C	$(3/4, 3/4)$	$(0, 1)$
D	$(1, 0)$	$(1/4, 1/4)$

Denote the strategy of a player as a tuple $(\lambda, 1-\lambda)$*, where* $\lambda \in [0, 1]$ *is the probability of playing strategy C. If player 1 had an adversarial belief* p_1^A*, then she believes that player 2 will defect no matter what her strategy is, i.e.* $p_1^A(\lambda, 1-\lambda) := (0, 1)$*. The best response to* p_1^A *would be to always defect, since this maximises her utility. Her expected utility is equal to the value of the game,* $\frac{1}{4}$*. Thus, the belief* p_1^A *is safe.*

Now, suppose that she has the belief that no matter what her strategy is, player 2 will cooperate with probability ϵ*. Call this belief* p_1^ϵ*. Then* $p_1^\epsilon(\lambda, 1-\lambda) := (\epsilon, 1-\epsilon)$*. Since* $||p_1^A - p_1^\epsilon||_\infty \leq \epsilon$*, Proposition 3.5.3 tells us that the best response to this belief is* ϵ*-safe. We shall confirm that this is true.*

Let $(\mu, 1-\mu)$ *be player 1's strategy such that* $\mu \in (0, 1]$*. Then her expected utility is*

$$\begin{aligned} E[u(\mu, 1-\mu, p_1^\epsilon(\mu, 1-\mu))] =& \frac{3}{4}(\mu)(\epsilon) + 0(\mu)(1-\epsilon) + 1(1-\mu)(\epsilon) + \frac{1}{4}(1-\mu)(1-\epsilon) \\ =& \frac{1}{4}(1-\mu) + \frac{3}{4}\epsilon \end{aligned}$$

The expected utility of defecting, $(0, 1)$*, is*

$$\begin{aligned} E[u(0, 1, p_1^\epsilon(0, 1))] =& \epsilon + \frac{1}{4}(1-\epsilon) \\ =& \frac{1}{4} + \frac{3}{4}\epsilon \end{aligned}$$

Since $\mu > 0$*, we have*

$$E[u(0, 1, p_1^\epsilon(0, 1))] < E[u(\mu, 1-\mu, p_1^\epsilon(\mu, 1-\mu))].$$

Thus, the best response to p_1^ϵ *is* $(0, 1)$*, which is safe. Hence, it is also* ϵ*-safe.*

Now that we have considered the ϵ-safety of policy-conditioned beliefs, we move on to consider how to encourage social welfare maximising strategies.

3.6 Cooperation Inducing Beliefs

In Section 3.5, we proved that we could take a policy-conditioned belief that is cooperating-promoting and still maintain the safety property through uncertainty

about whether the opponent faced is, in fact, an adversary. Now, we must develop a policy-conditioned belief that we know will create cooperative behaviour in sequential social dilemmas. Here, we consider two-player matrix SSD games.

Let the strategy of player i be $(\alpha, \bar{\alpha})$, where $\alpha \in [0,1]$ is the intended probability of cooperating in the next round and $\bar{\alpha} \in [0,1]$ is the probability of cooperating for all subsequent rounds. We use this format for its simplicity; to achieve the cooperation-inducing property of a policy-conditioned belief. Since the opponent's strategy can change over time, $\bar{\alpha}$ can be considered as the expected average future strategy of their opponent. Similarly, the current and future strategies of their opponent are denoted by $(\beta, \bar{\beta})$. For discounted future returns, define the *expected returns* $V_i : \Sigma_i \times \Sigma_i \times \Sigma_{-i} \times \Sigma_{-i} \to \mathbb{R}$ as

$$V_i((\alpha, \bar{\alpha}), (\beta, \bar{\beta})) := E[u_i(\alpha, \beta)] + \sum_{t=1}^{n-1} \gamma^t E[u_i(\bar{\alpha}, \bar{\beta})],$$

where $\gamma \in (0,1]$ is the discount factor. A policy-conditioned belief in the sequential game is a function $p_i : \Sigma_i \times \Sigma_i \to \Sigma_{-i} \times \Sigma_{-i}$.

Although players cannot see their opponent's mixed strategy, suppose that a player believes that their chosen strategy for the next round will change the strategy of their opponent for all subsequent rounds. This will, of course, depend on their chosen level of cooperation. For some $x \in (0,1]$, if player i chooses to cooperate with at least proportion x, then their opponent's future cooperation level will not decrease, and for cooperation less than x, their opponent's level of cooperation will not increase for future rounds. Let player i have such a policy-conditioned belief p_i^C, where C stands for cooperation-promoting, formally defined as

$$p_i^C(\alpha) := \begin{cases} (\beta, \beta^+) & \alpha \geq x \\ (\beta, \beta^-) & \alpha < x \end{cases}$$

for some threshold $x \in (0,1]$, where $\beta \leq \beta^+$ and $\beta \geq \beta^-$. Note, this belief has similarities with a belief that an opponent plays tit-for-tat. Both believe that an opponent will cooperate in the future if they cooperate now, and defect in the future if they defect now. However, here we have the additional condition that we imagine that the opponent can view the intended mixed strategies of the player. The willingness to invest or reject risk capital is also extended into the long-term.

Let us find the necessary conditions on p_i^C for cooperation to be a best response strategy against similar agents, and hence, be a cooperation inducing policy-conditioned belief, as required. For cooperation to occur, we require that defection

is not a best response, i.e. $\forall \alpha > 0$,

$$V_i((\alpha, \bar{\alpha}), p_i^C(\alpha)) \geq V_i((0, \bar{\alpha}), p_i^C(0)).$$

Proposition 3.6.1. *For any two-player matrix SSD and policy-conditioned belief p_i^C, where β, β^+, β^- satisfy*

$$\alpha\beta(R+P-S-T) + \alpha(S-P) + \sum_{t=1}^{n-1} \gamma^t(\beta^+ - \beta^-)[\bar{\alpha}(R+P-S-T) + T - P] \geq 0$$

for some $\bar{\alpha} \in [0,1]$, cooperation is a best response.

Proof. For cooperation to be a best response, we need that $V_i((\alpha, \bar{\alpha}), p_i^C(\alpha)) \geq V_i((0, \bar{\alpha}), p_i^C(0))$. So, we can write the expected returns of cooperating with positive probability in terms of payoffs as

$$\begin{aligned} V_i((\alpha, \bar{\alpha}), p_i^C(\alpha)) =& \alpha\beta R + \alpha(1-\beta)S + \beta(1-\alpha)T + (1-\alpha)(1-\beta)P + \\ & \sum_{t=1}^{n-1} \gamma^t[\bar{\alpha}\beta^+ R + \bar{\alpha}(1-\beta^+)S + \beta^+(1-\bar{\alpha})T + (1-\bar{\alpha})(1-\beta^+)P]. \end{aligned}$$

The expected return of defecting ($\alpha = 0$) is:

$$\begin{aligned} V_i((0, \bar{\alpha}), p_i^C(0)) =& \beta T + (1-\beta)P + \sum_{t=1}^{n-1} \gamma^t[\bar{\alpha}\beta^- R + \bar{\alpha}(1-\beta^-)S + \beta^-(1-\bar{\alpha})T \\ & + (1-\bar{\alpha})(1-\beta^-)P]. \end{aligned}$$

Now, by substituting these into the inequality we get

$$\alpha\beta(R+P-S-T) + \alpha(S-P) + \sum_{t=1}^{n-1} \gamma^t(\beta^+ - \beta^-)[\bar{\alpha}(R+P-S-T) + T - P] \geq 0,$$

as given. □

If $\beta(R+P-S-T) + (S-P) > 0$ (such as in Stag Hunt), then $E[u_i((\alpha, 1-\alpha), (\beta, 1-\beta))]$ is increasing in α. As such, full cooperation will be dominant, so i will play $\alpha = 1$. Otherwise (such as in Prisoner's Dilemma), they will play $\alpha = x$. In which case, x should be set to 1 to induce fully cooperative behaviour.

These beliefs can be summarised as those who believe their next action will affect opponent's future strategies such that they prefer to cooperate now to avoid a reduction in future utility. We call a strategy that follows this system p^C. Other

beliefs from the literature that are cooperation-promoting could also be substituted here, such as translucency Capraro & Halpern (2019).

3.7 Trade-Off Between Cooperation and Safety

In Sections 3.5 and 3.6, we described how to use policy-conditioned beliefs to promote both safety and cooperation, respectively. In this section, we will use these results to demonstrate that the two concepts are necessarily in tension, and how this tension disappears in the long run. To make this concrete, we will define the value of cooperation to be the value expected against the policy-conditioned belief p_i^C, that is:

$$V^C(\pi_i) = E[u_i(\pi_i, p_i^C(\pi_i))].$$

Similarly, we can think of the value of safety to be the value expected against the policy-conditioned belief p_i^A, that is:

$$V^A(\pi_i) = E[u_i(\pi_i, p_i^A(\pi_i))].$$

We will define $\overline{V}^C = \max_\pi\{V^C(\pi_i)\}$ to be the optimal cooperative value and $\overline{V}^A = \max_\pi\{V^A(\pi_i)\}$ to be the optimal safe value. Here, we assume for simplicity that defection is both the safe and adversarial strategy here, such as in Prisoner's Dilemma and Stag Hunt[5]. For the sake of simplicity, we will also assume that $\beta = \beta^- = 0$ and $x = 1$, which would be the case for cooperation-promoting policies that are perfectly safe when defection is a safe strategy and cooperation is socially optimal.

Proposition 3.7.1. *For any two-player matrix SSD game, let π_i be an ϵ-safe policy. Let α_t be the probability π_i cooperates in round t against a cooperation-promoting belief and assume $E[r_t] \leq d\alpha_{t-1} + v_i$ for some constant $d > 0$, as is the case when $\beta = \beta^- = 0$. Then*

$$\overline{V}^C - V^C(\pi_i) \geq \frac{I}{T}\overline{V}^C - d\epsilon\frac{1 - \Phi_C^{I+1}}{1 - \Phi_C},$$

where $C = \frac{d}{P-S}$, $\Phi_x = \frac{1+\sqrt{1+4x}}{2}$, notated as such because Φ_1 is the golden ratio, and $I = \min\{\lceil -\log_{\Phi_C}(\epsilon)\rceil, T\}$.

Proof. Let π_i be ϵ-safe so $\epsilon = \overline{V}^A - V^A(\pi_i)$. The result follows from the fact that cooperation at each round gives a bound for safety, which we define to be $\tilde{\alpha}_k$ as

[5]This is not true for all SSDs. In Route Choice, the adversarial strategy is to cooperate when you know your opponent will cooperate.

follows:

$$\alpha_k \leq \frac{\epsilon + \sum_{t=0}^{k-1} E[r_t] - kv_i}{P - S} \leq \frac{\epsilon + \sum_{t=0}^{k-1} (d\alpha_{t-1} + v_i) - kv_i}{P - S} = \tilde{\alpha}_k.$$

By substitution we have:

$$\begin{aligned}
\tilde{\alpha}_k &= \frac{\epsilon + \sum_{t=0}^{k-1} (d\alpha_{t-1} + v_i) - kv_i}{P - S} \\
&= \frac{\epsilon + d\sum_{t=0}^{k-1} \alpha_{t-1}}{P - S} \\
&= \frac{\epsilon + d\alpha_{k-2} + d\sum_{t=0}^{k-2} \alpha_{t-1}}{P - S} \\
&= \frac{\epsilon + d\sum_{t=0}^{k-2} \alpha_{t-1}}{P - S} + \frac{d}{P - S}\alpha_{k-2}
\end{aligned}$$

Since we have the relationship $\tilde{\alpha}_k = \frac{\epsilon + d\sum_{t=0}^{k-1} \alpha_{t-1}}{P-S}$, we can substitute for $\tilde{\alpha}_{k-1}$ to get

$$\begin{aligned}
&= \tilde{\alpha}_{k-1} + \frac{d}{P - S}\alpha_{k-2} \\
&\leq \tilde{\alpha}_{k-1} + C\tilde{\alpha}_{k-2}
\end{aligned}$$

We can then show by induction that $\tilde{\alpha}_t \leq \epsilon\Phi_C^t$. In the base case, $\tilde{\alpha}_0 = \epsilon = \epsilon\Phi_C^0$. For the inductive case, assume it is true for $k-1$ or smaller. Note that $\Phi_C + C = \Phi_C^2$ by definition. Then,

$$\tilde{\alpha}_k \leq \tilde{\alpha}_{k-1} + C\tilde{\alpha}_{k-2} = \epsilon\Phi_C^{t-1} + C\epsilon\Phi_C^{t-2} = \epsilon(\Phi_C + C)\Phi_C^{t-2} = \epsilon\Phi_C^t$$

Thus, it holds for general k.

This inequality and $\alpha_k \leq 1$, allow us to bound $\overline{V}^C - V^C(\pi_i)$ by simply summing the expected rewards:

$$V^C(\pi_i) = \sum_{t=0}^{T} E[r_t] \leq d\sum_{t=0}^{T} \alpha_{t-1} + Tv_i \leq d\sum_{t=0}^{T} \min\{\epsilon\Phi_C^t, 1\} + \overline{V}^A.$$

Since the optimal behaviour against the cooperation promoting belief is deterministic cooperation, the difference from optimal behavior only occurs for t such

that $\epsilon\Phi_C^t < 1$. This happens when $t < -\log_{\Phi_C}(\epsilon)$. Note that if $-\log_{\Phi_C}(\epsilon)$ is bigger than T the second term of the min would never occur in the summation. Thus, this point where the summation starts using the second term rather than the first term occurs at $I = \min\{\lceil -\log_{\Phi_C}(\epsilon)\rceil, T\}$. So, we can finally plug this into the final inequality and use the geometric series formula to get:

$$V^C(\pi_i) \leq d\sum_{t=0}^{I} \epsilon(\Phi_C)^t + \overline{V}^C(1 - \frac{I}{T}) = d\epsilon\frac{1-\Phi_C^{I+1}}{1-\Phi_C} + \overline{V}^C(1-\frac{I}{T})$$

We multiply both sides by -1 and add $\overline{V}^C$ to get

$$\overline{V}^C - V^C(\pi_i) \geq \frac{I}{T}\overline{V}^C - d\epsilon\frac{1-\Phi_C^{I+1}}{1-\Phi_C},$$

as required. □

This result formally captures the trade-off between safety and cooperation by bounding from below the difference between a policy's value and that of the socially optimal policy. We can even find a policy such that this bound is tight if we have $E[r_t] = d\alpha_{t-1} + v_i$. The proof follows the same structure as Proposition 3.7.1.

Proposition 3.7.2. *For any two-player matrix SSD game, let $\bar{\pi}_i$ be an ϵ-safe policy, α_t be the probability $\bar{\pi}_i$ cooperates on round t against a cooperation-promoting belief, and assume $E[r_t] = d\alpha_{t-1} + v_i$ for some constant $d > 0$, as is the case when $\beta = \beta^- = 0$. Then,*

$$\overline{V}^C - V^C(\bar{\pi}_i) = \frac{I}{T}\overline{V}^C - d\epsilon\frac{1-\Phi_C^{I+1}}{1-\Phi_C},$$

where $C = \frac{d}{P-S}$, $\Phi_x = \frac{1+\sqrt{1+4x}}{2}$, notated as such because Φ_1 is the golden ratio, and $I = \min\{\lceil -\log_{\Phi_C}(\epsilon)\rceil, T\}$.

Proof. Let $\bar{\pi}_i$ be ϵ-safe so $\epsilon = \overline{V}^A - V^A(\bar{\pi}_i)$. Then, by construction of $\bar{\pi}$ we have

$$\alpha_k = \frac{\epsilon + \sum_{t=0}^{k-1} E[r_t] - kv_i}{P-S} = \frac{\epsilon + \sum_{t=0}^{k-1}(d\alpha_{t-1} + v_i) - kv_i}{P-S}.$$

By substitution we have:

$$\begin{aligned}
\alpha_k &= \frac{\epsilon + \sum_{t=0}^{k-1}(d\alpha_{t-1} + v_i) - kv_i}{P - S} \\
&= \frac{\epsilon + d\sum_{t=0}^{k-1}\alpha_{t-1}}{P - S} \\
&= \frac{\epsilon + d\alpha_{k-2} + d\sum_{t=0}^{k-2}\alpha_{t-1}}{P - S} \\
&= \frac{\epsilon + d\sum_{t=0}^{k-2}\alpha_{t-1}}{P - S} + \frac{d}{P - S}\alpha_{k-2}
\end{aligned}$$

Since we have the relationship $\alpha_k = \frac{\epsilon + d\sum_{t=0}^{k-1}\alpha_{t-1}}{P-S}$, we can substitute for α_{k-1} to get an upper bound of

$$\alpha_k = \alpha_{k-1} + \frac{d}{P - S}\alpha_{k-2} = \alpha_{k-1} + C\alpha_{k-2}.$$

We show by induction that $\alpha_t = \epsilon\Phi_C^t$. In the base case, $\alpha_0 = \epsilon = \epsilon\Phi_C^0$. For the inductive case, assume it is true for $k-1$ or smaller. Since $\Phi_C + C = \Phi_C^2$ by definition, we have

$$\alpha_k = \alpha_{k-1} + C\alpha_{k-2} = \epsilon\Phi_C^{t-1} + C\epsilon\Phi_C^{t-2} = \epsilon(\Phi_C + C)\Phi_C^{t-2} = \epsilon\Phi_C^t$$

Thus, $\alpha_t = \epsilon\Phi_C^t$ holds for all $t \in \mathbb{N}$. Now we bound $V^C(\bar{\pi}_i)$,

$$V^C(\bar{\pi}_i) = \sum_{t=0}^{T} E[r_t] = d\sum_{t=0}^{T}\alpha_{t-1} + Tv_i = d\sum_{t=0}^{T}\min\{\epsilon\Phi_C^t, 1\} + \overline{V}^A.$$

As in the proof of Proposition 3.7.1, we define $I = \min\{\lceil -\log_{\Phi_C}(\epsilon)\rceil, T\}$. Using the geometric series formula we can write this summation as:

$$V^C(\bar{\pi}_i) = d\sum_{t=0}^{I}\epsilon(\Phi_C)^t + \overline{V}^C(1 - \frac{I}{T}) = d\epsilon\frac{1 - \Phi_C^{I+1}}{1 - \Phi_C} + \overline{V}^C(1 - \frac{I}{T}).$$

We multiply both sides by -1 and add $\overline{V}^C$ to get

$$\overline{V}^C - V^C(\bar{\pi}_i) = \frac{I}{T}\overline{V}^C - d\epsilon\frac{1-\Phi_C^{I+1}}{1-\Phi_C}.$$

□

It is important to note from Propositions 3.7.1 and 3.7.2, that this trade-off, after a certain point, does not grow with T. Moreover, the return on cooperation value for small reductions in the optimal safety value grows very quickly, as the small loss in safety can effectively be reinvested at each successive time step as the policy receives gains from cooperation. Thus, in long iterated games, the optimal policy for this objective is nearly-rational for the designer to deploy into either fully adversarial settings or fully cooperative settings. In the next section, we take this core insight as the motivation for an algorithm that uses the ideas of reinvesting this risk capital in order to achieve high degrees of both safety and cooperation.

3.8 Investing in Cooperation

In SSDs, if an opponent is cooperative, surplus payoff above the value of the game can be reinvested for higher long-term gain. However, we also need to be able to invest safely to avoid exploitation.

To begin with, we give agents an adversarial policy-conditioned belief to avoid exploitation and maintain safety[6]. For any surplus capital gained, ϵ, we can deviate from our safe beliefs to be ϵ-safe and begin to reinvest to build trust with opponents. One such belief is $p_i^{\epsilon C} := (1-\epsilon)p_i^A + \epsilon p_i^C$, where the sequential adversarial belief is

$$p_i^A(\sigma) = \arg\min_{\sigma_{-i}\in\Sigma_{-i}} E[u_i(\sigma_i, \sigma_{-i})],$$

and β, β^+, β^- satisfy the conditions in Proposition 3.6.1. By Proposition 3.5.3, $p_i^{\epsilon C}$ is ϵ-safe.

Again, consider the two-player matrix SSDs of the form in Table 2.3. If $p_i^A(\sigma_i) = (0, 1, 0, 1)$, which is true for both Stag Hunt and Prisoner's Dilemma,

[6] We could forgo full safety in the first round here to increase cooperation between agents initially for better long-term rewards with only a small amount of risk, however, we will mainly focus on the safe version.

$\forall \sigma_i \in \Sigma_i$ - we can write the belief $p_i^{\epsilon C}$ as

$$p_i^{\epsilon C}(\alpha) = \begin{cases} (\epsilon\beta, \epsilon\beta^+) & \alpha \geq x \\ (\epsilon\beta, \epsilon\beta^-) & \alpha < x. \end{cases}$$

The condition for cooperation to occur is now:

$$\alpha(P-S) - \epsilon\alpha\beta(R+P-S-T) \leq \sum_{t=1}^{n-1} \gamma^t \epsilon(\beta^+ - \beta^-)[\bar{\alpha}(R+P-S-T)+T-P] \quad (3.2)$$

Naturally, for $\epsilon = 1$ this always holds. Ideally, we want to choose $\alpha, \beta, \beta^+, \beta^-$ such that this holds for the smallest possible ϵ so that we need to invest the least amount of risk to allow cooperation to be a best response to these beliefs.

Pseudocode for the Accumulating Risk Capital Through Investing in Cooperation is given in Algorithm 7. Following from results in (Ganzfried & Sandholm, 2012, Proposition 5.4), this algorithm is safe.

Algorithm 7: ARCTIC

Initialise $x \in [0,1]$, $\beta \in [0,1]$, $\beta^+ \leftarrow 1$, $\beta^- \leftarrow 0$;
$\epsilon \leftarrow 0$, $v_i \leftarrow$minimax value ;
for $t = 1$ *to* T **do**
  $p_i \leftarrow (1-\epsilon)p_i^A + \epsilon p_i^C$;
  $\pi_i \leftarrow \arg\max E[u_i(\sigma_i, p_i(\sigma_i))]$;
  i plays σ_i from π_i;
  $-i$ plays σ_{-i} from unknown π_{-i};
  $\epsilon \leftarrow \min(\epsilon + E[u_i(\pi_i, \sigma_{-i})] - v_i, 1)$
end

We can think of ϵ as representing the amount of risk capital we are willing to invest in an opponent. For safe play in a sequential game, we begin with no risk capital and play the minimax strategy, since we assume our opponent is adversarial. As the game goes on, the amount of risk capital will increase against non-adversarial opponents and we can safely invest such risk capital with the expectation of a return on the investment. Gradually, we build trust against similar opponents that reciprocate collaborative behaviours.

If our opponent has been cooperating in the past, then there is enough risk capital for ARCTIC to cooperate with such an opponent. However, if they then defect, the amount of risk capital drops and they are more likely to defect in the next round - similar to the way a tit-for-tat strategy punishes defections. If the risk

capital is high enough, then punishment of a defection is less common since ARCTIC has learnt to trust the good behaviour of its opponent. This mechanism stops the strategy from being exploited against adversaries whilst maximizing cooperation with allies.

The ARCTIC algorithm can be used in conjunction with a reinforcement learning algorithm such as Q-learning. The policy used in ARCTIC is dependent upon the policy-conditioned beliefs, whereas for Q-learning to incorporate and respond to the risk capital ϵ, it would need to be included in the state space. ARCTIC treats the risk capital as a random variable, but the policy is able to form stable cooperation with the opponent despite fluctuations to risk capital. Reinforcement learning algorithms that incorporate risk capital in their state space are unlikely to exhibit the same properties.

3.8.1 Matrix Games

The two-player matrix games we use are Prisoner's Dilemma, Stag Hunt, and Route Choice. In each of these games, the safe strategy is to defect[7]. Nevertheless, each game has a slightly different challenge to maximise social welfare.

In Stag Hunt, mutual cooperation gives the highest joint utility and is a Nash equilibrium; however, it is not stable. If either player makes a mistake and defects, then the best response is to defect since mutual defection is also a Nash equilibrium. Thus, the challenge is that players have to learn to trust each other's intentions of cooperating, even when the actions are noisy. Prisoner's Dilemma is the most renowned social dilemma. It is difficult to solve since mutual cooperation maximises joint payoff, but the only Nash equilibrium is mutual defection. To maximise payoff in Route Choice, each player needs to play a strategy that mixes between cooperation and defection in order to coordinate their route choice. In fact, mutual cooperation in Route Choice gives the lowest payoff to each player. As with Prisoner's Dilemma, defection is a strictly dominant strategy, so achieving strategy coordination is hard. The Route Choice game can represent any of the real-world social dilemmas found in transport: route choice, mode of transport, and departure times.

To test the algorithm's performance, we simulated an ARCTIC agent for 100 rounds of each game where its opponent followed either a simple strategy or best responded to their policy-conditioned beliefs. The payoff matrices for these games can be found in Table 2.3. The strategies played against were tit-for-tat (T4T) and

[7]This is not true of all social dilemmas. For example, cooperation is a safe strategy in the game of Chicken.

pure defector (Adv), with the policy-conditioned belief opponents following either ARCTIC or the cooperation inducing belief p^C.

Figures 3.1, 3.3, and 3.5 show the cooperation levels and risk capital ϵ for these simulations. The parameters used in the setup were a random action noise 5%, $\bar{\alpha} = \alpha$, and a discount factor of $\gamma = 0.9$. The results were then averaged over 200 runs and shown with 95% confidence intervals.

3.8.1.1 Prisoner's Dilemma

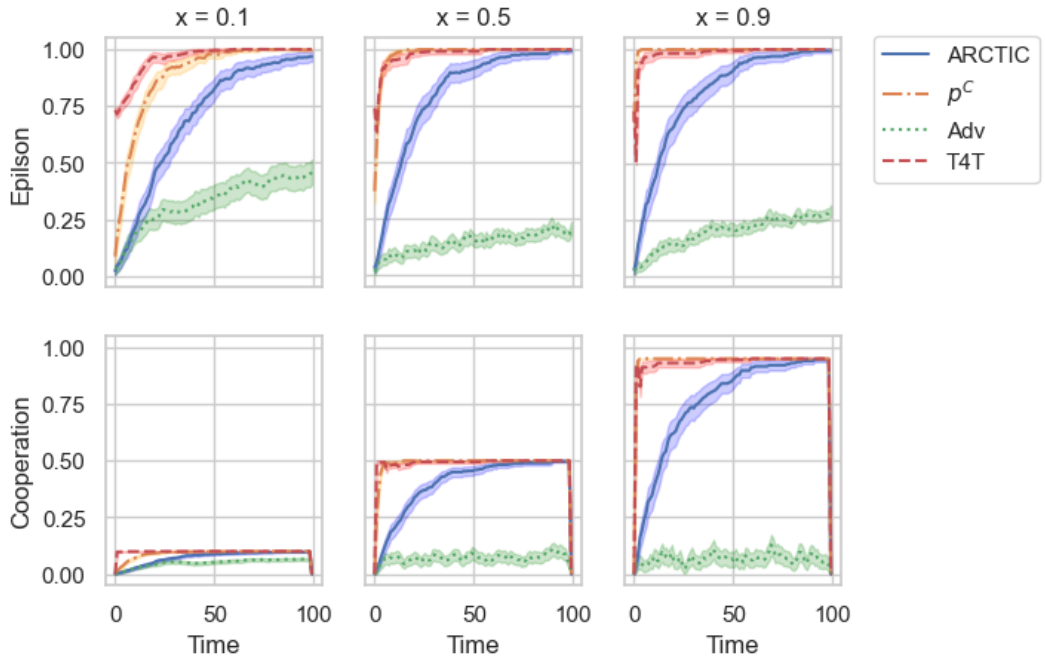

Figure 3.1: Simulations of ARCTIC playing different opponents in 100 rounds of Prisoner's Dilemma with various x. Against the adversarial strategy, ARCTIC does not learn to cooperate as there is not enough risk capital gained from their interactions. When two ARCTIC players interact, the risk capital slowly builds over time for all x as they both play cautiously. Note that on these graphs it shows that ϵ increases over time on average. The reason for this is when Adv randomly chooses to cooperate and ARCTIC defects it wins surplus payoff. This behaviour is averaged over the 200 runs, we do not see than ϵ increases over time in the individual runs, it is only that the expected number of Adv cooperation is increasing over time.

In Prisoner's Dilemma (Figure 3.1), the ARCTIC agent quickly learns to cooperate at rate x against the cooperation incentivised T4T and p^C players. When playing itself, the amount of risk capital, ϵ, increases more gradually since they are

learning to trust more cautiously than a T4T or p^C player. Against the adversarial player, the cooperation levels are very low, and thus the ARCTIC agent maintains the safety property. The play in Prisoner's Dilemma is unchanged by parameter β - inferred from Proposition 3.6.1 and confirmed through simulations - so the results shown are for a fixed value of $\beta = 0.5$. The values for x were chosen to show the range of behaviour by selecting low (0.1) and high (0.9) values, as well as a mid range value (0.5).

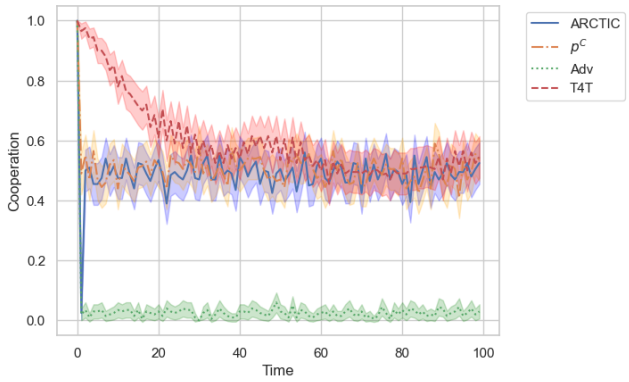

Figure 3.2: Cooperation of tit-for-tat playing 100 rounds of Prisoner's Dilemma. Here, the ARCTIC opponent has $x = 0.5$. When tit-for-tat plays against itself, it cooperates less on average over time due to the random action noise causing a cycle of punishments (of defection) after an accidental defection. Tit-for-tat mimics the same level of cooperation as the ARCTIC agent it plays against, here at 0.5 since this is the ARCTIC agent's x parameter, and therefore is ARCTIC's best response. A tit-for-tat agent will be more cooperative in an environment with noisy actions against an ARCTIC agent with $x > 0.5$ than it is against itself.

Let us compare the performance of ARCTIC with tit-for-tat play. Figure 3.2 shows the cooperation of tit-for-tat during the tournament. It is able to defect against Adv, which is the best-response play for this opponent. However, when it plays against itself, due to the random noise in the action, once a tit-for-tat player has defected, the players continue in a cycle of punishing each other's previous defection. This means that they achieve only an average of around 50% cooperation at round 100. In contrast, ARCTIC's cooperation levels improve over time as it learns to invest earned risk-capital and build trust with its cooperation reciprocating opponents. A similar behaviour is seen when playing against ARCTIC and p^C, where the initial cooperative behaviour is quickly reduced to only 50% cooperation.

Tit-for-tat and ARCTIC certainly exhibit similar properties such as reciprocation and forgivingness. However, they do have discernible differences. For in-

stance, ARCTIC will outperform tit-for-tat in noisy environments. Also, ARCTIC does not have the property that it will begin be cooperating in the first round, as this will not uphold the safety property for all SSDs.

To achieve cooperation using ARCTIC, we need an environment that is noisy, i.e. that there is some positive probability that players will make an error when choosing their action. If this was not the case, then the ARCTIC agents would always choose to defect when playing against themselves. A noisy environment makes it more difficult to achieve higher long term payoffs when using a tit-for-tat strategy, but it actually improves the payoffs for ARCTIC. The noise in the environment makes the cooperation reached by ARCTIC agents more stable, since they have learned the behaviour of their opponent and are willing to forget about defections that may be mistakes. More complex environments are often noisy, so this is a useful property for application to other multi-agent domains.

3.8.1.2 Stag Hunt

Similarly, in Stag Hunt (Figure 3.3) the ARCTIC agent learns to cooperate with all but the adversary. For x and β large enough to satisfy condition (3.2), ARCTIC learns to fully cooperate by the end of the 100 rounds of play. Cooperation is much easier to achieve in Stag Hunt than in Prisoner's Dilemma since defect is not a dominant strategy. Against the adversary, not enough risk capital is collected for the best response to be cooperation, which preserves the safety of the strategy.

Since the level of cooperation does change depending on β in Stag Hunt, we included three different β parameters. The same x parameters were chosen as the Prisoner's dilemma setting, in order to show a low, middle, and high value over the range $[0, 1]$. From Figure 3.3, we see that the value of β has more of an effect when the x value is small. When both of the x and β values are small, the ARCTIC agent only manages high levels of cooperation against the tit-for-tat agent. When the x parameter is high, ARCTIC performs best; it is able to maintain high cooperation against ARCTIC, p^C, and tit-for-tat quickly, as well as having a low rate of cooperation against the adversarial agent.

In Figure 3.4, we see that the tit-for-tat strategy chooses the correct levels of cooperation when they play against p^C and Adv, always or none respectively. However, when playing against itself, we see the same behaviour as in Prisoner's Dilemma where they end up punishing their opponent's last defection once an error has been made by either player. The tit-for-tat strategy increases it's proportion

of cooperation with ARCTIC over time, since ARCTIC prefers to cooperate with opponents that are willing to return the cooperation.

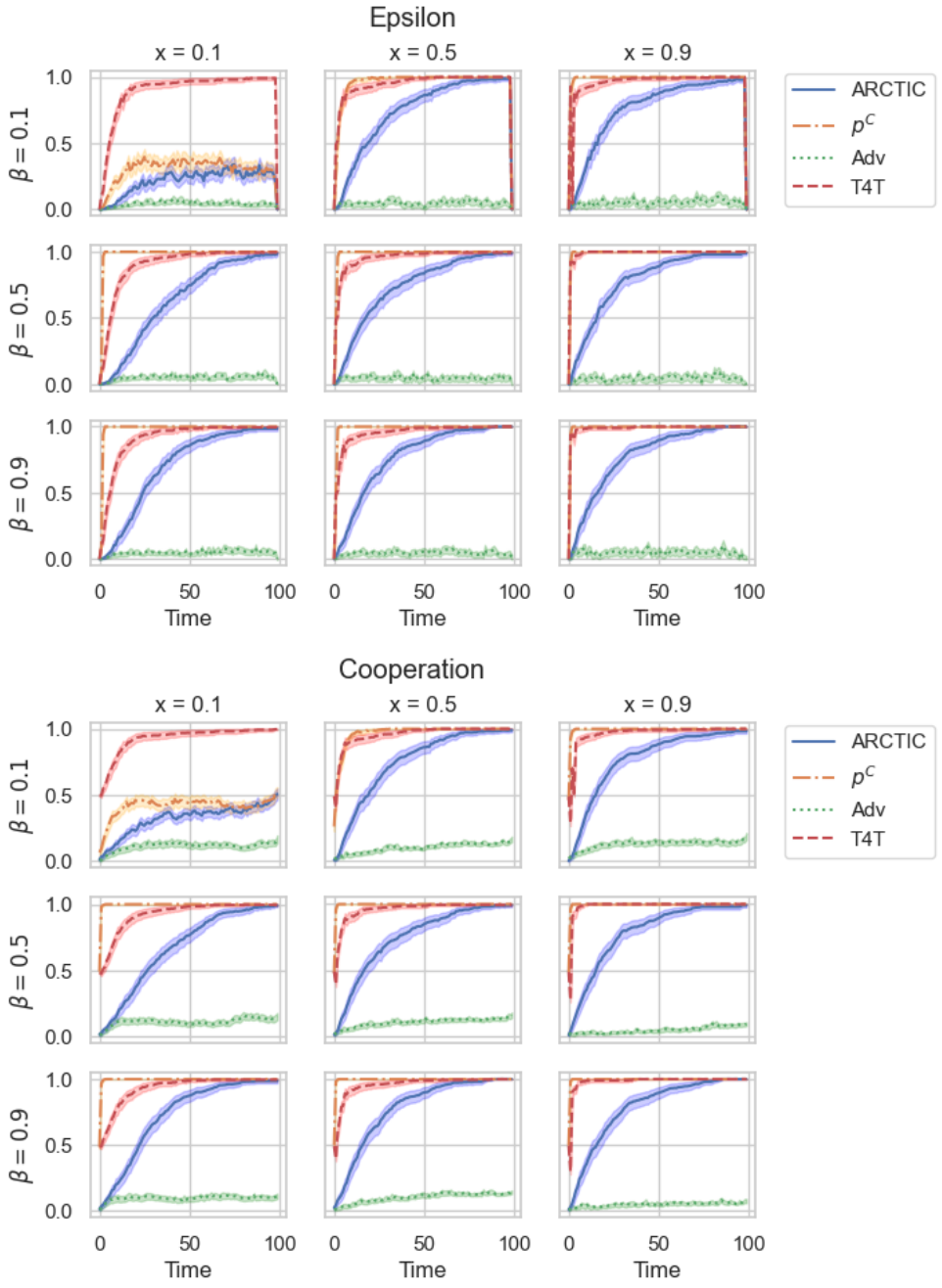

Figure 3.3: Simulations of 100 rounds of Stag Hunt whilst playing the ARCTIC strategy against 5 different opponent strategies. For cooperation to occur here, we need x and β to be large enough to satisfy condition 3.2.

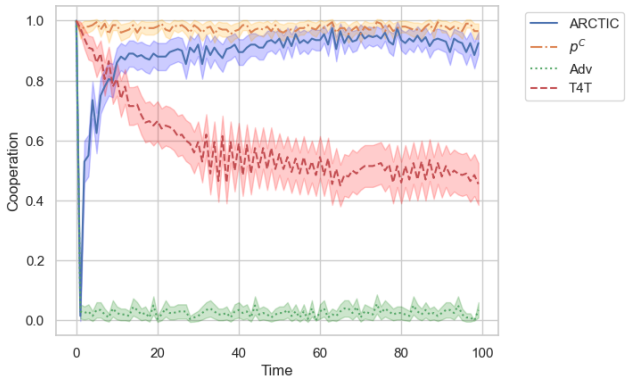

Figure 3.4: Cooperation of tit-for-tat playing 100 rounds of Stag Hunt. Here, the ARCTIC opponent has $x = 0.5$. Here, tit-for-tat is able to maintain high levels of cooperation against ARCTIC and p^C. However, when playing itself, the cooperation reduces on average over time until only 50% cooperation.

3.8.1.3 Route Choice

For the Route Choice game (Figure 3.5), the ARCTIC agent learns to cooperate with all strategies except the adversarial agent for values of $x < 0.5$. ARCTIC cooperates at the highest levels with the p^C agent and behaves the same against itself as it does with the tit-for-tat agent. For values of $x \geq 0.5$, the risk capital ϵ increases over time, but the best response is still for the agent to defect. The best-response of ARCTIC is not affected by β so this is set to 0.5.

Tit-for-tat is not designed for Route Choice's coordination problem. However, we include the tit-for-tat performace in Figure 3.6 for completeness. Again, we see that tit-for-tat decreases its cooperation over time rather than building it up the way ARCTIC does.

One noticeable difference between the Route Choice simulations and the Prisoner's Dilemma and Stag Hunt simulations, is that when playing against the adversarial opponent, ϵ increases over time for larger values of x. This happens since, for the adversarial player to minimise its opponent's utility, it must play defect if the opponent defects and cooperate is the opponent cooperates. In the other games, the adversarial agent always defects. In our simulation, the adversarial agent cannot predict when the opponent will cooperate; thus, the implementation is that they will always defect since the Nash equilibrium and safe strategy for their opponent is to defect. Consequently, when the adversarial agent makes a mistake, ARCTIC's ϵ increases at a larger rate than it decreases, i.e. when ARCTIC makes a mistake.

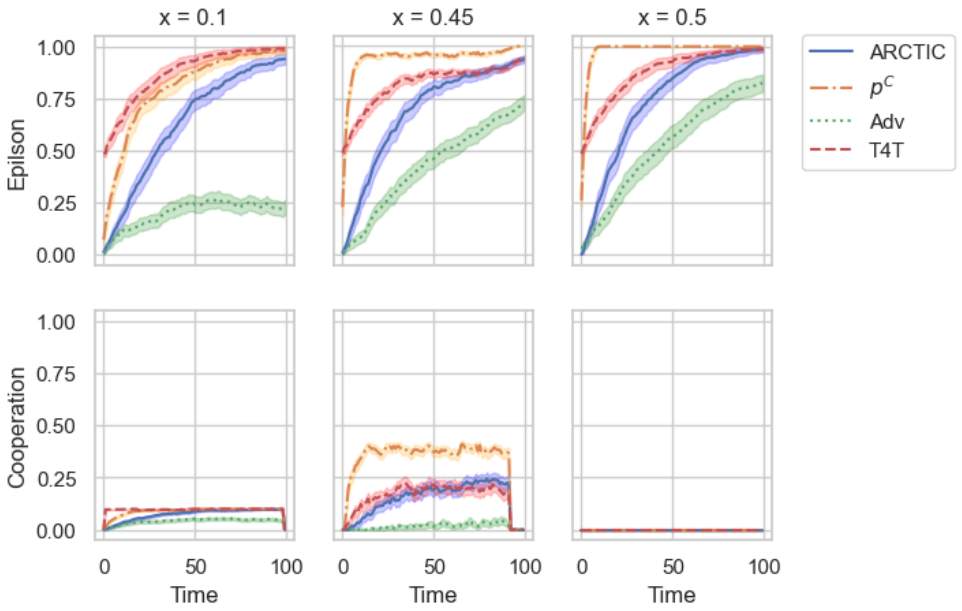

Figure 3.5: Simulations of 100 rounds of Route Choice whilst playing the ARCTIC strategy against 5 different opponent strategies. For cooperation to occur here, we need that x is not too large as the payoff is maximised by mixing between both strategies equally. Strategies mixing greater than or equal to 0.5 are strictly worse off than those cooperation less than 50% of the time.

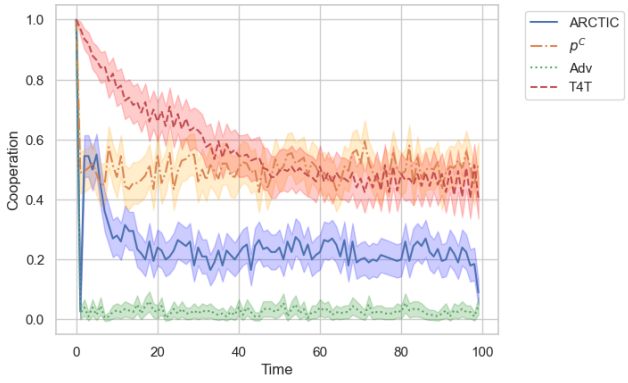

Figure 3.6: Cooperation of tit-for-tat playing 100 rounds of Route Choice. Here, the ARCTIC opponent has $x = 0.45$. Due to the random action noise in the simulation, although tit-for-tat is able to cooperate early on, it only decreases in cooperation over time as it is not able to build any trust with their opponent.

The minimum payoff of 0 is rarely achieved, so, the overall trend of ϵ is increasing. If we included a more complex adversarial agent implementation, this would not occur.

Here, we have used ARCTIC for payoffs normalised on $[0, 1]$ as seen in Table 2.3. For games where payoffs have a range greater than 1, the ϵ update step should be

$$\epsilon \leftarrow \min(\epsilon + \frac{1}{K}(E[u_i(\pi_i, \sigma_{-i})] - v_i), 1),$$

where K is the greatest difference between possible payoffs in order to normalise risk capital to be in $[0, 1]$.

3.9 Multi-Agent Reinforcement Learning

SSDs are often subgames of more complex multiplayer games. These types of games are of particular interest in multi-agent RL due to the difficulty of learning co-operative policies. In more complex games where teamwork is required, it is only sequences of actions that create cooperative or selfish behaviours. Thus, we consider long-term behaviours in order to capture the nature of the agents.

In this section, we include some examples of MARL environments with two players to show ARCTIC's capability in a reinforcement learning domain. For MARL environments with greater than two agents, the theory should be easily extendable to games with more than two players. First, the safety property of the Risk What You've Won in Expectation (Algorithm 1) from (Ganzfried & Sandholm, 2012, Proposition 5.4) should be extended to games with more than two players. Some interesting environments to try and apply ARCTIC too would be Harvest and Cleanup (Hughes *et al.* (2018)), where coordination and cooperation is required between more than two players to achieve high rewards. These environments are often used in the MARL literature as complex environments with an element of sequential social dilemmas Eccles *et al.* (2019a); Jaques *et al.* (2019); McKee *et al.* (2020).

We assume that the minimax value, v, of the game can be determined. An opponent can therefore detect whether the agent is cooperating by measuring whether their rewards are at least the value of the game. The level of cooperation is now $x_t := \sum_{k=0}^{t} \gamma^{t-k} \mathbb{1}_{r_k > v}$.

Let player i's cooperation inducing policy-conditioned belief p_i^C, be defined as

$$p^C(\pi_i) := \begin{cases} (\pi_j, \pi_j^+) & x_t^i \geq x \\ (\pi_j, \pi_j^-) & \text{otherwise} \end{cases}$$

where $V_{\pi_i^{p^C}, \pi_j^+}(s) > V_{\pi_i^{p^C}, \pi_j^-}(s)$ for some threshold $x \in (0, 1]$.

If cooperation level x_i^t is above a certain threshold, then the opponent will behave cooperatively. Otherwise, they will act in their own self-interest. To train

agents with these cooperative beliefs, we can adapt the reward functions of their opponents as such:

$$r_t^{-i} \leftarrow \begin{cases} r_t^i + r_t^{-i} & x_t^i \geq x \\ r_t^{-i} & \text{otherwise} \end{cases}$$

To train an agent with a policy-conditioned belief, it can be trained in an environment where those beliefs are true and transferred into the standard environment for deployment.

We trained distributed asynchronous advantage actor-critic (A3C) Mnih *et al.* (2016) agents on Prisoner's Dilemma, Stag Hunt, and Route Choice environments. The A3C algorithm was chosen since it is employed in much of the MARL social dilemmas literature Eccles *et al.* (2019b); Hughes *et al.* (2018); Jaques *et al.* (2019); Lerer & Peysakhovich (2019). It is more suitable to environments with a large number of agents, but applying it to smaller problems first is useful to understand its properties before application to larger environments.

Agent policies were trained with the policy-conditioned beliefs p^C and ARCTIC where $x = 0.5$ and $\bar{\alpha} = \alpha$. Agents without beliefs were the baseline agent and the adversarial (Adv) agent. The neural network consists of two fully connected layers of size 32 and a Long Short Term Memory (LSTM) recurrent layer Gers *et al.* (1999). This network architecture was taken from Jaques *et al.* (2019). The learning rate for baseline, adversarial, and p^C agents was 0.001. For ARCTIC, the learning rates were 0.00007 for Prisoner's Dilemma and 0.0001 for Stag Hunt and Route Choice, chosen from Hughes *et al.* (2020) and using learning rate provided by the Ray library respectively. The entropy coefficient was 0.01, the base value provided for A3C in Ray. The state space for the ARCTIC agents was a onehot encoded ϵ value. Each agent was trained on 3 different random seeds and results are average across these policies for 300 rollouts.

3.9.1 Prisoner's Dilemma

The difficulty of playing Prisoner's Dilemma with a generic multi-agent RL algorithm is that defection is a strictly dominant strategy and, thus, usually converge to defecting. This means that a mechanism for agents to cooperate must be used to promote cooperation, which leaves them open to exploitation. By using ARCTIC here, the agent still acts rationally by playing a best-response, but it does so with respect to their policy-conditioned belief.

Table 3.1 shows the cumulative rewards for players after 100 rounds of playing Prisoner's Dilemma. The baseline agent performs poorly against itself due to its

inability to cooperate whereas the ARCTIC agent cooperates with itself some of the time and achieves a more socially optimal outcome. With the p^C player, ARCTIC cooperates at higher levels than with itself. Against the adversary, ARCTIC achieves close to the value of the game on average. The adversarial and baseline agent learn a similar policy to each other and so perform similarly in the tournament.

Table 3.1: Scores for trained agents (ARCTIC, A3C baseline, cooperating-promoting belief, and adversarial) playing a tournament of 100 rounds of Prisoner's Dilemma.

	Baseline	ARCTIC	p^C	Adv
Baseline	25.01, 25.01	25.54, 24.83	70.29, 9.91	25.00, 25.01
ARCTIC	24.83, 25.54	34.12, 34.12	57.84, 46.72	24.82, 25.56
p^C	9.91, 70.29	46.72, 57.84	55.21, 55.21	9.89, 70.34
Adv	25.01, 25.00	25.56, 24.82	70.34, 9.89	25.00, 25.00

Figure 3.7 shows how the level that the ARCTIC agent cooperates over the 100 rounds of Prisoner's Dilemma agents the different opponents.

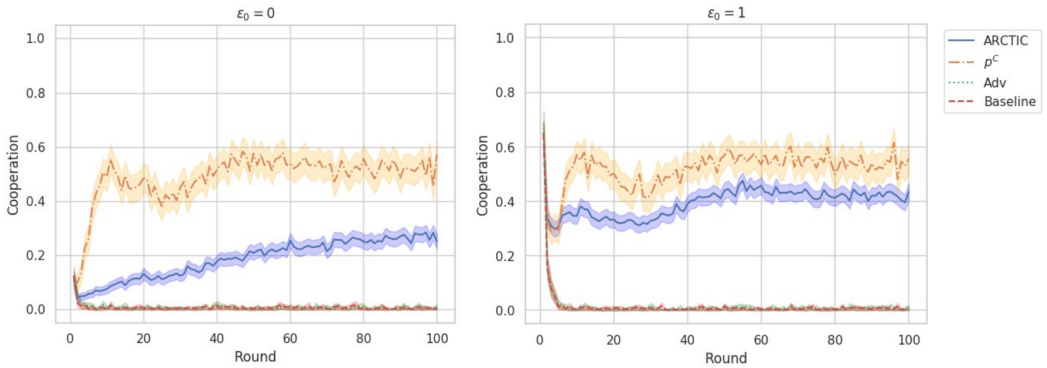

Figure 3.7: Cooperation of ARCTIC against different opponents over 100 rounds of Prisoner's Dilemma with initial risk capital of 0 (left) and 1 (right), where $x = 0.5$. Against the adversarial and baseline opponents, ARCTIC learns to not cooperate. Whereas, when playing against p^C, it cooperates around x. Against itself, ARCTIC cooperates at lower levels, but is increasing.

To try and improve the level of cooperation with itself, the initial level of risk capital can be increased to improve the cooperativeness with only increased risk in the first round of play. From Table 3.2, we see that this leads to a better outcome when played against itself without risking much against defectors.

Table 3.2: Scores for trained agents playing a tournament of 100 rounds of Prisoner's Dilemma when the ARCTIC agent has an initial risk captial of $\epsilon_0 = 1$.

	Baseline	ARCTIC	p^C	Adv
ARCTIC	24.65, 26.05	45.35, 45.35	56.23, 50.46	24.64, 26.06

3.9.2 Stag Hunt

When playing Stag Hunt, multi-agent RL algorithms learn to cooperate more effectively than Prisoner's Dilemma, but this leaves them exploitable to adversaries. Thus, the challenge for agents is to protect themselves and play safely.

In Table 3.3, the scores for agents in the Stag Hunt tournament can be found. Here, the baseline agent performs well against itself but achieves a poor score against adversaries. On the other hand, ARCTIC agents achieve the value of the game against adversaries.

Table 3.3: Scores for trained agents (ARCTIC, A3C baseline, cooperating-promoting belief, and adversarial) playing a tournament of 100 rounds of Stag Hunt.

	Baseline	ARCTIC	p^C	Adv
Baseline	99.53, 99.53	78.61, 94.31	99.78, 99.34	0.13, 74.84
ARCTIC	94.31, 78.61	27.02, 27.02	93.52, 78.17	24.85, 25.31
p^C	99.34, 99.78	78.17, 93.52	98.66, 98.66	0.34, 74.43
Adv	74.84, 0.13	25.31, 24.85	74.43, 0.34	25.02, 25.02

To encourage ARCTIC to cooperate more against itself, we can introduce a small amount of risk in the initial round. From Figure 3.8, we see that ARCTIC achieves low levels of cooperation against themselves, but when $\epsilon_0 = 1$, they are willing to cooperate significantly more.

When ARCTIC is equipped with a positive initial risk capital of $\epsilon_0 = 1$, the ARCTIC agent is able to achieve better outcomes against all players except for the adversary, where it achieves marginally less than when $\epsilon_0 = 0$. See the results for this tournaments in Table 3.4.

Table 3.4: Scores for trained agents playing 100 rounds of Stag Hunt when the ARCTIC agent has an initial risk capital of $\epsilon_0 = 1$.

	Baseline	ARCTIC	p^C	Adv
ARCTIC	95.19, 81.92	73.36, 73.36	94.56, 82.03	24.60, 25.82

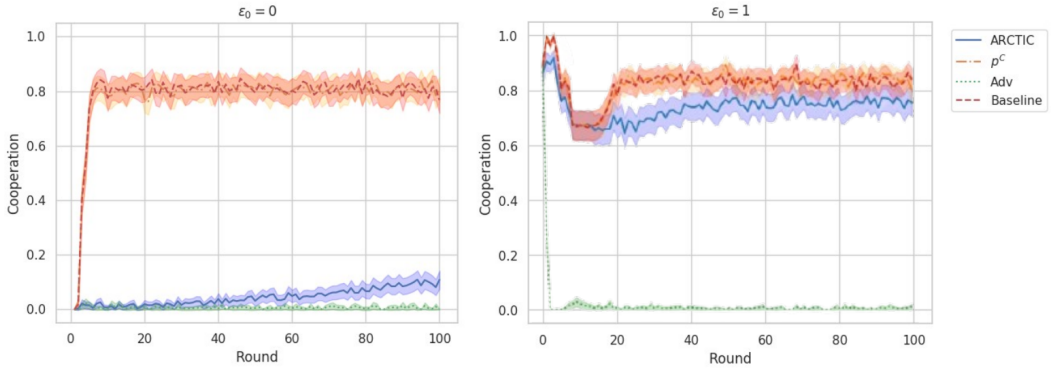

Figure 3.8: Cooperation of ARCTIC against different opponents over 100 rounds of Stag Hunt when starting with risk capital of 0 (left) or 1 (right). In both cases, ARCTIC cooperates the most with p^C and baseline agents and the least with adversaries.

3.9.3 Route Choice

In Route Choice, multi-agent RL algorithms learn to defect, since this is the safe strategy and a Nash equilibrium. However, joint utility is maximised when players both have a low level of cooperation; they coordinate their route choices. In these simulations, we set the value of x to be 0.2. If we set the x value too high here, then it would discourage cooperation as shown in Figure 3.5.

The scores for agents in the Route Choice tournament can be found in Table 3.5. Here, the baseline agent performs well against themself, but achieves a poor score against adversaries. On the other hand, ARCTIC agents achieve the value of the game against adversaries.

Table 3.5: Scores for trained agents (ARCTIC, A3C baseline, cooperating-promoting belief, and adversarial) playing a tournament of 100 rounds of Stag Hunt.

	Baseline	ARCTIC	p^C	Adv
Baseline	50.00, 50.00	50.03, 49.98	54.32, 47,89	50.09, 49.9
ARCTIC	49.98, 50.03	50.06, 50.06	51.36, 51.30	50.02, 50.07
p^C	47.89, 54.43	51.30, 51.36	51.63, 51.63	47.87, 54.32
Adv	49.98, 50.03	50.07, 50.02	51.36, 51.30	50.02, 50.02

Figure 3.9 shows the level of cooperation of the ARCTIC agent over the rounds of Route Choice for both $\epsilon_0 = 0$ and $\epsilon_0 = 1$.

The effects of increasing the initial level of risk capital in Route Choice is less apparent than the Prisoner's Dilemma and Stag Hunt games. This is because we are

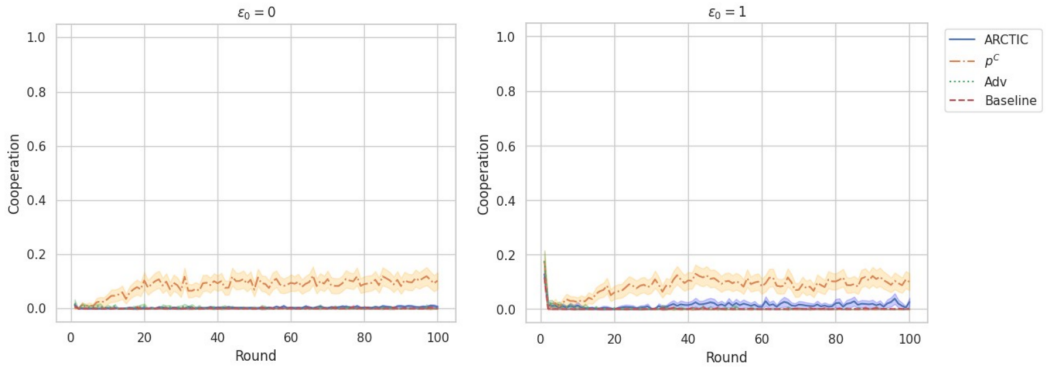

Figure 3.9: Cooperation of ARCTIC against different opponents over 100 rounds of Route Choice when starting with risk capital of 0 (left) or 1 (right). In both cases, ARCTIC cooperates the most with p^C. To cooperate with other ARCTIC agents, it requires an initial positive risk capital to begin cooperation. It does not cooperate with adversaries and the baseline.

aiming for a low level of cooperation here, only 20%. However, this does mean that it does not lose much payoff compared to when $\epsilon_0 = 0$, against the adversarial and baseline players. Table 3.6 shows the scores for tournaments with ARCTIC agents that have $\epsilon_0 = 1$.

Table 3.6: Scores for trained agents playing 100 rounds of Route Choice when the ARCTIC agent has an initial risk capital of $\epsilon_0 = 1$.

	Baseline	ARCTIC	p^C	Adv
ARCTIC	49.96, 50.08	50.19, 50.19	51.20, 51.33	49.97, 50.15

3.10 Discussion

In summary, we studied the trade-off between cooperation and safety, first showing how to unify these two objectives in the formalism of policy-conditioned beliefs and then characterizing a trade-off between them. We find that small risks to safety can lead to large returns in cooperation. The concept of risk capital made this trade-off more intuitive; cooperation representing the compounding returns on investment. We use this intuition to build Accumulating Risk Capital Through Investing in Cooperation (ARCTIC) which exploits this trade-off by achieving safe cooperation. ARCTIC strategies were for the two-player games Prisoner's Dilemma, Stag Hunt, and Route Choice.

The stable cooperation in noisy environments exhibited by ARCTIC is simi-

lar to notions of trust, social capital, and reputation. This build-up of trust between the players is represented by the risk capital parameter. ARCTIC uses a model of direct reciprocity to achieve this, whereby the cooperative behaviour of their opponent is rewarded by an increase in cooperation. Indirect reciprocity is not currently observed, but an extension of ARCTIC to games with more than two players could assess the capacity for indirect reciprocity in adapted ARCTIC agents.

The ideas behind ARCTIC, keeping track of and investing risk capital, should be applicable to any multi-agent scenario where the population of opponent strategies is unknown, and cooperation is beneficial. Often when deploying an agent into a multi-agent system we only have control of only one of the agents involved in the interaction. This could occur in robot-robot interactions, such as self-driving cars where cars may be operated by different companies, or in human-robot interactions, such as a robot negotiating with a human. In these domains, cooperation can be useful to promote long-term performance. For example, cars cutting each other off to improve their journey time could result in traffic jams, wasting everyone's time. Acting cooperatively in a negotiation could preserve trust in the relationship, resulting in better deals for both sides in future negotiations. These applications require solutions that both promote cooperation and are rational for an individual designer to deploy.

Although Prisoner's Dilemma, Stag Hunt, and Route Choice are quite simplified, they have long served as a good testing ground to understand whether a technique can work in principle before addressing the challenges of scale. The ARCTIC algorithm can already be directly applied to MARL environments with only 2 agents by combining it with any algorithm, including A3C as we have demonstrated. Possibly the most difficult part of applying ARCTIC to real-world domains is that you must know the value of the game. It is possible to try and obtain this value or estimate it, but this is also application dependent.

Being able to cooperate while maintaining approximate safety allows us to design agents that individual developers would want to use out of their own self-interests. This is a promising development but leaves open questions that will be important in more complex environments. For instance, when there are many styles of successful cooperative strategies, agent designers would need to coordinate on a particular style of cooperation, or build their agents to be adaptive to the techniques of other agents. In addition, although our method protects the agent against adversaries, agents who want to minimise our reward, it does not protect the agent against exploitative agents, agents which want to maximise their own reward at the cost of our reward. This becomes more complex when combined with the coordina-

tion problems, as different coordination solutions could have different payouts which must be somehow distinguished from exploitative strategies. Extending the ideas of risk capital to these settings is left to future work.

There are also interesting challenges in scaling ARCTIC to larger environments. Our method is currently reliant on both knowing the expected minimax value and a clear notion of cooperation. In larger environments, these are both less accessible. To extend ARCTIC to these settings these environment features would either have to be estimated or the reliance on these features would have to be removed. Thus, the work has less of a practical application to traffic as it stands, but could impact future work in the area. Ultimately, addressing these issues could lead to algorithms for multi-agent cooperation, which individual developers would find worth the risk.

CHAPTER 4

Bounding the Inefficiencies of Route Control

4.1 Introduction

Reducing traffic congestion has been a goal of many cities for decades, with benefits including reduced travel times and decreased air pollution. With the prevalence of automatic route planners such as GPS navigation, Google Maps, Waze, etc., congestion could be improved by intelligent routing systems. The capacity for routing systems to control the flow of congestion is only increasing. Autonomous vehicle development will allow route planners to control the exact routing of vehicles with minimal input from drivers. The efficiency of using navigation applications as socially beneficial route planners is currently an open problem Dafoe *et al.* (2020).

In distributed Artificial Intelligence, congestion games Rosenthal (1973) have emerged as a reference model to analyse the inefficiency of traffic flows, with important implications for the design of better road systems Wu *et al.* (2019). In congestion games, self-interested players travel between origin and destination nodes in a network, choosing paths that minimise their travel time. Players' strategies constitute a Nash, or user, equilibrium when they have no incentive to unilaterally deviate, and we want to compare these equilibria against the total travel times, yielding the players' social welfare. The most often used measure of inefficiency is the Price of Anarchy (PoA) Koutsoupias & Papadimitriou (1999), which compares the worst Nash equilibrium routing with that of the optimal flow.

While Nash equilibria are important predictors, it is also well-known that their assumptions on individuals' rationality are frequently not met in practice. In large transportation networks, it is often the case that individuals have incomplete knowledge of the network (see, e.g., the bounded rationality approaches in Acemoglu *et al.* (2018) or Meir & Parkes (2018)) and rely on personal route planners to figure

out their optimal route. This intermediate perspective, where competing controllers act on the same network, has been surprisingly overlooked in the congestion game literature.

4.2 Contributions

In this chapter, we study intelligent routing systems that act as distributed controllers on a traffic network and analyse their impact on the overall efficiency. We expand upon ideas introduced in Chapter 3 by addressing the social dilemmas caused by route choices of automated path planners in a traffic network.

We develop a two-level game, called the *network control game*, where route planners have control over the routing choices of the nonatomic congestion game. Each route planner controls a finite predetermined fraction of the total traffic with the goal of minimising the travel time incurred by that fraction only. We show that network control games are potential games and have an essentially unique equilibrium.

It is essentially a distributed resource allocation problem with separable welfare functions, where the resource sets are edges on a network and the strategies of a player must correspond to their given origin and destination pair, i.e., on a nonatomic congestion game. From now on, we refer to the players of the nonatomic congestion game as vehicles and the route controllers as route planners.

We then study equilibrium efficiency, showing that the Price of Anarchy is highest when the allocation of vehicles to route planners is (approximately) proportional. We also give Price of Anarchy bounds over polynomial cost functions, depending on the polynomial degree and the number of controllers and give a MARL example to show that this Price of Anarchy occurs in practice. Finally, we allow vehicles to choose their route planner, showing that the equilibrium reached has the highest total cost.

4.3 Literature Review

Congestion games are a class of games in game theory first proposed by Rosenthal Rosenthal (1973), utilised in research for modelling the behaviours of network systems. These networks were initially studied in the transportation literature by Wardrop Wardrop (1952) who established the conditions for a system equilibrium to exist when all travellers have minimum and equal costs. Their applications have increased to include many other situations that can be modelled with selfish players

routing flow in a network, for example, machine scheduling or communication networks Orda *et al.* (1993a), as well as physical systems such as bandwidth allocation Haïkel Yaïche *et al.* (2000) or electrical networks Ibars *et al.* (2010). However, their main application is for congestion games is transportation Fisk (1980); Sheffi (1985); Yao *et al.* (2019).

Games, where the utilities of all players can be described with a single function, are called potential games Monderer & Shapley (1996), and these are, in fact, equivalent to congestion games. A useful property of potential games is that they always admit a pure Nash equilibrium. Finding a pure Nash equilibrium in an exact potential game is a PLS-complete problem Fabrikant *et al.* (2004). However, improvement paths Monderer & Shapley (1996) converge at equilibrium for all potential games.

The Price of Anarchy Koutsoupias & Papadimitriou (1999) was proposed as a measure of inefficiency representing the cost ratio of the worst possible Nash equilibrium to the social optimum. The Price of Anarchy in network congestion games is a phenomenon that is independent of network topology Roughgarden (2003). The *biased* price of anarchy Meir & Parkes (2018) compares the cost of the worst equilibrium to the social optimum, when players have "wrong" cost functions, i.e. differing from the true cost due to biases or heterogeneous preferences.

The work in this chapter connects to a number of research lines in algorithmic game theory focusing on the quality of equilibria in congestion games and resource allocation, and the research in distributed artificial intelligence studying planning and control with boundedly rational agents.

From the point of view of distributed control, an important related model is Stackelberg routing games, where a portion of the total flow is controlled centrally by a "leader", while the "followers" play as selfish vehicles. Stackelberg routing was first proposed by Korilis *et al.* (1997), characterising which instances are optimal. Roughgarden Roughgarden (2004) found the ratio between worst-case and best-case costs in these games, and the impact of Stackelberg routing on the PoA has also been established for general networks Bonifaci *et al.* (2010). Single-leader Stackelberg equilibria in congestion games have been looked at, and it is known that they cannot be approximated in polynomial time Castiglioni *et al.* (2019b). Multi-leader Stackelberg games are, instead, largely unexplored in this context Castiglioni *et al.* (2019a). Our approach features multiple leaders, but not Stackelberg-like "followers", which impacts our results on the PoA.

Much of the transport literature is aimed at reducing congestion and increasing efficiency in traffic networks focuses on introducing tolls Karakostas & Kol-

liopoulos (2009); Meir & Parkes (2016); Sandholm (2002). However, information design has more recently been used to reduce congestion.

Information design, which is closely related to our approach, has more recently been considered as a mechanism to reduce congestion Acemoglu *et al.* (2018); Meir & Parkes (2018); Roman & Turrini (2019). The information constrained variant of nonatomic congestion game was first introduced to show that information could cause vehicles to change their departure times in such a way as to exacerbate congestion rather than ease it Arnott *et al.* (1991). The set of outcomes that can arise in equilibrium for some information structure is equal to the set of Bayes correlated equilibria Bergemann & Morris (2013). Das et al. Das *et al.* (2017) considered an information designer seeking to maximise welfare and restore efficiency through signals using information design. Tavafoghi and Teneketzis Tavafoghi & Teneketzis (2017) showed that the socially efficient routing outcome is achievable through public and private information mechanisms. Moreover, Ikegami et al. Ikegami *et al.* (2020) consider a centralised mediator to recommend routing to users taking into account their preferences for incomplete information games. Routing mediators have been used as a tool to coordinate agent's behaviour in games to find stable and efficient outcomes through their ability to collect information about multiple players Rozenfeld & Tennenholtz (2007). Our work differs from the private information design literature. In our model, the route planners control the routing rather than provide signals, and there are multiple agents attempting to optimise 'group' welfare.

A similar game is that of splittable congestion game, first studied in the context of communication networks Orda *et al.* (1993b). Here, each player in the congestion game assigns a weight to the possible strategies which arises when considering coalitions of players in nonatomic congestion games. The bounds on the Price of Anarchy for splittable congestion games is known for polynomial cost functions Roughgarden & Schoppmann (2015) has the same bound as when there exist an infinite number of route planners in a network control game.

Network control games can be seen as resource allocation games where the resources are edges in a network and the potential function is given by the total cost of all players' travel times. Distributed resource allocation problems aim to allocate a set of resources for optimal utilisation, such as distributed welfare games Marden & Wierman (2013) and cost-sharing protocols Chen *et al.* (2010). A recent survey of game-theoretic control of networked systems highlights the other major advancements applications Wu *et al.* (2019).

Finally, the related distributed welfare games Marden & Wierman (2013) utilise game-theoretic control for distributed resource allocation where the distri-

bution rule is chosen to maximise the welfare of resource utilisation. Different distribution rules can be compared by their desirable properties such as scalability, the existence of Nash equilibria, Price of Anarchy, and Price of Stability. In this context, protocols have been studied to improve equilibria of network cost-sharing games Chen *et al.* (2010), while Hao *et al.* (2018) studied welfare-optimising designers under full and partial control.

4.4 Network Control Games

Suppose that the routing choices of vehicles in a nonatomic congestion game $\mathcal{M}$ are controlled by a set of route planners R, where each route planner aims at minimising the total travel cost of the (nonempty) portion of vehicles assigned to them.

The way in which the route planners have control over the routing choices is by choosing which knowledge set is available to each player. Thus, the route planners control the demand for each knowledge type within the fraction of flow they control. For instance, a navigation app would give its users a choice between multiple routes; drivers have incomplete knowledge of the network available to them. Autonomous vehicles may not give their passengers a choice of route. In this case, the knowledge set would contain only the route that the autonomous vehicle follows. In the preliminaries (Chapter 2), we denoted a player from population $i \in N$ of knowledge type $k \in K_i$ as a tuple (i, k). In this chapter, we will refer to such a player as k, for simplicity of notation.

Suppose that we have a nonatomic congestion game $\mathcal{M}$ where the routing choices of all cars is dictated a set of route planners R. A route planner $r \in R$ has a set of drivers which they control N_r, where $\emptyset \neq N_r \subseteq N$. Each car has their route chosen by a single route planner. The route planners wish to minimise the total cost of travel for the drivers, i.e., minimise journey times for those players which they control. Let the size of each population $i \in N$ controlled by $r \in R$ be denoted d_i^r, where $\sum_{r \in R} d_i^r = d_i$ and $d_i^r = 0$ for $i \notin N_r$.

We can view the game as an information design problem where a player r partitions populations in N_r into sets of information types $\boldsymbol{K}_r = (K_i)_{i \in N_r}$, to minimise the social cost of N_r. Thus, a route planner chooses the information type demands d_k such that $\sum_{k \in (\boldsymbol{K}_r)_i} d_k = d_i^r$, $\forall i \in N_r$.

Let the strategy space for route planners be D_{κ_r} where κ_r is the set of all irredundant information sets K_i, for any $i \in N_r$. Moreover, for any $\boldsymbol{d} \in D_{\kappa_r}$ and $\forall i \in N_r$, we have $\sum_{k \in \kappa_r} d_k \mathbb{1}_{K_i}(k) = d_i^r$. Let the combined strategy space of all route planners be denoted as D_κ, where κ is the set of all possible irredundant information

types for populations in N.

Now, we can define a network control game.

Definition 4.4.1 (Network Control Game)**.** *A* ***network control game*** *is a tuple*

$$(\mathcal{M}, R, (N_r)_{r\in R}, (d_i^r)_{i\in N_r}, (D_{\kappa_r})_{r\in R}),$$

where $\mathcal{M}$ is a nonatomic congestion game, R is the set of route planners, N_r is the population controlled by $r \in R$, d_i^r is the demand of population i controlled by r, and D_{κ_r} is the strategy space of r.

Let the *share of control* of route planner r be $\frac{\sum_{i\in N_r} d_i^r}{\sum_{i\in N} d_i}$. If a route planner has a share of control equal to one, then we say it has *full control* of the game. The control of r over a population i is defined as $\frac{d_i^r}{d_i}$. If $\forall r \in R$ and $\forall i \in N$, the control of r over population i is $\frac{1}{|R|}$, then we say that the game is *proportional*.

Observe now that the outcome of all route planners' strategies $\boldsymbol{d} := (\boldsymbol{d}^r)_{r\in R}$ leads to an ICUE $\boldsymbol{x}$ in the underlying game. Given this, the cost function of a route planner $C_r : D_{\kappa_r} \to \mathbb{R}_{\geq 0}$ is defined as $C_r(\boldsymbol{d}^r, \boldsymbol{d}^{-r}) := \sum_{k\in\kappa_r} C_k(\boldsymbol{x}) d_k^r \ \forall r \in R$, where $\boldsymbol{x}$ is the ICUE from $(\boldsymbol{d}^r, \boldsymbol{d}^{-r})$. Here, the notation $-r$ means all players in R excluding r i.e., $\{1, 2, .., r-1, r+1, ..., |R|\}$. For instance, we use $C_r(\boldsymbol{d})$ and $C_r(\boldsymbol{d}^r, \boldsymbol{d}^{-r})$ interchangeably.

An outcome $\boldsymbol{d}$ is then a Nash equilibrium of the network control game if $\forall r \in R$ we have $C_r(\boldsymbol{d}) \leq C_r(\boldsymbol{d}', \boldsymbol{d}^{-r}) \ \forall \boldsymbol{d}' \in D_{\kappa_r}$. We can show the existence of Nash equilibria in network control games by showing that these are, in fact, exact potential games.

Theorem 4.4.2. *A network control game is an exact potential game for potential Φ defined as*

$$\Phi(\boldsymbol{d}) := \sum_{e\in E} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}) d\boldsymbol{z},$$

where $\boldsymbol{x}$ is the ICUE formed from $\boldsymbol{d}$.

Proof. Consider a unilateral deviation $\hat{\boldsymbol{d}}^r$ of route planner r from an outcome $\boldsymbol{d}$ with respective ICUE profiles $\hat{\boldsymbol{x}}$ and $\boldsymbol{x}$.

$$\Phi(\hat{\boldsymbol{d}}^r, \boldsymbol{d}^{-r}) - \Phi(\boldsymbol{d}) = \sum_{e\in E} \int_0^{f_e(\hat{\boldsymbol{x}})} c_e(\boldsymbol{z}) d\boldsymbol{z} - \sum_{e\in E} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}) d\boldsymbol{z} \tag{4.1}$$

Since the deviation from $\boldsymbol{x}$ to $\tilde{\boldsymbol{x}}$ only involves edges in κ_r, rewrite equation 4.1 as

$$\begin{aligned}
&= \sum_{k \in \kappa_r} \sum_{e \in s_k} \Big[\int_0^{f_e(\hat{\boldsymbol{x}})} c_e(\boldsymbol{z}) d\boldsymbol{z} - \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}) d\boldsymbol{z} \Big] \\
&= \sum_{k \in \kappa_r} \Big[C_k(\hat{\boldsymbol{x}}) - C_k(\boldsymbol{x}) \Big] \\
&= C_r(\hat{\boldsymbol{d}}^r, \boldsymbol{d}^{-r}) - C_r(\boldsymbol{d})
\end{aligned}$$

Thus, the function Φ is an exact potential function. By definition, the network control game is an exact potential game. □

Since we have an exact potential game with nondecreasing edge-costs, Corollary 4.4.3 follows directly from the literature e.g., (Acemoglu *et al.*, 2018, Theorem 1).

Corollary 4.4.3. *For every network control game, there exists a Nash equilibrium and it is essentially unique.*

As the network control game is an exact potential game, we know that all of the results that hold for congestion games will also be true here, e.g. Roughgarden (2003); Milchtaich (2006). Nonetheless, these games will provide an insight into how the distribution of vehicle route planners will affect traffic equilibria, a novel contribution to the literature.

We now define the Price of Anarchy of a network control game as

$$PoA = \frac{\max_{\boldsymbol{d} \in NE} C(\boldsymbol{d})}{\min_{\boldsymbol{d} \in D_\kappa} C(\boldsymbol{d})}$$

where NE is the set of Nash equilibria.

We note that our setup can be extended to incorporate vehicles that are not fully controlled by a route planner, e.g., by allowing route planners that give full information sets to their populations. However, we only consider vehicles following a route planner directly, to more easily classify the best and worst-case equilibria from full route control of populations. We also note that, for any strategy distribution in a (information constrained) nonatomic congestion game, we can, without loss of generality, only consider pure strategy equivalents. Thus, we consider the case where all information sets chosen by the route planners contain only one strategy. As such, the profile set by the route planners $\boldsymbol{d}$ has a deterministic associated ICUE $\boldsymbol{x}$.

4.5 Inefficiency of Route Controllers

To see how the network control game creates inefficient equilibria, consider what happens as we change the number of route planners in a proportional game. First, suppose that a route planner has full control of the game, then all vehicles follow the same route planner. Thus, the route planner has an objective function equal to the social cost of the system: $C_r(\boldsymbol{d}) = \sum_{k \in \kappa_r} C_k(\boldsymbol{x}) d_k = \sum_{k \in K} C_k(\boldsymbol{x}) d_k = SC(\boldsymbol{x})$. As such, the case with $|R| = 1$, will implement the socially optimal routing allocation.

Now, as we increase the number of route planners, the demand of the population controlled by a single player decreases. As $|R| \to \infty$, and since the game is proportional, we have that $d_{N_r} \to 0$, $\forall r \in R$. As we now have an infinite number of agents controlling a negligible amount of flow, we are back to a simple nonatomic congestion game. This occurs since $C_{-r}(\boldsymbol{d}^r, \boldsymbol{d}^{-r}) = C_{-r}(\boldsymbol{d}^{-r})$ $\forall \boldsymbol{d}^r \in D_\kappa$, when the proportion of control of r is negligible. The Price of Anarchy of the game is now the same as in its underlying nonatomic congestion game. Thus, if the number of route planners controlling the flow in a proportional network control game is greater than 1, which is true by definition, there is an inefficient equilibrium if the nonatomic congestion game admits one.

Proposition 4.5.1. *A proportional network control game has a Price of Anarchy greater than 1 if, and only if, the Price of Anarchy of the congestion game it controls is strictly greater than 1.*

Proof. If the Price of Anarchy of the network control game is strictly greater than 1, then there is an incentive to choose suboptimal routing at the Nash equilibrium. As the number of route planners in the game tends to infinity we approach the congestion game. As such, the suboptimal routing exists in the congestion game and so the Price of Anarchy for the congestion game is strictly greater than 1 too.

If the Price of Anarchy of the congestion game is strictly greater than 1, then we know that at the UE, there exist suboptimal selfish routing of drivers. Let the Nash equilibrium of routing be $\boldsymbol{D}$ and the social optimum be $\boldsymbol{C}$. Since the game is proportional, all route planners have the same strategy space. Thus, $\forall r \in R$,

$$\begin{aligned} C_r(\boldsymbol{D}) &\leq C_r(\boldsymbol{C}_r, \boldsymbol{D}_{-r}) \\ SC(\boldsymbol{C}) &\leq SC(\boldsymbol{D}) \\ SC(\boldsymbol{C}) &\leq SC(\boldsymbol{C}_r, \boldsymbol{D}_{-r}) \end{aligned}$$

We can write this as a two-player (r and $-r$) normal form game with the actions C_r, D_r, C_{-r}, and D_{-r}. This, combined with the inequalities above, indicates that

the payoffs comply with the conditions required for a social dilemma, as stated in the preliminaries (see Table 2.2). As such, the Price of Anarchy is strictly greater than 1. □

Since the Price of Anarchy is independent of network topology, we can use the Pigou example to illustrate the inefficiency of having multiple route planners. We assume that the cost functions are polynomial with degree p. To begin, let us consider linear cost functions, i.e., $p = 1$.

Figure 4.1: A Pigou network with two route planners. Left: The strategy for route planner $r \in R$ where $x_r \in [0, d^r]$. Right: The edge costs where $p \geq 0$ where x_1 and x_2 are defined from the flows in Left.

Example 2. *Suppose we have a total flow of 1 and two route planners 1 and 2 with respective population control of* d^1 *and* $d^2 = 1 - d^1$ *on a Pigou network.*

Each route planner must solve the following minimisation problem to find their equilibrium routing defined by the variable x_r *for* $r \in \{1, 2\}$*, as defined in Figure 4.1.*

$$\min_{x_r} x_r(x_1 + x_2) + (d^r - x_r)$$

subject to $0 \leq x_r \leq d^r$*. This gives us the Lagrangian function (where* $s \in \{1, 2\}$*,* $s \neq r$*):*

$$L(x_r, \lambda_1, \lambda_2) = x_r(x_r + x_s) + d^r - x_r - \lambda(d^r - x_r).$$

The corresponding Karush-Kuhn-Tucker conditions are:

$$\frac{\delta L}{\delta x_r} = 2x_r + x_s - 1 + \lambda = 0 \quad \text{and} \quad \lambda\frac{\delta L}{\delta \lambda} = \lambda(x_r - d^r) = 0.$$

First, consider the case where $x_r = d^r$*. Since* $\lambda \geq 0$*, we must have* $d^r \leq \frac{1}{2}(1 - x_s)$*. Route planner* r *plays selfishly by routing along the bottom edge only if their control is small. Now, suppose that* $x_r \neq d^r$ *and* $x_s \neq d^s$*. The solution here is* $x_1 = x_2 = \frac{1}{3}$*. The last possible case is where* $x_s = d^s$*, and similarly, this occurs when* $d^s \leq \frac{1}{2}(1 - x_r)$*. The optimal routing of splitting the vehicles equally between routes only occurs when there is one route planner with full control.*

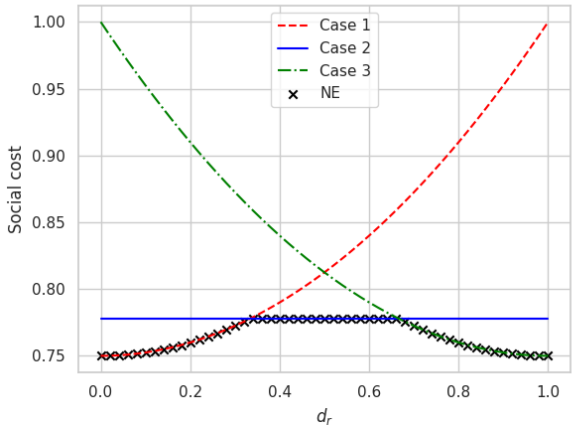

Figure 4.2: The social cost of routing on the Pigou with 2 route planners and $p = 1$.

The social cost of the three cases is shown in Figure 4.2. Note that the social cost of the equilibrium is highest in the set of demand that are close to, or exactly, proportional. Remember that the social cost of the nonatomic congestion game is 1, so the route planner are more efficient at routing socially than the vehicles of a nonatomic congestion game are. However, there is still a significant difference between the worst-case and best-case equilibria here.

Now consider what happens when we have a polynomial cost function of where $p \geq 1$.

Example 3. *As before, we let each route planner* $r \in \{1, 2\}$ *solve the following minimisation problem:*

$$\min_{x_r} x_r(x_1 + x_2)^p + (d^r - x_r),$$

with the constraint of $0 \leq x_r \leq d^r$*. This optimisation problem gives us the following Lagrangian function:*

$$L(x_r, \lambda_1, \lambda_2) = x_r(x_r + x_s)^p + d^r - x_r - \lambda(d^r - x_r)$$

Again, we find the Karush-Kuhn-Tucker conditions:

$$\frac{\delta L}{\delta x_r} = (x_r + x_s)^p + p x_r(x_r + x_s)^{p-1} - 1 + \lambda = 0$$

$$\lambda \frac{\delta L}{\delta \lambda} = \lambda(x_r - d^r) = 0$$

When we solve these Karush-Kuhn-Tucker conditions we find similarities with the $p = 1$ *case. For* $p \geq 1$*, the three cases remain the same as* $p = 1$*. However, now the best response to* $x_r = d^r$ *is to route* $x_s = (1+p)^{-1/p}$*, and when* $x_r = x_s$*, we have* $x_r = (p2^{p-1} + 2^p)^{-1/p}$*. For example,* $p = 2$ *is shown in Figure 4.3.*

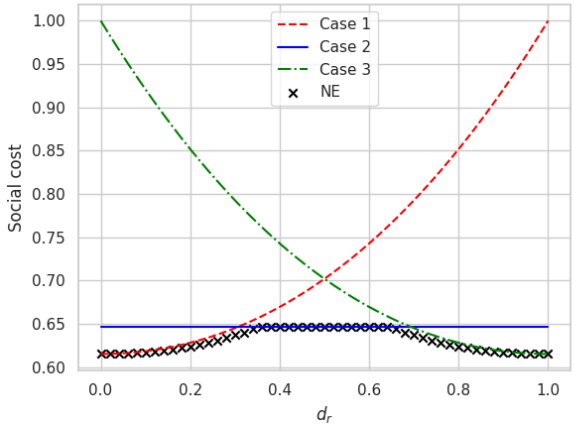

Figure 4.3: The social cost of routing on the Pigou with 2 route planners for $p = 2$.

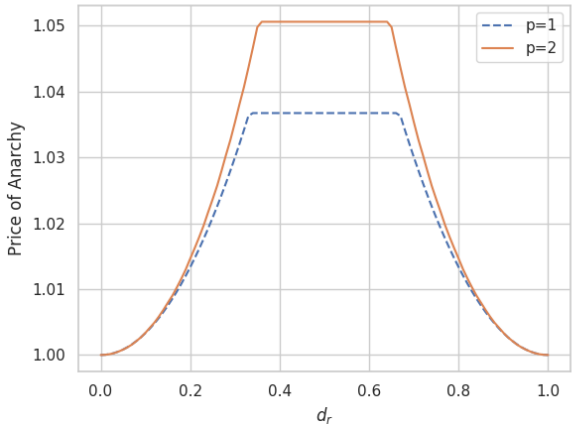

Figure 4.4: The Price of Anarchy of network control on the Pigou with 2 route planners and $p = 1$ and $p = 2$.

The social cost of the three cases is shown in Figure 4.3. It has distinct similarities with Figure 4.2, but with a different scale.

The comparison of the PoA for Examples 2 and 3 is shown in Figure 4.4.

From this figure, we can predict that the Price of Anarchy is increasing in p, a result which we will formalise later.

We will now analyse what happens to the Price of Anarchy when we introduce more route planners. Suppose there are three route planners controlling the flow on the same Pigou network. For simplicity, we will consider linear cost functions again, i.e., $p = 1$.

Example 4. *Consider the Pigou network with three route planners. As before, each route planner $r \in \{1, 2, 3\}$ performs a minimisation over their routing choice x_r. Since they choose their routing independently of one another, the same reasoning can be used to consider more populations. The optimal routing remains the same, but the effect of adding another selfish agent increases the worst possible cost. This can be seen in Figures 4.5 and 4.6, where same behaviour is seen compared to the share of control of the populations. The set of route planners is represented by $\{r, s, t\}$.*

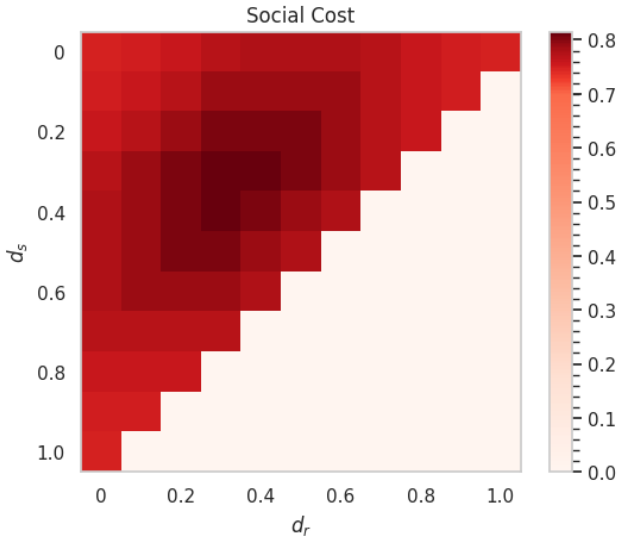

Figure 4.5: The social cost of routing on the Pigou network with $p = 1$ for a network control game with three route planners.

As with Examples 2 and 3, when the game is proportional it has the worst possible ICUE cost, therefore has the greatest Price of Anarchy. These behaviours are the same with two and three route planners, thus we can predict this to also be true for all $|R| > 2$.

Let us formalise the result that proportionality is linked to a high Price of Anarchy. Once again, since the Price of Anarchy is independent of network topology,

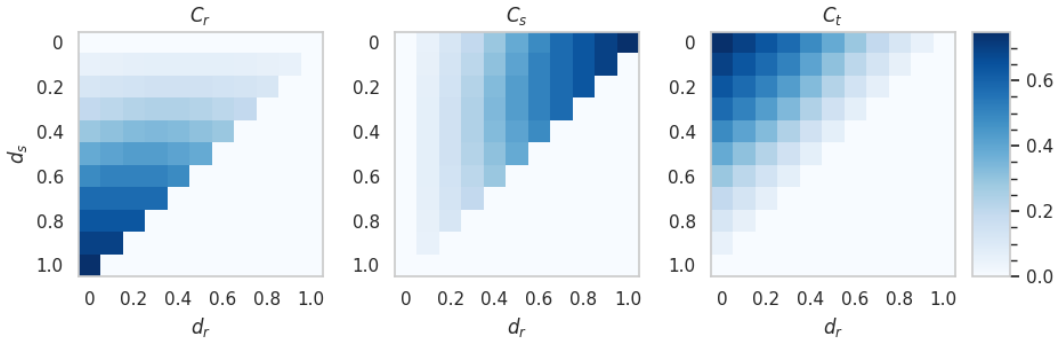

Figure 4.6: The route planner cost for a network control game on the Pigou network with three route planners and $p = 1$.

we can find the worst-case example of it using the Pigou example. Thus, Examples 2, 3, and 4 found the worst-case ratio of selfish route planners to fully cooperative route planners for 2 and 3 route planners, respectively.

Let the number of route planners be $R \in \mathbb{N}$. We can find the Price of Anarchy using the same method as the example for general R.

Proposition 4.5.2. *The Price of Anarchy of a network control game is highest when the game is proportional.*

Proof. To find the worst-case Price of Anarchy of route control, we want that no route planner is acting socially optimally. We can find the worst-case of routing on the Pigou example, since it is independent of topology Roughgarden (2003). Thus, we solve the minimisation problem:

$$\min_{0 \leq x_r \leq d_r} x_r(\sum_{s \in R} x_s)^p + (d^r - x_r).$$

To do so, we use the following Lagrangian function:

$$L(x_r, \lambda_1, \lambda_2) = x_r(\sum_{s \in R} x_s)^p + d^r - x_r - \lambda(d^r - x_r).$$

The corresponding Karush-Kuhn-Tucker conditions are:

$$\frac{\delta L}{\delta x_r} = (\sum_{s \in R} x_s)^p + p x_r(\sum_{s \in R} x_s)^{p-1} - 1 + \lambda = 0$$
$$\lambda \frac{\delta L}{\delta \lambda} = \lambda(x_r - d^r) = 0$$

For general $p \geq 0$ and $R \geq 2$, the three cases remain the same as Example

2. The best response to $x_r = d^r$ is to choose $x_s = (1+p)^{-1/p}$, and when $x_r = x_s$, we have $x_r = (pR^{p-1} + R^p)^{-1/p}$. For no route planner to choose the socially optimal routing in Pigou's example, each route planner must have a control of population i of at least $(pR^{p-1} + R^p)^{-1/p}$ and less than or equal to $1 - (pR^{p-1} + R^p)^{-1/p}$. For all R and p, $(pR^{p-1} + R^p)^{-1/p} \geq \frac{1}{R}$. As $R \to \infty$, $(pR^{p-1} + R^p)^{-1/p} \to \frac{1}{R}$. Thus, the worst-case equilibrium cost can be achieved through a proportional assignment of populations. □

The maximum social cost of Nash equilibria of the network control game also occurs for other distributions of route planner control. From Figures 4.2 and 4.5, we see that there is a set of population controls that maximise social cost existing around the proportional version of the game. This set is characterised by each route planner having a share of control of at least $(R^p + pR^{p-1})^{-1/p}$ for each population. For example, with linear cost functions and two route planners, each route planner must control at least $1/3$ of each population or for three route planners they must control $1/4$.

Theorem 4.5.3. *The worst-case Price of Anarchy for a network control game with R route planners and polynomial edge-cost functions at most degree p is*

$$\frac{1 - R\left(R^{p-1}(p+R)\right)^{-\frac{1}{p}} + R^{p+1}\left(R^{p-1}(p+R)\right)^{-\frac{p+1}{p}}}{1 - (p+1)^{-\frac{1}{p}} + (p+1)^{-\frac{p+1}{p}}}.$$

Proof. By Proposition 4.5.2, the worst-case equilibrium can be found when the game is proportional. Thus, we let each route planner, $r \in R$, solve the objective function

$$\min_{x_r} x_r\big(\sum_{s\in R} x_s\big)^p + d^r - x_r.$$

At the minimum, we have

$$\big(\sum_{s\in R} x_s\big)^p + x_r p\big(\sum_{s\in R} x_s\big)^{p-1} - 1 = 0.$$

Since the strategy spaces are symmetric and the game has an exact potential function, there exists a Nash equilibrium where each route planner plays the same strategy. The Nash equilibria of an exact potential game all have the same social cost so this instance is also the worst Nash equilibrium. Thus,

$$(Rx_r)^p + x_r p(Rx_r)^{p-1} - 1 = 0 \tag{4.2}$$

Equation 4.2 rearranges to give the strategy distribution $\forall r \in R$ as

$$x_r = (pR^{p-1} + R^p)^{-1/p} \tag{4.3}$$

The social cost of the worst-case Nash equilibrium (defined by 4.3) is

$$R^{p+1}(pR^{p-1} + R^p)^{-1-1/p} + 1 - R(pR^{p-1} + R^p)^{-1/p} \tag{4.4}$$

The social optimum of the game is where the total congestion on the bottom edge is $(p+1)^{-1/p}$, with a social cost of

$$(p+1)^{-1-1/p} + 1 - (p+1)^{-1/p} \tag{4.5}$$

This ratio of equation 4.4 to equation 4.5 gives us the result. □

For $R = 1$, by Theorem 4.5.3, the Price of Anarchy is 1. Thus, the system is efficient when a route planner has full control of all vehicles. As $R \to \infty$, Theorem 4.5.3 implies that the Price of Anarchy tends to that of the nonatomic congestion game it controls (Roughgarden (2003)):

$$\frac{(p+1)^{\frac{1}{p}+1}}{(p+1)^{\frac{1}{p}+1} - p}$$

Figure 4.7 plots the Price of Anarchy as a function of p for the network control games with varying R and p. The Price of Anarchy for the network control game is significantly better than that of the congestion game (where $R = \infty$) for a small number of route planners, for all $p > 0$. But as the number of route planners increases, the system gets more inefficient.

Table 4.1 summarises the PoA for small R and p.

Table 4.1: Price of Anarchy for network control game with R route planners and polynomial edge-cost functions.

Edge Costs	$R = 2$	$R = 3$	$R = 4$	$R = 5$
Linear	1.037	1.083	1.12	1.148
Quadratic	1.051	1.122	1.183	1.233
Cubic	1.058	1.143	1.221	1.288
Quartic	1.061	1.156	1.246	1.325

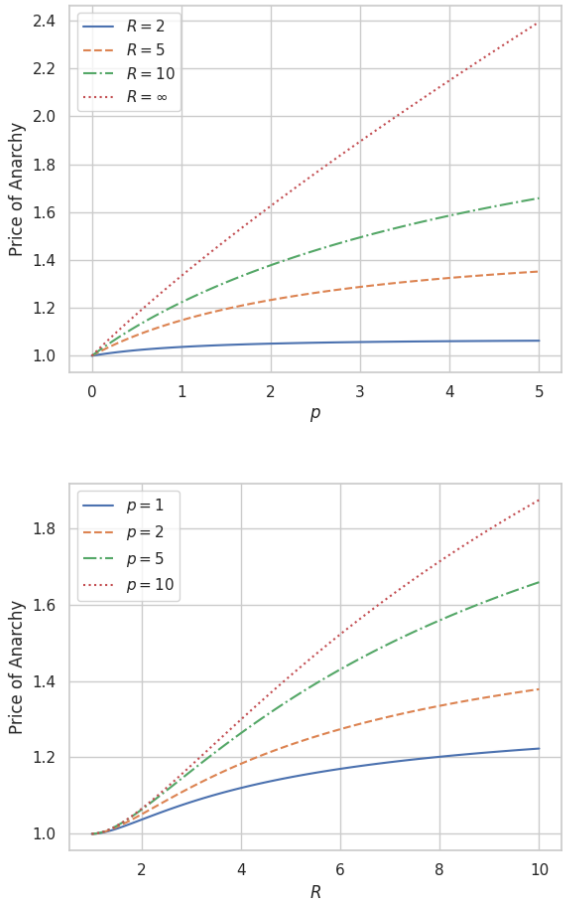

Figure 4.7: The Price of Anarchy for the network control game for various p and r.

4.6 Multi-Agent Learning Example

Now let us consider the application of this theory to multi-agent reinforcement learning. We will take an instance of the Braess network with cost functions that are known to induce suboptimal selfish-routing. The cost functions are shown in Figure 4.8.

To show that these results align with multi-agent learning, we simulated an instance of the network control game on this example for linear and quadratic edge-cost functions. We chose a proportional game, since this case has worst-case selfish-routing as indicated by Proposition 4.5.2.

The multi-agent RL algorithm chosen was the A3C algorithm Mnih *et al.* (2016), with either 1, 2, or 3 route planner agents controlling the flow. We chose

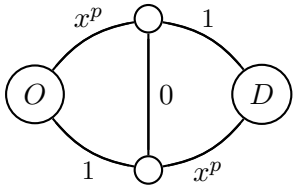

Figure 4.8: The Braess example where $d = 1$. With $p = 1$, the cost functions are linear, and for $p = 2$, the cost functions are quadratic.

this algorithm as it is often used the a baseline algorithm in MARL social dilemmas literature (e.g. Eccles *et al.* (2019b); Hughes *et al.* (2018); Jaques *et al.* (2019); Lerer & Peysakhovich (2019)) Each game consisted of playing the network control game on the nonatomic congestion game shown in Figure 4.8 for 100 repeated rounds, with proportional control. Note for linear cost functions, the social optimum cost is 150 and for quadratic the social optimum is 123. In both instances, the worst possible cost is 200.

Each instance was averaged over 3 different random seeds. The neural network consisted of two fully connected layers of size 32 and a Long Short Term Memory (LSTM) recurrent layer Gers *et al.* (1999), this architecture was chosen from Jaques *et al.* (2019). We used the Ray library[1] for a standard implementation of A3C. The hyperparameters are of the simulation are a learning rate of 0.001, and an entropy coefficient of -0.01 as these are the default parameters for A3C in Ray.

The learning curves for these experiments are shown in Figure 4.9. These results show that the agents learn to play strategies with a total cost that is close to the predicted Price of Anarchy (from Theorem 4.5.3) for the edge-cost type and number of agents. Thus, reinforcement learning agents that cannot solve SSDs are vulnerable to choosing suboptimal routing as predicted by the theory. Additional mechanisms, such as the ARCTIC algorithm from Chapter 3, could be used to encourage route planners to cooperate their routing.

[1] https://github.com/ray-project/ray

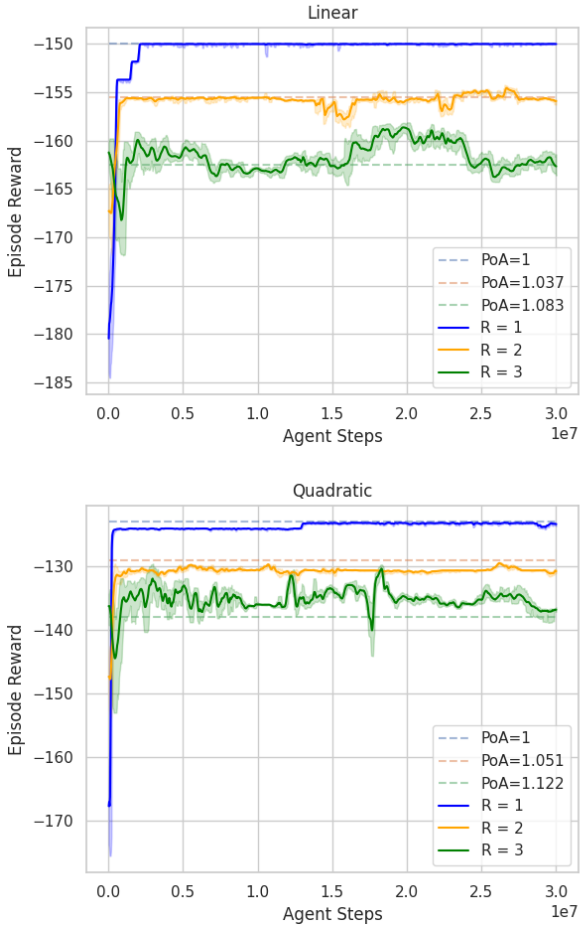

Figure 4.9: The learning curves for A3C agents playing a network control game on Braess' example for linear and quadratic edge costs.

4.7 Choosing Route Planners

So far, we have studied vehicles that are assigned to route planners controlling their choices. Suppose that instead, we allow vehicles to strategically select their route planner, prior to their journey (for instance, by selecting or purchasing a specific routing application or software). In this extension, Nash equilibrium outcomes are such that no vehicle has an incentive to unilaterally deviate from the route planner they selected, given the prescribed route choices.

Definition 4.7.1 (Route Planner Game)**.** *A* ***route planner game*** *is a tuple* $(\mathcal{M}, R)$ *where* $\mathcal{M}$ *is a nonatomic congestion game, and* R *is the set of route planners*

Furthermore, the strategy space of players in $\mathcal{M}$ is R, since their routing is

selected by the route planner they choose. Let y_r^i indicate the share of control of $r \in R$ selected by population $i \in N$. Then a strategy profile $\boldsymbol{y} = (\boldsymbol{y}^i)_{i \in N}$ is feasible if $\forall i \in N$, $\sum_{r \in R} y_r^i = d_i$. Each feasible $\boldsymbol{y}$ has a corresponding network control game, where $\forall r \in R$ and $\forall i \in N$, $y_r^i = d_i^r$, and $i \in N_r$ if $y_i^r > 0$. Thus, each $\boldsymbol{y}$ has an essentially unique Nash equilibria $\boldsymbol{d}$ deciding the distribution of information.

Define the cost function of a vehicle $i \in N$ to be

$$C_i(\boldsymbol{y}) := \sum_{r \in R} y_r^i \sum_{k \in \kappa_r} C_k(\boldsymbol{x}) d_k^r \mathbb{1}_{k \in K_i},$$

where $\boldsymbol{x}$ is the ICUE that results from $\boldsymbol{d}$. Moreover, a Nash equilibrium is $\boldsymbol{y}$ such that $\forall i \in N\ C_i(\boldsymbol{y}) \leq C_i(y', \boldsymbol{y})\ \forall y' \in R$.

Proposition 4.7.2. *A route planner game is an exact potential game for potential* Φ*, defined as*

$$\Phi(\boldsymbol{y}) := \sum_{e \in E} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}) d\boldsymbol{z},$$

where $\boldsymbol{x}$ *is the ICUE formed from* $\boldsymbol{d}$ *and* $\boldsymbol{y}$*.*

Proof. Consider the change in potential function between strategy distributions $\boldsymbol{y}$ and $\boldsymbol{y}' = (y'_j, \boldsymbol{y}_{-j})$ for some $j \in N$, with respective ICUE profiles $\boldsymbol{x}'$ and $\boldsymbol{x}$.

$$\Phi(\boldsymbol{y}') - \Phi(\boldsymbol{y}) = \sum_{e \in E} \int_0^{f_e(\boldsymbol{x}')} c_e(\boldsymbol{z}) d\boldsymbol{z} - \sum_{e \in E} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}) d\boldsymbol{z} \tag{4.6}$$

Rewrite equation 4.6 as a sum over possible strategies in S,

$$= \sum_{i \in N} d_i \sum_{k \in K_i} \sum_{s \in S_k} \left[x_s'^i \sum_{e \in s} \int_0^{f_e(\boldsymbol{x}')} c_e(\boldsymbol{z}) d\boldsymbol{z} - x_s^i \sum_{e \in s} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}) d\boldsymbol{z} \right]$$

Reformulate as a sum over route planners strategies,

$$= \sum_{i \in N} \sum_{r \in R} \sum_{k \in \kappa_r} d_k^r \mathbb{1}_{\{k \in K_i\}} \left[y_r'^i \sum_{e \in K_i} \int_0^{f_e(\boldsymbol{x}')} c_e(\boldsymbol{z}) d\boldsymbol{z} - y_r^i \sum_{e \in K_i} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}) d\boldsymbol{z} \right]$$

Since the only difference between $y_r'^i$ and y_r^i is when $i = j$,

$$= \sum_{r \in R} \sum_{k \in \kappa_r} d_k^r \mathbb{1}_{\{k \in K_j\}} \left[y_r'^j \sum_{e \in K_j} \int_0^{f_e(\boldsymbol{x}')} c_e(\boldsymbol{z}) d\boldsymbol{z} - y_r^j \sum_{e \in K_j} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}) d\boldsymbol{z} \right]$$

$$= \sum_{r \in R} \sum_{k \in \kappa_r} d_k^r \mathbb{1}_{\{k \in K_j\}} \left[y_r'^j C_k(\boldsymbol{x}') - y_r^j C_k(\boldsymbol{x}) \right]$$

$$= C_j(\boldsymbol{y}') - C_j(\boldsymbol{y})$$

Thus, Φ is an exact potential function. By definition, the network control game is an exact potential game. □

Thus, Corollary 4.7.3 follows.

Corollary 4.7.3. *There exists a Nash equilibrium and it is essentially unique.*

Now suppose we have a congestion game with a socially inefficient UE and at least two route planners controlling the flow. Any route planner that has a small share of control of a population will choose the same strategy as players in a congestion game. Similarly, any route planner with a large share of control of a population plays by routing according to the social optimum. Since the UE of the game is socially inefficient, we know that the players choosing the route planner with a large share of control will have a strictly greater cost than those choosing a route planner with a small share of control. Thus, vehicles choosing their route planners have an incentive to choose the one with the least control.

For example, consider a route planner game with the same setup as Example 2. Figure 4.10 shows the cost to each route planner for Example 2 divided by the share of control of the population, i.e., the cost to each vehicle in the game. It shows that when one route planner has strictly more control than the other, vehicles who chose the most popular route planner have an incentive to switch route planners.

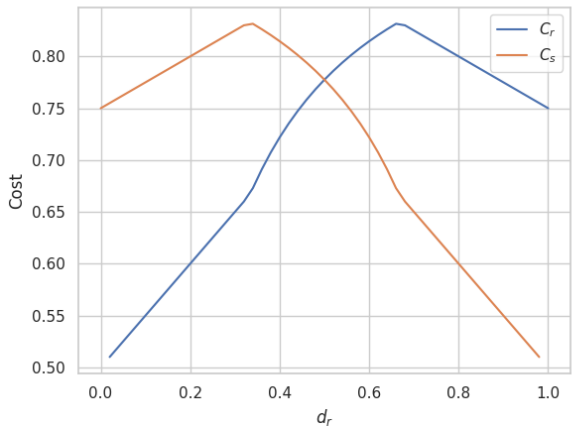

Figure 4.10: The cost to vehicle players choosing the 2 route planners, r and s, from Example 2, plotted against the share of control of the route planner r.

Any route planner that has less control over the population than any other route planner is more desirable to vehicles. Thus, there cannot be a route planner with strictly less control than all other route planners at the Nash equilibrium. We

have ruled out the case where a route planner has no control over any population, so the flow must be proportional at the equilibrium.

Proposition 4.7.4. *The Nash equilibrium of vehicles choosing route planners is proportional.*

Proof. Any route planner with share of control less than $(pR^{p-1}+R^p)^{-1/p}$ for population i will choose the same inefficient selfish routing as the vehicles of the congestion game. Since this is the UE of the game, the other routing must be greater than or equal to this cost. Thus, vehicles prefer to choose a route planner with less than $(pR^{p-1}+R^p)^{-1/p}$ control over their population. Since, $(pR^{p-1}+R^p)^{-1/p} \geq \frac{1}{R}$, the best response dynamics will end when all route planners have proportional control of all populations. □

Following from Proposition 4.5.2, we see that allowing vehicles to choose their route planner enforces the worst possible Price of Anarchy.

4.8 Discussion

We studied multiple route planners optimising the routing of subpopulations in a nonatomic congestion game. As their number grows, the game goes from achieving socially optimal routing to achieving the same inefficient routing as the original congestion game. We have found the exact bound on the Price of Anarchy of the induced game for polynomial edge-cost functions. Additionally, we allowed vehicles to choose their route planner and showed that this only increases the overall inefficiency.

Natural extensions include analysing games with partial route planner control and the rest as selfish players with full or partial information. Another line of further work is to discover under what conditions is there an incentive to follow a route planner rather than autonomous routing. Designing incentive mechanisms for drivers to choose route planner control whilst achieving some level of fairness would impact the use of route controllers in real-world traffic.

The results from this chapter theoretically support the implementation of a single route planner route planner to be implemented in autonomous vehicles. The higher the number of route planners controlling vehicles on the roads, the larger the inefficiency of suboptimal routing. This is also the case for navigation apps; the more applications available to drivers, the worse the outcome of selfish routing will be. However, the suboptimal routing could be mitigated if route planners cooperate with each other. The problem constitutes a social dilemma, so if route planners

were able to detect if their rivals were cooperating or defecting, algorithms such as ARCTIC could be adapted and utilised for safe cooperation in route control.

This work has a direct application to any type of congestion game where there exists social planners that can control the routing of it's users. For instance, in the distribution network of power grid demand.

CHAPTER 5

Distribution of Information in Nonatomic Congestion Games

5.1 Introduction

With the rising popularity of GPS route-guidance systems, many travellers rely on information about route choice to help them make navigation decisions. It is natural to assume that more information about paths available to a driver would only reduce their expected journey time. However, this was shown not to be the case Acemoglu *et al.* (2018); presenting new options can change the decision-making of some self-interested agents, causing an overall re-routing which makes them worse off.

It is likely that real-world travel costs will change over time e.g., from road improvements or temporary construction works. Braess' paradox (BP) assumes the state of the system is in a user equilibrium, where every player has minimised personal costs of travel given the actions of others. BP is based on the assumption that agents have complete information about the network structure. However, this assumption is often not met in practical situations, when, typically, actors have incomplete knowledge of their available paths. Recently, information constrained user equilibrium (ICUE) Acemoglu *et al.* (2018) has been introduced, where each player minimises their travel costs given their current knowledge. The equilibrium is reached through myopic improvements and gives rise to informational Braess' paradox (IBP), which occurs when users' cost at ICUE increase as a result of their information set expanding.

ICUE is a more better tool than UE to predict traffic flows, but the study of equilibria in nonatomic congestion games with incomplete information is still at its infancy. In particular, while the relationship between network structure and immunity to IBP has been characterised for single-population nonatomic congestion

games Acemoglu *et al.* (2018), for the more complex and realistic multi-population variant the exact conditions are still unknown. On top of that, IBP has only been formulated looking at one group, i.e., the one that acquires new knowledge, but not at the welfare of all agents, which is the standard social cost metric used for nonatomic congestion games.

5.2 Contributions

Here, we consider the properties of information expansion on nonatomic congestion games with players that respond rationally given their restricted knowledge about the available network. This expands upon the research on information constrained congestion games from Chapter 4, but without the influence of any social planner agents. Here, we consider heterogeneous players and focus on network design to improve routing efficiency under information distribution.

We begin by analysing how matroidal properties on a network will impact Braess' paradox, naturally leading to results on the immunity of matroids to Informational Braess' paradox in two-terminal networks. This is not an expansion on known results, but these results motivate the use of concepts from matroid theory to the asymmetric case.

We then advance the analysis of asymmetric nonatomic congestion games played by heterogeneous boundedly rational agents in two important ways. Firstly, we establish how a ubiquitous class of networks, rings, are immune to IBP, settling a conjecture in Acemoglu *et al.* (2018). We prove this for the more general structure of a circuit. Secondly, we extend the analysis of IBP to take into account the welfare of all agents, rather than a subset of them, showing that IBP is a widespread phenomenon and no network is immune to it under this measure. Our analysis is an important first step for the design of road systems that do not penalise the acquisition of new knowledge.

5.3 Literature Review

The first congestion game routing 'paradox' was posed by Braess Braess (1968) through an example of a network that suffers from the paradoxical increase of total cost when demand or cost of a resource is strictly decreased. Since this seminal work, the topic of routing paradoxes has been extensively explored Murchland (1970); Pas & Principio (1997); Zhao *et al.* (2014). Pas and Principio Pas & Principio (1997) classified demand constraints and linear cost functions that cause Braess' paradox in

traffic games on Braess-like networks. Later, Milchtaich Milchtaich (2006) illustrated the topological conditions required for an undirected network to be immune to Braess' paradox for a single population. More generally, Epstein et al. Epstein *et al.* (2009) examined topologies in which every Nash equilibrium is socially optimal. In asymmetric games, the exact conditions for network immunity are known Chen *et al.* (2015). Moreover, Tumer and Wolpert Tumer & Wolpert (2000) proposed a utility function allowing players are able to avoid Braess' paradox through the collective intelligence idea, that their decisions make an impact on the global cost.

Player heterogeneity has also been a topic of much interest, and research has covered many different types of heterogeneity. Many consider player-specific resource cost functions Milchtaich (1996) representing varied preferences Cole *et al.* (2018) or uncertainties Beier *et al.* (2004); Sekar *et al.* (2018). For example, drivers may vary in their 'value of time': some prefer to save money, and others choose to pay a toll to reduce their journey time. Additionally, more complex models may consider drivers' uncertainties over road conditions or demand Meir & Parkes (2015); this may occur in extreme weather events where roads are obstructed, for example, if flooding causes certain routes to become impassable.

In the age of the "internet of things", many drivers use software to choose their routes for them. Many journey planning applications select only a few routes for users to choose from, which severely limits the information about routes available to them. There is evidence that providing incomplete information to drivers about road capacities may be worse than providing no information at all Arnott *et al.* (1991). Along the same line, Liu et al. Liu *et al.* (2016) studied heterogeneity among players regarding the quality of information they receive and how this affects the equilibrium costs. Moreover, Acemoglu et al. Acemoglu *et al.* (2018) posed informational Braess' paradox for differing levels of information about possible paths, and classified network topologies as immune to this when considering a single OD pair. For asymmetric games, only certain network properties are known to guarantee immunity, full characterisation has not been achieved Acemoglu *et al.* (2018); Roman & Turrini (2019).

Sheffi and Daganzo Sheffi & Daganzo (1977) first posed the stochastic user equilibrium (SUE), one in which players have perception errors whilst comparing strategy costs. Several different paradoxes occur when studying SUE. They also observed Daganzo & Sheffi (1978) that if path travel times are fixed, the stochastic network loading results in an increase in total travel costs, when a new link is added to the network. This phenomenon is similar to Braess' paradox, but results from the positive assignment to all paths instead. If travel times are flow-dependent, this

stochastic loading effect is somewhat balanced by the effects of congestion.

Congestion games do not have to exist on a network, but the underlying structure that exists between resources has a significant impact on their equilibria. Matroid congestion games, where the strategy space of each player consists of the bases of a matroid on the set of resources, have been proven to have important properties. Ackermann et al. Ackermann *et al.* (2009) showed that both weighted and player-specific nonatomic congestion games admit pure Nash equilibria in the case of matroid congestion games. This matroid property is maximal in the sense that whenever there are two players both having allowable sets of resources that are not matroidal, then there is a prescribed embedding of the sets into the ground set of resources and cost functions so that the resulting game does not have an equilibrium. More recently, Fujishige et al. Fujishige *et al.* (2017) proved the sufficient combinatorial property of strategy spaces for nonatomic congestion games for which Braess' paradox cannot occur, are those with matroid bases.

Meir and Parkes Meir & Parkes (2018) introduced the biased Price of Anarchy (bPoA) as a measure of inefficiency for games with boundedly rational players. They considered games where play is affected by mistakes or behavioural biases including altruism, toll-sensitivity, and partial information. To bound the effects on social welfare for diverse populations, if taking into account network topology, the bPoA is calculated using the average bias of the population, otherwise, it depends on the maximal bias of a player. They showed that the equilibrium cost is dependent on the serial-parallel width of the network. More specifically, if a network is series-parallel, then their result (Meir & Parkes, 2018, Theorem 3.1) implies that it is immune to informational Braess' paradox.

5.4 Known Results on Immunity to IBP

To begin, let us review theory from previous literature that handles the known immunity to IBP. The topological conditions for immunity to BP with information heterogeneity have been characterised for nonatomic congestion games with only one population of players to be series linearly independent (SLI) networks.

Theorem 5.4.1. *Acemoglu* et al. *(2018) A two-terminal network nonatomic congestion game played on network G is immune to IBP if, and only if, G is an SLI network.*

The intuition for Theorem 5.4.1 comes from the properties of linearly independent (LI) networks. If we reduce the demand in an asymmetric LI network, then

there exists a route with strictly less flow (see Theorem 2.3.2). When we distribute information in an LI network, these players will strictly reduce demand of their original subnetwork and increase demand elsewhere. Thus, the costs of those using routes in the original subnetwork do not increase. Those receiving the information, therefore, have strictly reduced costs since otherwise they would not have opted to change routes. Hence, we maintain the property that any rerouting for players given information can only reduce their own cost in SLI networks, since it holds for subnetworks connected in series.

Now suppose there are multiple populations of players, i.e., the OD pairs vary across players. The conditions for immunity to IBP for such games are not yet fully know. However, some of the conditions known to hold for the full information case (see Theorem 2.3.3) correspond to immunity of IBP, as shown by Acemoglu *et al.* (2018).

Theorem 5.4.2. *Acemoglu* et al. *(2018) For any nonatomic congestion game on asymmetric network G, where $\forall i \in N$, $G_i = (V_i, E_i)$ is the relevant network, IBP does not occur if the following hold:*
(a) $\forall i \in N$, G_i is SLI
(b) For all distinct $i, j \in N$, either $E_i \cap E_j = \emptyset$, or $E_i \cap E_j$ consists of all coincident blocks of G_i and G_j.

One notes that the conditions from Theorem 5.4.2 that imply immunity to IBP depend only on the relevant network for each population and not on the available resources of information types.

5.5 Matroid Games and IBP

In this section, we consider Wardrop's traffic model and the properties of the underlying network that would generate a matroid. We establish that there are certain transformations that can be applied to a matroid to conserve its immunity to BP and IBP.

A *graphic matroid* is a matroid such that the set of resources are the edges of a network and the independent sets are acyclic subnetworks. It is well-known (White *et al.* (1986)) that one can form a matroid $M = (E, \mathcal{I})$ by representing the resource set as the set of edges and the independent sets are generated by its spanning trees.

Definition 5.5.1 (Network routing matroid)**.** *A network G and an OD pair form a* ***network routing matroid*** *if a matroid exists such that the resources are relevant*

edges of G and the independent sets are sections of acyclic paths between the pair of OD nodes.

Note here that the bases of a network routing matroid are distinct OD paths.

The following three lemmas are direct consequences of the definitions of trees and forests and are stated here to aid the understanding of the proof of Lemma 5.5.5 and Proposition 5.5.6 .

Lemma 5.5.2. *The union of any two trees is a forest.*

Lemma 5.5.3. *The subnetwork of any forest is also a forest.*

Lemma 5.5.4. *The common edge set of any two subnetworks of a tree is either empty or also a tree.*

Given these lemmas, we can now consider the formation of a matroid on a network. Let the ground set E be all the edges in a network G. A subset of E is independent if, and only if, it is a forest; that is, if it does not contain a simple cycle. Thus, we have the following lemma by definition of a matroid.

Lemma 5.5.5. *Every finite network G gives rise to a network routing matroid $M = (E, \mathcal{I})$ where the independent sets are forests.*

The combination of these four lemmas can be applied to network nonatomic congestion games as follows.

Proposition 5.5.6. *For any $i \in N$, relevant network $G_i = (V_i, E_i)$ will form a network routing matroid $M_i = (E_i, S_i)$ if, and only if, irredundant G is a multi-edge forest.*

Proof. "$\Rightarrow$"
If every relevant network G_i gives rise to a matroid, then by Lemma 5.5.5, the matroid is equivalent to a network routing matroid where the independent sets are forests. Thus, G_i is itself a forest. By definition, a relevant network contains only those edges which can be used in at least one OD path. Hence, G_i must be connected so each relevant network is a multi-edge tree. By Lemma 5.5.2, irredundant G formed from $G_1 \cup ... \cup G_n$ must be a multi-edge forest.
"$\Leftarrow$"
Suppose that irredundant G is a multi-edge forest. Consider an arbitrary relevant network G_i. By definition, it is a connected subnetwork of G, therefore G_i is a multi-edge tree. The strategy set necessarily consists of O_iD_i paths of equal length, where each path is a spanning tree. By Lemmas 5.5.5 and 5.5.3, the hereditary

property of matroids must hold true. Take any two subnetworks X and Y of G_i such that $|X| < |Y|$, then by Lemma 5.5.4 there must exist an edge $e \in Y \backslash X$ such that $X \cup \{e\}$ is acyclic. Hence, the augmentation property of matroids holds. Hence, G_i is a matroid. □

A network routing matroid is a type of graphic matroid since the spanning trees of each relevant network are the supersets of possible strategies.

Define a *subdivision of a routing matroid* G^* to be such that there exists a network routing matroid G where G^* can be formed by iterations of replacing edges in G by two edges with a single common vertex. This is simply a type of embedding. Here, note that a subdivision does not change the connectivity of a network or properties other than a number of edges of G^* can now represent a single edge of G.

Proposition 5.5.7. *If $G^* = (V^*, E^*)$ is a subdivision of a matroid $G = (V, E)$, where $O_i, D_i \in V \ \forall i \in N$, then G^* is immune to Braess' paradox.*

Proof. Any node added to the subdivision of G must not belong to any OD pair. As cost functions are separable, division of an edge into two will not affect loads between the original two vertices in any equilibrium. Hence, there will be no change to the number of possible paths between any OD pair. Therefore, the strategy sets of G^* will be isomorphic to the network routing matroid G. Hence, G^* is immune to Braess' paradox. □

Now we will consider the occurrence of heterogeneous information sets. On multi-edge forests, the expansion of information sets only occurs when at least one of the multi-edges is unknown to a population. First, let us assess the simplest case, a two-terminal network routing matroid.

Proposition 5.5.8. *Any two-terminal network routing matroid is immune to IBP.*

Proof. Proposition 5.5.6 implies that any relevant network G that forms a matroid $M = (E, S)$ is a multi-edge tree. The relevant network G contains only edges that are used in possible paths between the associated OD pair. Each path must necessarily pass through the same vertices since a tree contains only one unique set of vertices to pass through in any path between two nodes. The subnetwork of any two adjacent nodes is LI, and G is made from adjoining pairs of adjacent nodes in series. Hence G is SLI. Thus, using Theorem 5.4.1, we have that any two-terminal matroid nonatomic congestion game is immune to IBP. □

Now that we have established that two-terminal network routing games are immune to IBP we can consider the case for multiple populations.

Theorem 5.5.9. *If $\forall i \in N$, relevant network $G_i = (V_i, E_i)$ forms a matroid $M_i = (E_i, S_i)$, then the associated nonatomic congestion game is immune to IBP.*

Proof. By Proposition 5.5.6, we have illustrated that the irredundant network G is a multi-edge forest. For any $i \in N$, the relevant network G_i is a multi-edge tree and therefore is LI. For any two distinct $i, j \in N$, either $E_i \cap E_j = \emptyset$ or there exists some edges common to both relevant networks. Since both G_i and G_j are both multi-edge trees, by Lemma 5.5.4, any common edge sets must be a single LI block. Since any tree is LI, the common LI block must have the same terminal nodes. Therefore, the conditions of Theorem 5.4.2 are satisfied. □

The assumptions about the underlying networks of a matroid rely upon the structure of the connected resources and the locations of the OD pairs. Therefore, a subdivision will also uphold these properties.

Proposition 5.5.10. *Any subdivision of a network routing matroid G^* is immune to IBP.*

Proof. Let G^* be a subdivision of a network routing matroid G. By Theorem 5.5.9, G is immune to IBP. If $\forall i \in N$, G_i is LI, then $\forall i \in N$, G_i^* is also LI since the subdivision of a matroid does not alter its independence properties. Now consider relevant edge sets for all distinct $i, j \in N$. If $E_i \cap E_j = \emptyset$, then $E_i^* \cap E_j^* = \emptyset$. Finally, if $E_i \cap E_j$ is LI with the same terminal nodes, then $E_i^* \cap E_j^*$ will also be a tree when no terminal nodes have been altered. Hence, the conditions of Theorem 5.4.2 are again satisfied. □

Note that IBP cannot occur on a forest that is simple, since there is necessarily only one possible route between any two nodes on a tree by definition.

5.6 Circuit Games and IBP

In this section, we formally introduce circuits, most commonly found in matroid theory. Consider a set system $(E, \boldsymbol{\mathcal{C}})$, where E is the set of resources and $\boldsymbol{\mathcal{C}} \subseteq 2^E$, where the axioms A, B, and C hold true.

A. $\emptyset \notin \boldsymbol{\mathcal{C}}$;

B. If $\mathcal{C}_1, \mathcal{C}_2 \in \boldsymbol{\mathcal{C}}$ and $\mathcal{C}_1 \subseteq \mathcal{C}_2$, then $\mathcal{C}_1 = \mathcal{C}_2$,

C. For any two distinct $\mathcal{C}_1, \mathcal{C}_2 \in \boldsymbol{\mathcal{C}}$ such that $e \in \mathcal{C}_1 \cap \mathcal{C}_2$, there is a member $\mathcal{C}_3 \in \boldsymbol{\mathcal{C}}$ such that $\mathcal{C}_3 \subseteq (\mathcal{C}_1 \cup \mathcal{C}_2) \backslash \{e\}$.

Then $\boldsymbol{\mathcal{C}}$ is a *circuit* over E.

Definition 5.6.1 (Circuit game)**.** *A **circuit game** is a nonatomic congestion game in which every relevant network $G_i = (V_i, E_i)$ $\forall i \in N$ is a circuit.*

We now establish this useful proposition on circuit games.

Proposition 5.6.2. *In any circuit game on a network G, either G is a two-edge two-node ring, or G is simple.*

Proof. First, consider a network with two nodes. On any network with two nodes, a two-edge two-node ring (the Pigou network) is a circuit since the removal of either edge will create a spanning tree. Adding or removing edges to the network will lose the circuit properties of a two-edge two-node ring. Thus, the Pigou network is the only circuit on a network with two nodes.

Now we will prove the statement for networks with greater than two nodes. This part is done by contradiction. Suppose we have a network G that is not simple. Choose two nodes where there exist multiple edges between, call them e_1 and e_2. Suppose a population of players, i, can use e_1 in their strategy, then $e_2 \in E_i$ since they connect the same nodes. By definition of a circuit game, the relevant network of that population must be a circuit $\mathcal{C}_i$. We must have $e_1, e_2 \in \mathcal{C}_i$. Consider $\mathcal{C}_i' = \{e_1, e_2\}$. It is a dependent set since it is a cycle. Any of its proper subsets are independent, therefore, it is a circuit. By the circuit axioms A, B, and C, if $\mathcal{C}_i' \subseteq \mathcal{C}_i$ then $\mathcal{C}_i = \mathcal{C}_i'$. So the population i can only travel between the end nodes of e_1 and e_2. There must exist another population j, whose strategies also include e_1 and e_2 since, otherwise, the nonatomic congestion game of populations $N \backslash \{i\}$ is an equivalent game (one with the same equilibria). Population j must also have a relevant network that is a circuit and, hence, must travel on the circuit $\mathcal{C}_j = \{e_1, e_2\} = \mathcal{C}_i$. This implies that the populations are indistinct, which is a contradiction. Hence, G must be simple. □

Proposition 5.6.2, notice, implies that any ring forms a circuit game. Now we are in a position to prove the following[1]:

Proposition 5.6.3. *Any two-terminal circuit game is immune to IBP.*

[1]This result does not add anything to current existing literature, since it can be shown using Theorem 5.4.2. It is included only to improve the intuitive flow of reasoning.

The proof of Proposition 5.6.3 follows from Theorem 5.4.1 since it can be shown, using Proposition 5.6.2, that a circuit game network is SLI.

Lemma 5.6.4. *Any circuit game is either a cycle or it can be partitioned into multiple games where each relevant network of the game is a cycle.*

Proof. If the network is a cycle, then we are done. So let us assume the network is not a cycle. For any population i, the relevant network $G_i = (V_i, E_i)$ is a circuit. A circuit over a set of edges is either a cycle or self-loop, and we do not wish to consider self-loops as we are looking at simple networks. Hence, a circuit game is a collection of cycles. If these cycles are not connected then we are done, so assume at least one pair of cycles, $\mathcal{C}_i$ $\mathcal{C}_j$, contain at least one vertex in common. If $\mathcal{C}_i$ and $\mathcal{C}_j$ not distinct then the lemma holds, hence, assume they are distinct.

Now, suppose that these populations have at least one edge in common, e. Then by the definition of a circuit, $\exists \mathcal{C}_k \subseteq (\mathcal{C}_i \cup \mathcal{C}_j) \backslash \{e\}$. The vertices O_i, O_j, D_i, D_j all belong in $\mathcal{C}_k$. Since $e \in \mathcal{C}_i, \mathcal{C}_j$ $e \notin \mathcal{C}_k$, there must be one strategy from S_i and one from S_j that exist in $\mathcal{C}_k$. Yet $\mathcal{C}_k$ a circuit, so there are two possible strategies that connect every pair of its nodes. This is a contradiction since we must have removed a strategy from each of S_i and S_j since $e \notin \mathcal{C}_k$. Hence, any two distinct circuits cannot have an edge in common, i.e., have no two vertices in common. Any two circuits with a only single vertex in common can be reduced to two separate games, since we only consider common edge costs. □

When considering multiple origin-destination pairs, the circuit axioms A, B, and C now apply to slightly more complex structures than simple rings. A circuit game can comprise connected rings such that the OD pairs do not allow for traversal between rings. Before we can prove the immunity to IBP for such structures, we pose the more general statement that not all player types can be negatively impacted by a single information expansion.

Proposition 5.6.5. *For an asymmetric circuit game with $|N| \geq 2$, where $(\tilde{E}_{(i,k)})_{\{i \in N, k \in K_i\}}$ are expanded information sets such that $E_{(1,1)} \subset \tilde{E}_{(1,1)}$ and $E_{(i,k)} = \tilde{E}_{(i,k)}$ for any $(i,k) \neq (1,1)$, with associated ICUE $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$. Then, there exists at least one player type (i,k) $i \in N, k \in K_i$ such that $C_{(i,k)}(\tilde{\boldsymbol{x}}) \leq C_{(i,k)}(\boldsymbol{x})$.*

Before we formally prove this proposition, we will first motivate the reasoning with a brief explanation of the structure of the proof. Consider a circuit. Note that each population can have at most two strategies. Each population that has only one strategy will not affect the equilibria before and after the information expansion, except the type $(1,1)$ whose strategy set expands. Thus, we can assume that for n

populations there are at most $n+1$ information types since an equivalent game with $n+1$ information types will have the same equilibria. Since there are n populations with distinct OD pairs, the game must be embedded in a $2n$-edge circuit. We will then prove the proposition through contradiction. For each of the types, we know that the strategy they use before the expansion must have strictly less flow on at least one of the edges in the strategy than afterwards. If we then compare all $n+1$ inequalities, we will find that there is always a contradiction since the demands of populations must be nonnegative.

First, we consider the case where $n = 2$, since we will use this in our proof of Proposition 5.6.5. The case where $n = 2$ is as displayed in Figures 5.1 and 5.2. There can be up to four information types when $n = 2$, but it is sufficient to show the result for only two types, as the same reasoning holds with or without the other types. Firstly, we will prove the result to be true on the circuit game shown in Figure 5.1.

Proposition 5.6.6. *For the circuit game shown in Figure 5.1 with two populations, there exists at least one player type such that their equilibrium cost is strictly reduced after any information expansion.*

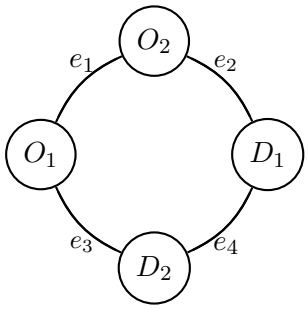

Figure 5.1: A circuit with two populations.

Proof. Let there be two populations each with distinct OD pairs as shown in Figure 5.1. Denote the strategy sets of player types be: type $(1,1)$ has strategy set $S_{11} = \{s_1\}$ before and $\tilde{S}_{11} = \{s_1, s_2\}$ after expansion; type $(1,2)$ has strategies $S_{12} = \{s_1, s_2\}$; and, type $(2,2)$ has $S_{22} = \{t_1, t_2\}$. The demands for these populations are $d_{11} > 0$, $d_{12} \geq 0$, $d_{22} > 0$. In order to reach a contradiction, suppose that there does not exist such a player. Namely, $\forall i \in \{1,2\}$ and $k \in K_i$, we must have $C_{(i,k)}(\tilde{\boldsymbol{x}}) > C_{(i,k)}(\boldsymbol{x})$.

Consider any feasible strategy distribution $\boldsymbol{x}$. Since type $(1,1)$ only has one strategy, we have $x_{11}^{s_1} = d_{11}$. In order for every player's cost to increase, we must

have that $(1,1)$ strictly prefers to deviate to their other strategy. Hence, $(1,2)$ must choose s_2 before information expansion, giving us $x_{12}^{s_2} = d_{12}$. Now, suppose that $(2,2)$ plays $x_{22}^{t_1} = pd_{22}$ and $x_{22}^{t_2} = (1-p)d_{22}$, where $p \in [0,1]$. The cost functions for the players are:

$$\begin{aligned} C_{11}(\boldsymbol{x}) = &\quad c_{e_1}(f_1) + c_{e_2}(f_2) \\ C_{12}(\boldsymbol{x}) = &\quad c_{e_3}(f_3) + c_{e_4}(f_4) \\ C_{22}(\boldsymbol{x}) = &\quad \begin{cases} c_{e_1}(f_1) + c_{e_3}(f_3) & p \in [0,1) \\ c_{e_2}(f_2) + c_{e_4}(f_4) & p \in (0,1] \end{cases} \end{aligned}$$

where f_1, f_2, f_3, f_4 are defined as

$$\begin{aligned} f_1 = &\quad d_{11} + (1-p)d_{22} \\ f_2 = &\quad d_{11} + pd_{22} \\ f_3 = &\quad d_{12} + (1-p)d_{22} \\ f_4 = &\quad d_{12} + pd_{22}. \end{aligned}$$

Now examine a feasible strategy distribution $\tilde{\boldsymbol{x}}$ which occurs after the information is distributed to player $(1,1)$. Without loss of generality, assume that both $(1,1)$ and $(1,2)$ have the same strategy distribution. Consider the following strategy distribution for population 1: $\tilde{x}_{1k}^{s_1} = d_{1k}$, $k \in \{1,2\}$. This could only be an ICUE if it was a dominant strategy given the total demand of population 2, which we know is not true given the deviation from s_1 in strategy distribution $\boldsymbol{x}$. So we must have $\tilde{x}_{1k}^{s_1} = qd_{1k}$ and $\tilde{x}_{1k}^{s_1} = (1-q)d_{1k}$, where $k \in \{1,2\}$ and $q \in [0,1)$. For population 2, let the strategy distribution be $x_{22}^{t_1} = \tilde{p}d_{22}$, $x_{22}^{t_2} = (1-\tilde{p})d_{22}$, where $\tilde{p} \in [0,1]$. The cost functions after information expansion are:

$$\begin{aligned} C_{11}(\tilde{\boldsymbol{x}}) = &\quad c_{e_3}(\tilde{f}_3) + c_{e_4}(\tilde{f}_4) \\ C_{12}(\tilde{\boldsymbol{x}}) = &\quad c_{e_3}(\tilde{f}_3) + c_{e_4}(\tilde{f}_4) \\ C_{22}(\tilde{\boldsymbol{x}}) = &\quad \begin{cases} c_{e_1}(\tilde{f}_1) + c_{e_3}(\tilde{f}_3) & \tilde{p} \in [0,1) \\ c_{e_2}(\tilde{f}_2) + c_{e_4}(\tilde{f}_4) & \tilde{p} \in (0,1] \end{cases} \end{aligned}$$

where $\tilde{f}_1, \tilde{f}_2, \tilde{f}_3, \tilde{f}_4$ are defined as

$$\begin{aligned} \tilde{f}_1 = &\quad q(d_{11} + d_{12}) + (1-\tilde{p})d_{22} \\ \tilde{f}_2 = &\quad q(d_{11} + d_{12}) + \tilde{p}d_{22} \\ \tilde{f}_3 = &\quad (1-q)(d_{11} + d_{12}) + (1-\tilde{p})d_{22} \\ \tilde{f}_4 = &\quad (1-q)(d_{11} + d_{12}) + \tilde{p}d_{22}. \end{aligned}$$

The contradiction assumption $C_{11}(\boldsymbol{x}) < C_{11}(\tilde{\boldsymbol{x}})$ gives us:

$$c_{e_1}(f_1) + c_{e_2}(f_2) < c_{e_3}(\tilde{f}_3) + c_{e_4}(\tilde{f}_4) \leq c_{e_1}(\tilde{f}_1) + c_{e_2}(\tilde{f}_2).$$

Since cost functions are nondecreasing, this implies that we must have $f_1 < \tilde{f}_1$ or $f_2 < \tilde{f}_2$. In terms of demands, either $d_{11} + (1-p)d_{22} < q(d_{11} + d_{12}) + (1-\tilde{p})d_{22}$ or $d_{11} + pd_{22} < q(d_{11} + d_{12}) + \tilde{p}d_{22}$. If both these hold, we have $\tilde{p} + \frac{(1-q)d_{11} - qd_{12}}{d_{22}} < p$ and $\tilde{p} + \frac{(1-q)d_{11} - qd_{12}}{d_{22}} > p$ which leads to a contradiction.

The contradiction assumption for player type $(1,2)$, $C_{12}(\boldsymbol{x}) < C_{12}(\tilde{\boldsymbol{x}})$, leads to

$$c_{e_3}(f_3) + c_{e_4}(f_4) < c_{e_3}(\tilde{f}_3) + c_{e_4}(\tilde{f}_4).$$

By nondecreasing cost functions, we must have at least one of $d_{12} + (1-p)d_{22} < (1-q)(d_{11} + d_{12}) + (1-\tilde{p})d_{22}$ and $d_{12} + pd_{22} < (1-q)(d_{11} + d_{12}) + \tilde{p}d_{22}$ hold. However, each of these either directly contradicts the previous two inequalities or is a direct implication of them. If one of the conditions for $(1,1)$ holds then exactly one of the assumptions for $(1,2)$ also holds.

Finally, consider $C_{22}(\boldsymbol{x}) < C_{22}(\tilde{\boldsymbol{x}})$. Suppose that $p, \tilde{p} \in (0,1)$. Then we must have

$$c_{e_1}(f_1) + c_{e_3}(f_3) < c_{e_1}(\tilde{f}_1) + c_{e_3}(\tilde{f}_3),$$

$$c_{e_2}(f_2) + c_{e_4}(f_4) < c_{e_2}(\tilde{f}_2) + c_{e_4}(\tilde{f}_4).$$

Similarly, by nondecreasing cost functions, we must have either $d_{11} + (1-p)d_{22} < q(d_{11} + d_{12}) + (1-\tilde{p})d_{22}$ or $d_{12} + (1-p)d_{22} < (1-q)(d_{11} + d_{12}) + (1-\tilde{p})d_{22}$. It must also be true that at least one of $d_{11} + pd_{22} < (q(d_{11} + d_{12}) + \tilde{p}d_{22}$ and $d_{12} + pd_{22} < (1-q)(d_{11} + d_{12}) + \tilde{p}d_{22}$ is true[2]. There is no combination of these inequalities which is not contradictory. Hence, for $p, \tilde{p} \in (0,1)$ the contradiction assumption is false.
Now suppose $p = [0,1)$, $\tilde{p} \in [0,1)$. Then we have

$$c_{e_1}(f_1) + c_{e_3}(f_2) < c_{e_1}(\tilde{f}_1) + c_{e_3}(\tilde{f}_3).$$

This gives us $\frac{(1-q)d_{11} - qd_{12}}{d_{22}} + \tilde{p} < p$ which means $p > 0$ since $d_{11}, d_{22} > 0$. So $\tilde{p} = 0$ otherwise, we have the case as above. Now this implies that

$$c_{e_2}(f_2) + c_{e_4}(f_4) < c_{e_2}(\tilde{f}_2) + c_{e_4}(\tilde{f}_4).$$

[2]Notice that d_{12} can be reduced from all inequalities so we can omit type $(1,2)$ and get the same result.

This again will lead to a contradiction.

Now suppose $p = (0, 1]$, $\tilde{p} \in (0, 1]$. It follows that

$$c_{e_2}(f_2) + c_{e_4}(f_4) < c_{e_2}(\tilde{f}_2) + c_{e_4}(\tilde{f}_4)$$

This gives us $\frac{(1-q)d_{12}-qd_{11}}{d_{22}} + p < \tilde{p}$, using similar reasoning as above we see that we must have $p < 1$ and $\tilde{p} = 1$. But this gives us

$$c_{e_1}(f_1) + c_{e_3}(f_3) < c_{e_1}(\tilde{f}_1) + c_{e_3}(\tilde{f}_3).$$

Hence, we reach a contradiction.

Now suppose $p = [0, 1)$, $\tilde{p} = (0, 1]$. Then we see that

$$c_{e_1}(f_1) + c_{e_3}(f_3) < c_{e_1}(\tilde{f}_1) + c_{e_3}(\tilde{f}_3).$$

So we must have the first assumption for $(1, 2)$ holding: $\frac{(1-q)d_{11}-qd_{12}}{d_{22}} + \tilde{p} < p$. Hence, we must have $p, \tilde{p} \in (0, 1)$ which we have already shown to be contradictory.

Now suppose $p = (0, 1]$, $\tilde{p} = [0, 1)$. Then we have

$$c_{e_2}(d_{11} + pd_{22}) + c_{e_4}(d_{12} + pd_{22}) < c_{e_2}(\tilde{f}_2) + c_{e_4}(\tilde{f}_4).$$

This implies $\frac{(1-q)d_{11}-qd_{12}}{d_{22}} + p < \tilde{p}$ must hold, which means that we have $p, \tilde{p} \in (0, 1)$, leading to the final contradiction. □

Now let us consider the other possible configurations of two population strategies on a circuit where $n = 2$. Again, we show that the same result holds true. Thus, Proposition 5.6.5 is true for $n = 2$.

Proposition 5.6.7. *For the circuit game shown in Figure 5.2 with two populations, there exists at least one player type such that their equilibrium cost is strictly reduced after any information expansion.*

Proof. As with the previous example we will show that Proposition 5.6.5 holds for $n = 2$ with the network shown in 5.2.

Let there be two populations each with distinct OD pairs as shown in Figure 5.1. Denote the strategy sets of player types be: type $(1, 1)$ with strategies $S_{11} = \{s_1\}$ before and $\tilde{S}_{11} = \{s_1, s_2\}$ after expansion; type $(1, 2)$ has strategies $S_{12} = \{s_1, s_2\}$; and type $(2, 2)$ has $S_{22} = \{t_1, t_2\} = \{e_3, e_4e_1e_2\}$. The demands for these

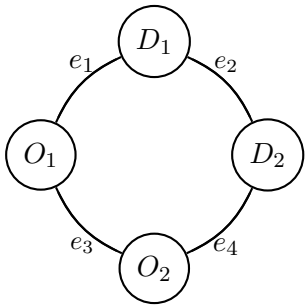

Figure 5.2: Another circuit with two populations.

populations are $d_{11} > 0$, $d_{12} \geq 0$, $d_{22} > 0$. In order to reach a contradiction, suppose that there does not exist such a player. Namely, $\forall i \in \{1, 2\}$ and $k \in K_i$, we must have $C_{(i,k)}(\tilde{\boldsymbol{x}}) > C_{(i,k)}(\boldsymbol{x})$.

Firsty, let us consider the case where $s_1 = e_1$ and $s_2 = e_4e_3e_2$.

Consider any feasible strategy distribution $\boldsymbol{x}$. Since type $(1, 1)$ only has one strategy we have $x_{11}^{s_1} = d_{11}$. In order for every player's cost to increase, we must have that $(1, 1)$ strictly prefers to deviate to their other strategy. Hence, $(1, 2)$ must choose s_2 before information expansion, giving us $x_{12}^{s_2} = d_{12}$. Now suppose that $(2, 2)$ plays $x_{22}^{t_1} = pd_{22}$ and $x_{22}^{t_2} = (1 - p)d_{22}$ where $p \in [0, 1]$.

The cost functions for the players are:

$$\begin{aligned} C_{11}(\boldsymbol{x}) = &\quad c_{e_1}(f_1) \\ C_{12}(\boldsymbol{x}) = &\quad c_{e_2}(f_2) + c_{e_3}(f_3) + c_{e_4}(f_4) \\ C_{22}(\boldsymbol{x}) = &\quad \begin{cases} c_{e_3}(f_3) & p \in (0, 1] \\ c_{e_1}(f_1) + c_{e_2}(f_2) + c_{e_4}(f_4) & p \in [0, 1) \end{cases} \end{aligned}$$

where f_1, f_2, f_3, f_4 are defined as

$$\begin{aligned} f_1 = &\quad d_{11} + (1 - p)d_{22} \\ f_2 = &\quad d_{12} + (1 - p)d_{22} \\ f_3 = &\quad d_{12} + pd_{22} \\ f_4 = &\quad d_{12} + (1 - p)d_{22}. \end{aligned}$$

Now examine a feasible strategy distribution $\tilde{\boldsymbol{x}}$ after the information is distributed to player $(1, 1)$. Without loss of generality, assume that both $(1, 1)$ and $(1, 2)$ have the same strategy distribution. Consider the following strategy distribution for population 1: $\tilde{x}_{1k}^{s_1} = d_{1k}$, $k \in \{1, 2\}$. This could only be an ICUE if it was a dominant strategy given the total demand of population 2, which we know is not true given the deviation from s_1 in strategy distribution $\boldsymbol{x}$. So we must have $\tilde{x}_{1k}^{s_1} = qd_{1k}$ and

$\tilde{x}_{1k}^{s_1} = (1-q)d_{1k}$, where $k \in \{1,2\}$ and $q \in [0,1)$. For population 2, let the strategy distribution be $x_{22}^{t_1} = \tilde{p}d_{22}$, $x_{22}^{t_2} = (1-\tilde{p})d_{22}$, where $\tilde{p} \in [0,1]$.

The cost functions after information expansion are:

$$\begin{aligned} C_{11}(\tilde{\boldsymbol{x}}) = &\quad c_{e_2}(\tilde{f}_2) + c_{e_3}(\tilde{f}_3) + c_{e_4}(\tilde{f}_4) \\ C_{12}(\tilde{\boldsymbol{x}}) = &\quad c_{e_2}(\tilde{f}_2) + c_{e_3}(\tilde{f}_3) + c_{e_4}(\tilde{f}_4) \\ C_{22}(\tilde{\boldsymbol{x}}) = &\quad \begin{cases} c_{e_3}(\tilde{f}_3) & \tilde{p} \in (0,1] \\ c_{e_1}(\tilde{f}_1) + c_{e_2}(\tilde{f}_2) + c_{e_4}(\tilde{f}_4) & \tilde{p} \in [0,1) \end{cases} \end{aligned}$$

where $\tilde{f}_1, \tilde{f}_2, \tilde{f}_3, \tilde{f}_4$ are defined as

$$\begin{aligned} \tilde{f}_1 = &\quad q(d_{11}+d_{12}) + (1-\tilde{p})d_{22} \\ \tilde{f}_2 = &\quad (1-q)(d_{11}+d_{12}) + (1-\tilde{p})d_{22} \\ \tilde{f}_3 = &\quad (1-q)(d_{11}+d_{12}) + \tilde{p}d_{22} \\ \tilde{f}_4 = &\quad (1-q)(d_{11}+d_{12}) + (1-\tilde{p})d_{22}. \end{aligned}$$

The contradiction assumption $C_{11}(\boldsymbol{x}) < C_{11}(\tilde{\boldsymbol{x}})$ gives us:

$$c_{e_1}(f_1) < c_{e_2}(f_2) + c_{e_3}(\tilde{f}_3) + c_{e_4}(\tilde{f}_4) \leq c_{e_1}(\tilde{f}_1).$$

Hence, by nondecreasing cost functions we must have $f_1 < \tilde{f}_1$. In terms of demand, we have $(1-q)d_{11} < qd_{12} + (p-\tilde{p})d_{22}$.

The contradiction assumption $C_{12}(\boldsymbol{x}) < C_{12}(\tilde{\boldsymbol{x}})$ gives us:

$$c_{e_2}(f_2) + c_{e_3}(f_3) + c_{e_4}(f_4) < c_{e_2}(f_2) + c_{e_3}(\tilde{f}_3) + c_{e_4}(\tilde{f}_4).$$

Hence, by nondecreasing cost functions we must have either $f_2 < \tilde{f}_2$, or $f_3 < \tilde{f}_3$ or $f_4 < \tilde{f}_4$. If $f_3 < \tilde{f}_3$, we have $(1-q)d_{11} > qd_{12} + (p-\tilde{p})d_{22}$, which directly contradicts the inequality necessary from $(1,1)$. For $f_2 < \tilde{f}_2$ and $f_4 < \tilde{f}_4$, we have $(1-q)d_{11} < qd_{12} - (p-\tilde{p})d_{22}$, which is a contradiction unless $p > \tilde{p}$.

Now consider the contradiction assumption for $(2,2)$, $C_{22}(\boldsymbol{x}) < C_{22}(\tilde{\boldsymbol{x}})$. Firstly, consider $p \in (0,1]$, which gives us

$$c_{e_3}(f_3) < c_{e_1}(\tilde{f}_1) + c_{e_2}(\tilde{f}_2) + c_{e_4}(\tilde{f}_4) \leq c_{e_3}(\tilde{f}_3) \qquad \tilde{p} \in [0,1)$$

$$c_{e_3}(f_3) < c_{e_3}(\tilde{f}_3) \qquad \tilde{p} \in (0,1]$$

Thus, by nondecreasing cost functions, $f_3 < \tilde{f}_3$. In terms of demand, we have $(1-q)d_{11} > qd_{12} + (p-\tilde{p})d_{22}$, which directly contradicts the inequality from population

$(1,1)$'s assumption. So we must have $p = 0$. However, the contradiction assumption from $(1,1)$ states that $p > \tilde{p}$ and since $\tilde{p} \geq 0$ we reach a contradiction.

Now lets consider the alternative case where $s_1 = e_4e_3e_2$ and $s_2 = e_1$.

As before, let us consider any feasible strategy distribution $\boldsymbol{x}$. We must have $x_{11}^{s_1} = d_{11}$ and $x_{12}^{s_2} = d_{12}$. Here, $(2,2)$ plays $x_{22}^{t_1} = pd_{22}$ and $x_{22}^{t_2} = (1-p)d_{22}$ where $p \in [0,1]$.

The cost functions for the players are:

$$\begin{aligned} C_{11}(\boldsymbol{x}) = &\quad c_{e_2}(f_2) + c_{e_3}(f_3) + c_{e_4}(f_4) \\ C_{12}(\boldsymbol{x}) = &\quad c_{e_1}(f_1) \\ C_{22}(\boldsymbol{x}) = &\quad \begin{cases} c_{e_3}(f_3) & p \in (0,1] \\ c_{e_1}(f_1) + c_{e_2}(f_2) + c_{e_4}(f_4) & p \in [0,1) \end{cases} \end{aligned}$$

where f_1, f_2, f_3, f_4 are defined as

$$\begin{aligned} f_1 = &\quad d_{12} + (1-p)d_{22} \\ f_2 = &\quad d_{11} + (1-p)d_{22} \\ f_3 = &\quad d_{11} + pd_{22} \\ f_4 = &\quad d_{11} + (1-p)d_{22}. \end{aligned}$$

Again, examine a feasible strategy distribution $\tilde{\boldsymbol{x}}$ after the information is distributed to player $(1,1)$. We must have $\tilde{x}_{1k}^{s_1} = qd_{1k}$ and $\tilde{x}_{1k}^{s_1} = (1-q)d_{1k}$, where $k \in \{1,2\}$ and $q \in [0,1)$. For population 2, let the strategy distribution be $x_{22}^{t_1} = \tilde{p}d_{22}$, $x_{22}^{t_2} = (1-\tilde{p})d_{22}$ where $\tilde{p} \in [0,1]$.

The cost functions after information expansion are:

$$\begin{aligned} C_{11}(\tilde{\boldsymbol{x}}) = &\quad c_{e_1}(\tilde{f}_1) \\ C_{12}(\tilde{\boldsymbol{x}}) = &\quad c_{e_1}(\tilde{f}_1) \\ C_{22}(\tilde{\boldsymbol{x}}) = &\quad \begin{cases} c_{e_3}(\tilde{f}_3) & \tilde{p} \in (0,1] \\ c_{e_1}(\tilde{f}_1) + c_{e_2}(\tilde{f}_2) + c_{e_4}(\tilde{f}_4) & \tilde{p} \in [0,1) \end{cases} \end{aligned}$$

where $\tilde{f}_1, \tilde{f}_2, \tilde{f}_3, \tilde{f}_4$ are defined as

$$\begin{aligned} \tilde{f}_1 = &\quad (1-q)(d_{11} + d_{12}) + (1-\tilde{p})d_{22} \\ \tilde{f}_2 = &\quad q(d_{11} + d_{12}) + (1-\tilde{p})d_{22} \\ \tilde{f}_3 = &\quad q(d_{11} + d_{12}) + \tilde{p}d_{22} \\ \tilde{f}_4 = &\quad q(d_{11} + d_{12}) + (1-\tilde{p})d_{22}. \end{aligned}$$

The contradiction assumption $C_{11}(\boldsymbol{x}) < C_{11}(\tilde{\boldsymbol{x}})$ gives us:

$$c_{e_2}(f_2) + c_{e_3}(f_3) + c_{e_4}(f_4) < c_{e_1}(\tilde{f}_1) \leq c_{e_2}(f_2) + c_{e_3}(\tilde{f}_3) + c_{e_4}(\tilde{f}_4).$$

By nondecreasing cost functions, we have either $f_2 < \tilde{f}_2$, or $f_3 < \tilde{f}_3$, or $f_4 < \tilde{f}_4$. For $f_2 < \tilde{f}_2$ and $f_4 < \tilde{f}_4$, we reach have the same inequality which is $(1-q)d_{11} < qd_{12} + (p-\tilde{p})d_{22}$. If $f_3 < \tilde{f}_3$ holds, we have $(1-q)d_{11} < qd_{12} - (p-\tilde{p})d_{22}$. The contradiction assumption $C_{11}(\boldsymbol{x}) < C_{11}(\tilde{\boldsymbol{x}})$ gives us:

$$c_{e_1}(f_1) < c_{e_1}(\tilde{f}_1).$$

Hence, by nondecreasing cost functions we have $f_1 < \tilde{f}_1$. In terms of demand, we have $(1-q)d_{11} > qd_{12} - (p-\tilde{p})d_{22}$, which directly contradicts that $f_3 < \tilde{f}_3$, hence we must have $(1-q)d_{11} < qd_{12} + (p-\tilde{p})d_{22}$. These two inequalities contradict each other for $p - \tilde{p} < 0$. Hence, we have that $p > \tilde{p} \geq 0$.

The contradiction assumption $C_{22}(\boldsymbol{x}) < C_{22}(\tilde{\boldsymbol{x}})$, where $p \in (0, 1]$, gives

$$c_{e_3}(f_3) < c_{e_3}(\tilde{f}_3) \qquad \tilde{p} \in (0, 1]$$

$$c_{e_3}(f_3) < c_{e_1}(\tilde{f}_1) + c_{e_2}(\tilde{f}_2) + c_{e_4}(\tilde{f}_4) \leq c_{e_3}(\tilde{f}_3) \qquad \tilde{p} \in [0, 1)$$

Thus, by nondecreasing cost functions we have $f_3 < \tilde{f}_3$. As before, this directly contradicts the assumptions from population $(1, 1)$. Hence, we have reached a contradiction for all games on the network shown in Figure 5.2. □

Now that we have proven the result true for the base case of $n = 2$, we will prove in for general n as stated in Proposition 5.6.5.

Proof of Proposition 5.6.5. We will show that the statement is true by contradiction. We must prove that if there always exists a subgame where the contradiction fails, then the contradiction assumption fails for the game. For any circuit game where $n > 2$, there exists a subgame where $n = 2$, since we can set the demand for $n - 2$ populations to be zero.

Let us consider all possible circuit games where $n = 2$. We have already considered all games with four edges through Propositions 5.6.6 and 5.6.7. Any circuit with two populations and more than four edges can be written as a four edge circuit game by the simple alteration of cost functions. The smallest possible circuit is a three edge ring, and this example is shown to be immune to IBP (Acemoglu *et al.*, 2018, Example 3). Thus, the contradiction statement cannot hold true in this case.

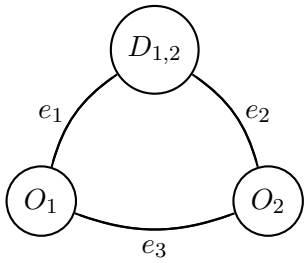

Figure 5.3: A three edge circuit with two populations.

Origin and destination nodes of a population are interchangeable since we consider undirected graphs. Hence, all cases of $n = 2$ have been shown in Figures 5.1, 5.2 and 5.3 to not uphold the contradiction assumption. For any n population ciruit games, there always exists a subgame which is equivalent to one of the games in Figures 5.1, 5.2 or 5.3. Since the contradiction assumption cannot hold for subgames of the game, it cannot hold for the general case. □

Now that we have shown that a circuit game will not increase all player's costs from informat ion distribution simultaneously, we can prove that information cannot harm the player who receives it.

Theorem 5.6.8. *Any circuit game is immune to IBP.*

Proof. By the definition of IBP, there exists an information type whose information set expands. Assume without loss of generality that type $(1, 1)$ are those players with expanded information sets. To reach a contradiction, assume that $C_{(1,1)}(\tilde{\boldsymbol{x}}) > C_{(1,1)}(\boldsymbol{x})$ where $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ are the ICUEs reached before and after the information set of type $(1, 1)$ is expanded respectively.

Since each player's relevant network is a circuit, the possible paths are divided into the two directions that one can travel around the circuit. By Proposition 5.6.2, there is only a single edge that can connect any two nodes, so each population must have two linearly independent paths available to them. Player type $(1, 1)$ with restricted information set has only one possible path before expansion and player type $(1, 2)$ has two paths available to them: $S_{(1,1)} = \{s_a\}$ and $S_{(1,2)} = \{s_a, s_b\}$.

For any two distinct $i, j \in N$, either $E_i \cap E_j = \emptyset$ or $E_i \cap E_j = \mathcal{C}$. If $E_1 \cap E_j = \emptyset$, then we do not need to consider population j as it will not affect the equilibrium costs of $(1, 1)$. So assume that for all $j \in N$, we have $E_1 \cap E_j = \mathcal{C}$. Now suppose that type $k_j \in K_j$ only has one choice of path. Then we can consider an equivalent game where the costs of resources $e \in E_{(j,k_j)}$ are increased to $c_e(f_e(\boldsymbol{x}) + d_{jk_j})$. Hence, we

can assume that for any $j \in N\backslash\{1\}$, there exists only one information type $(j, 2)$, where each player has full information about relevant resources.

If $(1, 1)$ does not choose s_b after the information set expansion, then the ICUE remains unchanged and we are done. Therefore, $\tilde{\boldsymbol{x}}_{11}^{s_b} > 0$ and $\tilde{\boldsymbol{x}}_{11}^{s_a} < \boldsymbol{x}_{11}^{s_a} = d_{11}$.

Suppose that $\sum_{e\in s_a} f_e(\tilde{\boldsymbol{x}}) \leq \sum_{e\in s_a} f_e(\boldsymbol{x})$. For $\tilde{\boldsymbol{x}}_{11}^{s_a} \geq 0$, then, following on from the definition of ICUE, we have $\sum_{e\in s_b} c_e(f_e(\tilde{\boldsymbol{x}})) \leq \sum_{e\in s_a} c_e(f_e(\tilde{\boldsymbol{x}}))$. Since c_e is continuous and nondecreasing, we reach the following contradiction:

$$C_{11}(\tilde{\boldsymbol{x}}) \leq \sum_{e\in s_a} c_e(f_e(\tilde{\boldsymbol{x}})) \leq \sum_{e\in s_a} c_e(f_e(\boldsymbol{x})) = C_{11}(\boldsymbol{x}).$$

Hence, we have that $\sum_{e\in s_a} f_e(\tilde{\boldsymbol{x}}) > \sum_{e\in s_a} f_e(\boldsymbol{x})$. Now divide the player types into two sets as follows:

$$A := \{i \in N, j \in \{1, 2\} : C_{ij}(\tilde{\boldsymbol{x}}) > C_{ij}(\boldsymbol{x})\}$$
$$B := \{i \in N, j \in \{1, 2\} : C_{ij}(\tilde{\boldsymbol{x}}) \leq C_{ij}(\boldsymbol{x})\}$$

By the contradiction assumption, A is nonempty and Proposition 5.6.5 tells us that B is nonempty.

All possible paths between any two nodes from the irredundant network $\hat{G}$ form the set $\mathcal{S}$. Divide this into two distinct sets as

$$S_A := \{s \in \mathcal{S} : C(s, \tilde{\boldsymbol{x}}) > C(s, \boldsymbol{x})\}$$
$$S_B := \{s \in \mathcal{S} : C(s, \tilde{\boldsymbol{x}}) \leq C(s, \boldsymbol{x})\}$$

Consequently, we must have that $\max_{s\in S_A}\{C(s, \boldsymbol{x}) - C(s, \tilde{\boldsymbol{x}})\} < 0$ and $\min_{s\in S_B}\{C(s, \boldsymbol{x}) - C(s, \tilde{\boldsymbol{x}})\} \geq 0$. Consider these two simple claims:

Claim 1: If $i \in A$ and $s \in S_B$, then $x_i^s = 0$.

This immediately follows from definitions: by definition of S_B, $C(s, \boldsymbol{x}) \geq C(s, \tilde{\boldsymbol{x}})$; by definition of ICUE, $C(s, \boldsymbol{x}) \geq C_i(\tilde{\boldsymbol{x}})$; by definition of A, we must have $C_i(\tilde{\boldsymbol{x}}) > C_i(\boldsymbol{x})$. Hence, $x_i^s = 0$.

Claim 2: If $i \in B$ and $s \in S_A$, then $\tilde{x}_i^s = 0$.

Again, this follows from definitions: by definition of S_B, $C(s, \tilde{\boldsymbol{x}}) > C(s, \boldsymbol{x})$; by definition of ICUE, $C(s, \boldsymbol{x}) \geq C_i(\boldsymbol{x})$; and finally, by definition of A, $C_i(\boldsymbol{x}) > C_i(\tilde{\boldsymbol{x}})$. Hence, $\tilde{x}_i^s = 0$.

For ease of notation, let demands for paths in S_A and S_B be

$$d_A = \sum_{s\in S_A}\sum_{i\in N} x_i^s \qquad d_B = \sum_{s\in S_B}\sum_{i\in N} x_i^s$$
$$\tilde{d}_A = \sum_{s\in S_A}\sum_{i\in N} \tilde{x}_i^s \qquad \tilde{d}_B = \sum_{s\in S_B}\sum_{i\in N} \tilde{x}_i^s$$

It follows from Claims 1 and 2 that we have $\tilde{d}_A \leq d_A$ and $\tilde{d}_B \geq d_B$. Since A and B are nonempty, it also follows that both S_A and S_B are nonempty.

Claim 3: Let S_α, S_β be any nonempty partition of $\mathcal{S}$. If we have $\tilde{d}_\alpha \leq d_\alpha$ and $\tilde{d}_\beta \geq d_\beta$, then

$$\max_{s\in S_\alpha}\{C(s,\boldsymbol{x}) - C(s,\tilde{\boldsymbol{x}})\} \geq \min_{s\in S_\beta}\{C(s,\boldsymbol{x}) - C(s,\tilde{\boldsymbol{x}})\}.$$

We will prove Claim 3 by induction on the number of edges and the number of populations. The base case is a circuit with three edges and one population. All possible paths of the network are $\mathcal{S} = \{e_1, e_2, e_3, e_1e_2, e_1e_3, e_2e_3\}$. Let the population's strategy set be $S = \{e_1e_2, e_3\}$, and their information set before expansion be $E = \{e_1, e_2\}$. This two-terminal game is SLI, so by Theorem 5.4.2, it is immune to IBP. For any $s \in \mathcal{S}$ such that $s \notin S$, s contributes no demand to $d_\alpha, \tilde{d}_\alpha, d_\beta$ or $\tilde{d}_\beta$, hence, any such s can be randomly assigned to one of the sets S_α and S_α. Since $e_1e_2 \in E$, the demand for this strategy can only reduce after the information expansion, $e_1e_2 \in S_\alpha$. The demand for e_3 can only increase after the information expansion, $e_3 \in S_\beta$. Since there is no IBP, we must have

$$C(e_1e_2,\boldsymbol{x}) \geq \begin{cases} C(e_1e_2,\tilde{\boldsymbol{x}}) & \text{if } C(e_1e_2,\tilde{\boldsymbol{x}}) \leq C(e_3,\tilde{\boldsymbol{x}}) \\ C(e_3,\tilde{\boldsymbol{x}}) & \text{if } C(e_1e_2,\tilde{\boldsymbol{x}}) \geq C(e_3,\tilde{\boldsymbol{x}}). \end{cases}$$

If $C(e_1e_2,\tilde{\boldsymbol{x}}) \leq C(e_3,\tilde{\boldsymbol{x}})$ then $C(e_1e_2,\boldsymbol{x}) - C(e_1e_2,\tilde{\boldsymbol{x}}) \geq 0$. If $C(e_1e_2,\tilde{\boldsymbol{x}}) \geq C(e_3,\tilde{\boldsymbol{x}})$, then $C(e_3,\boldsymbol{x}) \leq C(e_3,\tilde{\boldsymbol{x}})$ as $f_{e_3}(\boldsymbol{x}) < f_{e_3}(\tilde{\boldsymbol{x}})$. Hence, $\max_{s\in S_\alpha}\{C(s,\boldsymbol{x}) - C(s,\tilde{\boldsymbol{x}})\} \geq \min_{s\in S_\beta}\{C(s,\boldsymbol{x}) - C(s,\tilde{\boldsymbol{x}})\}$.

Now consider a subdivision of this circuit. The same reasoning holds, hence, it is true for a circuit with any number of edges. Now we assume that Claim 3 is true for $n-1$ populations on a circuit with an arbitrary number of edges and we will show how the addition of another population will not affect this property. WLOG, consider player type $(1,1)$ with strategy sets $S_{11} = \{s_\alpha\}$ and $\tilde{S}_{11} = \{s_\alpha, s_\beta\}$. If $\forall s \in \mathcal{S}$ such that $s \notin S_{11} \cup ... \cup S_{n2}$, then s contributes no demand to $d_\alpha, \tilde{d}_\alpha, d_\beta$ or $\tilde{d}_\beta$. Hence, can be arbitrarily assigned to one of the sets S_α and S_β. We can assume

$C(s_\alpha, \boldsymbol{x}) \geq C(s_\beta, \boldsymbol{x})$ since, otherwise, $C(s, \boldsymbol{x}) - C(s, \tilde{\boldsymbol{x}}) = 0 \ \forall s \in \mathcal{S}$ and, Claim 3 is immediately true. Hence, we know that the demand for s_α will strictly reduce, and the demand for s_β will strictly increase.

For any $i \in N\backslash\{1\}$, their strategy set is $S_i = \{s_{\alpha i}, s_{\beta i}\}$. It must be the case that $\forall i \in N\backslash\{1\}$ if demand for $s_{\alpha i}$ increases, the demand for $s_{\beta i}$ must reduce. Suppose that $\exists i \in N\backslash\{1\}$ such that the demands for $s_{\alpha i}, s_{\beta i}$ remain the same, then we can form an equivalent game of $n-1$ populations, for which we assumed the claim to be true. Therefore, in each population, we must have a strict increase in demand for one strategy and strict decrease for the other. Since we assumed $\tilde{d}_\alpha \leq d_\alpha$, $\tilde{d}_\beta \geq d_\beta$, and that both S_α and S_β are nonempty, in every possible allocation of strategies to S_α and S_β there exists $i \in N$ such that exactly one strategy of S_i belongs to S_α. Hence, there always exists a population $i \in N$ whose strategies belong in both S_α and S_β. Without loss of generality, let the demand for $s_{\alpha i}$ increase and belong to S_α and demand for $s_{\beta i}$ reduce and belong to S_β. Then we must have that $C_i(s_{\alpha i}, \tilde{\boldsymbol{x}}) \leq C_i(s_{\beta i}, \tilde{\boldsymbol{x}})$ and $C_i(s_{\alpha i}, \boldsymbol{x}) \geq C_i(s_{\beta i}, \boldsymbol{x})$. Therefore we have $C_i(s_{\alpha i}, \boldsymbol{x}) - C_i(s_{\alpha i}, \tilde{\boldsymbol{x}}) \geq C_i(s_{\beta i}, \boldsymbol{x}) - C_i(s_{\beta i}, \tilde{\boldsymbol{x}})$. This concludes the induction step and so we have proved Claim 3.

Finally, with the partition of S_A and S_B and the claims, we reach the following contradiction:

$$0 > \max_{s \in S_A} \{C(s, \boldsymbol{x}) - C(s, \tilde{\boldsymbol{x}})\} \geq \min_{s \in S_B} \{C(s, \boldsymbol{x}) - C(s, \tilde{\boldsymbol{x}})\} \geq 0.$$

□

An interesting application of this theorem is considering pedestrian exit-routing in sports stadiums, such as in Figure 5.4. Divide player populations as those who are seated in the same block and wish to use the same mode of transport. Knowledge of the layout can differ between people and can be distributed through signs or movement restrictions provided by the stadium. If pedestrians are only allowed to exit their seat block through a single exit, as in Figure 5.4, then the overlapping network of all populations is a ring. We can restrict the analysis to the circuit network since the flow on all of the other edges is deterministic. Thus, we can consider an equivalent circuit game with the same equilibria. Hence, information cannot harm the expected travel times of exiting the stadium. However, if visitors are allowed to walk between seat blocks before exiting the stadium the network no longer has immunity to IBP. These results, therefore, have useful implications for planning purposes e.g., evacuation routes.

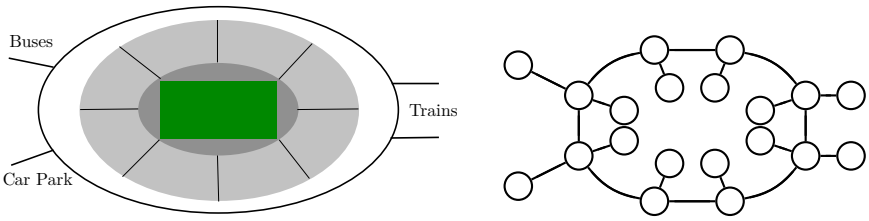

Figure 5.4: A sports stadium has a typical ring structure, hence, it is also a circuit.

5.7 IBP for Social Cost

Thus far, we defined IBP to be the comparison between equilibrium costs of a player whose information set is expanded. A natural weakness of using IBP to analyse the system as a whole is that it does not incorporate any effects that the information expansion has on all players. From a mechanism design perspective, it is also relevant to compare the social costs of the ICUEs. In this section, we define our own version of the IBP, which is measured against social cost, and show that its occurrence is independent of network topology.

Definition 5.7.1 (Informational Braess' Paradox for Social Cost)**.** ***Informational Braess' Paradox for Social Cost*** *(IBPSC) occurs if there exist expanded information sets* $(\tilde{E}_{(i,k)})_{\{i\in N,\, k\in K_i\}}$ *with* $E_{(1,1)} \subset \tilde{E}_{(1,1)}$ *and* $E_{(i,k)} = \tilde{E}_{(i,k)}$ *for any* $(i,k) \neq (1,1)$ *with associated ICUE* $\boldsymbol{x}$ *and* $\tilde{\boldsymbol{x}}$*, where the total costs to all players increases* $\sum_{i\in N}\sum_{k\in K_i} C_{(i,k)}(\boldsymbol{x}) < \sum_{i\in N}\sum_{k\in K_i} C_{(i,k)}(\tilde{\boldsymbol{x}})$.

IBPSC occurs when one player type has their information set strictly expanded resulting in an increase to the social cost. IBP is a special case of IBPSC that occurs when information harms the informed player. In order to consider how the social cost will change in an ICUE equilibrium, we need the following lemma for the simplest circuit network - two-node, two-edge ring (also known as the Pigou network).

Lemma 5.7.2. *Any two-terminal network with at least two distinct paths has the two-edge ring embedded in it.*

Proof. Suppose the statement is not true, then there does exist a network G with at least two distinct paths which do not embed a two-edge ring. Since a two-edge ring is the simplest network that contains a cycle, any cyclic network must embed the two-edge ring. Therefore, G is acyclic. Any acyclic undirected connected network must be a tree. In a tree, the number of possible paths between any two pairs of nodes is necessarily one. This contradicts the original statement. □

Now we show that there will always be a set of cost functions for any network that allows for IBPSC to occur.

Theorem 5.7.3. *For any two-terminal network there exists* $(K_i)_{i\in N}$ *and* $(c_e)_{e\in E}$ *such that IBPSC occurs.*

To prove this, it suffices to show that there always exists an assignment of information sets and cost functions on the two-edge ring such that IBPSC exists. This follows since we can always find a subnetwork of a network that is a two-edge ring, unless the network is a single edge, or edges attached only in series. For those types of networks, there is a single path connecting the origin and destination, thus, no information expansion can exist on these networks.

Proof of Theorem 5.7.3. IBPSC requires at least one type of player to be resource unaware. Therefore, at least two distinct strategies must exist. If there exist two distinct strategies then Lemma 5.7.2 the network must have the simplest circuit network embedded in it. By Lemma 5.7.2, it suffices to show that there always exists an assignment of information sets and cost functions on the two-edge ring such that IBPSC exists.

Consider two populations with demands $d_1 = 1, d_2 = 1$. Suppose their information sets are $E_1 = \{e_1\}$ and $E_2 = \{e_1, e_2\} = E$. Let $c_{e_1}(\boldsymbol{x}) = x$ and $c_{e_2}(\boldsymbol{x}) = 2$. At equilibrium, type 1 will choose e_1 and type 2 will choose e_1 with a social cost of 3. Now, if we expand type 1's information set to be $\tilde{E}_1 = \{e_1, e_2\}$, then since e_1 costs strictly less than e_2, type 1 will switch their strategy and the new equilibrium has a social cost of 4. □

The occurrence of IBPSC is independent of the topology of the network due to inefficiencies from selfish-routing. This is similar to Price of Anarchy results Roughgarden (2005). Through the assignment of information sets, an ICUE outcome can be found with a cost strictly less than any UE. Any stategy distribution that is not a UE must, therefore, be unstable, as at least one player wishes to deviate from it if they have their information set expanded. Hence, there will always exist an assignment of information sets for any network where the social cost expands as information sets expand. We conclude with the following result.

Theorem 5.7.4. *No network is immune to IBPSC.*

This result follows from the two-terminal case, implying there is further research needed to understand the impact of information distribution on different

network topologies. The worst-case informationally constrained selfish-routing social welfare, i.e., bounds on the biased Price of Anarchy, has been calculated for general networks Meir & Parkes (2018).

5.8 Discussion

We have analysed nonatomic congestion games where multiple populations of individuals have incomplete knowledge of the network structure, studying how the distribution of information affects utility. Specifically, we have identified a natural class of networks, i.e., rings, that are immune to performance deterioration for the agents acquiring new information, known as informational Braess' paradox. We have also shown that under an alternative definition of performance, analogous to Braess' paradox, no network has immunity. Previous results (Meir & Parkes, 2018, Theorem 3.1), implied that if a network has a serial-parallel width of one, then informational Braess' paradox will not occur on the network. However, we improved upon this by finding a class of networks, circuits, with a series-parallel width of two that also have this property.

Future directions of this work would be to see whether there are more network topologies that are immune to IBP. The characterisation of all immune structures is still an open problem. Additionally, there may be interesting results that occur when analysing the atomic congestion game variant of IBP. Atomic games could be more useful when looking at smaller areas of traffic. Moreover, there may be a link to games with unawareness, when players do not have knowledge of their imperfect information, that is still unexplored.

We believe that the identification of safe network structure is bound to fundamentally impact the design of transportation networks; aiding decongestion of ring roads as well as pedestrian evacuation in stadia. Further study into topologies immune to IBP could help reduce congestion caused by asymmetric information.

One limitation of information constrained congestion games is that they still assume that players have perfect information about congestion levels. This is restrictive in the real-world applications of the model since any congestion information about roads is time-dependent. Therefore, the assumption of perfect information cannot hold since accurate information about future congestion would be near impossible to obtain. In the case of a network with entirely autonomous vehicles, the perfect information assumption could hold under certain conditions.

Another practical application of this research is the design of peer-to-peer computing networks. The results could improve the efficiency of information distri-

bution in this area.

CHAPTER 6

Nonatomic Congestion Games with Traffic Lights

6.1 Introduction

Traffic lights have the ability to reduce urban congestion significantly, and finding their best configuration is paramount for the branch of AI concerned with optimising traffic. Due to the expense and restraints of changing existing road networks, the improvement and optimisation of traffic light systems is a highly active research area Mousavi *et al.* (2017); Pol & Oliehoek (2016). However, the role of traffic lights in inducing desirable equilibrium flows in transport networks is yet to be understood. In order to analyse the full effects of these changes, we model a system where intelligent drivers update their routing choices based on the knowledge of traffic lights.

Although traffic lights vary across the world, their critical function can be distilled as controlling whether or not traffic flows through repeated phases of coloured lights. A green light allows travel across the junction and a red light means that drivers must wait. The main purpose of the light is to allow the safe crossing of counterflow traffic. Nonetheless, they could play an essential role in coordinating traffic to reduce congestion, as well. Many traffic lights have a fixed period for each light colour and repeat in the same order. However, in some places, such as Germany, there are actuated traffic lights that do not change to the next phase in the cycle until traffic flow is lower than a certain threshold. Here, we only consider fixed time cycle lights and adaptive reinforcement learning lights due to their prevalence in the literature Laszka *et al.* (2016); Lopez *et al.* (2018).

In this chapter, we first formulate a nonatomic congestion game with traffic lights and show that it has an essentially unique equilibrium. We then consider the procedure of making a network resistant to Braess' paradox through changing

traffic light cycles and calculate the corresponding Price of Anarchy. We continue by extending the game to include adaptive traffic lights that change their light sequence in response to congestion. Finally, we quantify traffic lights' fairness properties and show that intelligent traffic lights optimising average travel times can cause an unfair bias to some cars' journey times.

6.2 Contributions

In this chapter, we continue to study the congestion game model and the existence of Braess' paradox as in Chapter 5. However, we now alter the cost functions to represent traffic lights.

We equip congestion games with junction-based waiting cycles, encoding traffic lights, and prove that the equilibria of such games exist and are essentially unique. We then consider how the presence of traffic lights effect Braess' paradox: there always exist an allocation of traffic light cycles that avoid it, but this is no longer the case when we relax the assumption that waiting times have no upper bound. However, we also include a result with conditions sufficient for immunity in the bounded case.

We continue by calculating the Price of Anarchy for these games as a function of the traffic light cycles for upper-bounded waiting times and polynomial cost functions. The Price of Anarchy bound is a function dependent on the minimum possible waiting time at any node.

Then we adapt the traffic light game to include traffic light agents that can change their light cycle in order to optimise local waiting times. When modelling traffic lights as optimisers of a natural social welfare function, we show the preservation of the game's potential function and, once again, the convergence to essentially unique equilibria. This suggests the effectiveness of simple and scalable reinforcement learning methods for practical applications whenever the underlying traffic network can reasonably be modelled as a nonatomic congestion game.

Finally, our experimental results use the Simulation of Urban Mobility (SUMO) software to indicate some critical issues of MARL traffic lights. Namely, unfair distribution of journey times can occur. We consider alternative reward functions as a method of reducing bias and suggest fairness metrics that could be applied to measure these effects.

6.3 Literature Review

Smith Smith & Van Vuren (1993); Smith (1985) was the first to consider traffic lights in the context of traffic assignment by showing there exists a unique solution to traffic light cycles, assuming infinite waiting times exist on edges at full capacity. Further work by Smith Smith (1981) introduced the responsive control policy P_0, which encourages the use of roads with a higher capacity, by giving these routes a green light bias. More recently, a continuous-time queue model was implemented with the dynamic user equilibrium Yu *et al.* (2018). As proposed in this chapter, our traffic light game differs from that of Smith by creating a format for both infinite and finite waiting times to be considered, consistent with implementing bounded green times in cycles.

There is a clear connection between traffic lights and tolls, with the latter widely investigated in algorithmic game theory. Originating from Pigou's observations Pigou (1920), marginal cost tolls have been studied extensively to improve system equilibria. Edge price schemes, designed by a social planner, have been shown to find an efficient equilibrium in nonatomic congestion games Sandholm (2002). Similar results hold for a nonatomic congestion game variant with heterogeneous players Karakostas & Kolliopoulos (2009). In atomic games, tolls are not guaranteed to strongly enforce optimal flow. Moreover, Meir and Parkes Meir & Parkes (2016) established the atomic congestion game efficiency bounds for dynamic marginal cost tolls, including players with variable tax sensitivity.

Static tolls assume that the network flow is always at equilibrium, which is unrealistic for real-world application. Dynamic tolls are able to adapt to varying congestion levels. Bonifaci et al. calculated the Price of Anarchy for bounded dynamic and static tolls Bonifaci *et al.* (2011). Dynamic and adaptive toll schemes that observe traffic conditions can improve equilibrium flow as successfully as marginal cost tolling Sharon *et al.* (2017). More recently, introducing tolls in multi-agent reinforcement learning has improved equilibrium efficiency for drivers with heterogeneous preferences Ramos *et al.* (2020). The waiting time function method we use to represent traffic lights are similar to tolls, since they both involve imposing additional costs on edges. However, in our model, the additional waiting times depend on properties of other incoming edges at a junction, thus, are distinctly different to tolls.

Frequently, game theoretic models are tested on a four-way intersection where each player represents the flow of vertical or horizontal traffic. Bui and Jung Bui & Jung (2017) use this format to apply the merge and split algorithm, from cooperative

game theory, with the aim that traffic light agents then negotiate time intervals at intersections. In addition, an intersection's traffic lights can be assumed to be a single agent, such as in work by Alvarez et al. Alvarez *et al.* (2008), who modelled traffic lights as players in a game where the payoff is gathered through minimising queue lengths.

Regulating the flow of traffic in complex road networks is an important application for artificial intelligence technologies, usually involving distributed optimisation and multi-agent learning methods. When modelling intelligent traffic light systems using reinforcement learning, the environment changes over time which can result in difficulties with algorithm convergence. The RL-CD algorithm da Silva *et al.* (2006) uses context detection to deal with the dynamic environment and also learns opponent behaviours. This type of learning may be too computationally expensive in practice, and we expect traffic lights to behave similarly due to their common goals. Hence, opponent modelling is less critical in this context. Wiering Wiering (2000) showed that a joint learning strategy for cars and traffic lights, that learn the same car-based value function, is significantly better at reducing waiting times than intelligent traffic systems without RL. This co-learning also improved the flow at intersections when compared to the MARL system without it. Bazzan and Klügl Bazzan & Klügl (2014) provide a summary paper of agent-based transport literature, highlighting the importance of scalable solutions.

Another approach to tackling intelligent traffic lights assumes that all vehicles are autonomous and can communicate with a central agent. Chouhan and Banda Chouhan & Banda (2018) consider a heuristic approach to intersection management, by adjusting the speed of vehicles through a central vehicle scheduler to avoid collisions. The heuristic algorithm is successful in minimising the average delay of vehicles in a variety of traffic densities in real-time. Similarly, Elhenawy et al. Elhenawy *et al.* (2015) adapt the game of chicken to reduce congestion at intersections by instructing cars to accelerate or decelerate. In the theoretical case of all autonomous vehicles, traffic lights can be replaced with a central coordination agent that advises speed adjustment to avoid collisions.

Despite the important contributions from the AI literature, we still do not know how intelligent traffic lights, adapting to self-interested drivers' behaviour, affect the equilibria. Therefore, we cannot fully assess the inefficiencies in resource usage caused by the underlying routing choices. Thus, we combine the nonatomic congestion game model with intelligent traffic lights to predict the effects of routing behaviours.

Current literature on adaptive traffic lights does not seek to consider the

disadvantages to their implementation, such as bias towards certain edges increasing journey times for drivers unfairly. However, Bertsimas et al. Bertsimas *et al.* (2012) have quantified the trade-off between fairness and efficiency in resource allocation, showing that deciding an objective function is essential in selecting the desired efficiency or fairness of solutions. Moreover, Lujak et al Lujak *et al.* (2015) encourage fair and envy-free allocation of routes through a traffic assignment algorithm that uses a normalised mean path duration cost. Here, we consider fairness of travel times by comparing envy-free and satisfaction levels within populations.

Our experimental results use the Simulation of Urban Mobility (SUMO) software Lopez *et al.* (2018) - a useful tool to model simple networks using microscopic simulation - due to its prevalence in the literature. For example, Mannion et al. Mannion *et al.* (2016) provide a review of applying reinforcement learning to traffic lights with experimental results presented using SUMO, highlighting the importance of appropriate state-space and reward selection. Moreover, Mousavi et al. Mousavi *et al.* (2017) implement deep policy-gradient methods for traffic signal control, and use the SUMO simulator to illustrate its reduced average journey times and queue lengths in practice. SUMO creates microscopic simulations for chosen network parameters and traffic demand. It uses a dynamic routing algorithm that estimates the user equilibrium in a nonatomic congestion game. We use SUMO to support the modelling assumptions of the traffic light game as well as to analyse the properties of intelligent traffic lights through SUMO simulated data.

6.4 Traffic Light Game

Introducing traffic lights at a junction periodically transforms the cost of waiting. In networks where traffic lights exist at junctions, we must represent the additional costs of using the preceding edge. We make the assumption that these costs are added to the edge costs of those which are directed into a junction where a traffic light is positioned. This can intuitively be divided into two parts: the cost of using the preceding edge if there was no traffic light at its end, and the additional cost that the traffic light imposes on the driver. Any edge that ends at a traffic light has a non-zero probability of having a nonnegative waiting time, due to encountering a red light. We make the simplifying assumption that traffic light waiting times at a junction are independent from other junctions.

Formally, we decompose the expected travel time of an edge c_e, into the expected travel time $\bar{c}_e$ that would occur from travelling across e, and the expected waiting time w_e that occurs at its end. The cost function for any edge e can,

therefore, be written as $c_e = \bar{c}_e + w_e$. For any edge e not ending with a traffic light, we have $w_e \equiv 0$. We continue by formulating a representation of traffic light cycles through w_e.

A traffic light cycle is formed of repeated phases of green and red light, when traffic is either serviced or not. Each direction of travel has a cycle of t_r^e red seconds and t_g^e green seconds. Between the phase changes, there is a period of amber light to warn drivers of the change. We will assume that the amber cycles are a constant time of 3 seconds throughout. Thus, without loss of generality, we include the period of amber light at the end of a green cycle in t_g^e and the amber cycle at the end of a red cycle is included in t_r^e. At the end of every edge there exists a probability, p_e, that a red light shows when the end node is reached. Thus, the waiting time is a function of p_e (as well as load f_e) where $p_e = \frac{t_r^e}{t_r^e + t_g^e} \in [0, 1]$. Finally, we define the total cycle time as $T_e := t_r^e + t_g^e$.

In general, the cost of waiting at a traffic light also depends on congestion levels. Here, we make the key assumption that we can separate the effects of congestion at a traffic light from that of normal travel, i.e., traffic lights only affect nearby congestion. For normal driving, congestion is created when the preceding car's speed is lower than the car travelling behind, whereas congestion at a traffic light junction depends on the number of cars waiting there and the traffic light cycle. In practice, there may not be conditional independence between the congestion of normal travel and at junctions, however, we choose to make this simplifying modelling assumption to allow for more detailed analysis of the effects of traffic lights.

Additionally, we make the following standard assumptions. Driving congestion functions $\bar{c}_e : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ are continuous, nondecreasing and nonnegative, while traffic light waiting functions $w_e : \mathbb{R}_{\geq 0} \times [0, 1] \to \mathbb{R}_{\geq 0} \cup \{\infty\}$ are continuous, nonnegative and nondecreasing in x; moreover, w_e is nondecreasing in p_e and if $p_e = 0$, then $w_e(f_e(\boldsymbol{x}), p_e) = 0$ and if $p_e \in (0, 1]$, then $w_e(f_e(\boldsymbol{x}), p_e) > 0$. Now we can write the cost function of using an edge in the general form of $c_e(f_e(\boldsymbol{x}), p_e) := \bar{c}_e(f_e(\boldsymbol{x})) + w_e(f_e(\boldsymbol{x}), p_e)$. As will be clear from the experiments in Appendix A, these assumptions are not only technically desirable but also justified empirically. In fact, they isolate the natural cost functions for nonatomic congestion games with traffic lights as emerging from the microscopic traffic simulation.

Definition 6.4.1 (Traffic light game)**.** *A* ***traffic light game*** *is a tuple*

$$\mathcal{M} = (N, (E_i), (S_i), (p_e), (c_e), (d_i)),$$

where $e \in E$ *and* $i \in N$*.*

The outcome of all players of population i choosing strategies leads to a strategy distribution $\boldsymbol{x}^i$ satisfying $\sum_{s\in S_i} x_s^i = d_i$ and $x_s^i \geq 0$, $\forall s \in S_i$. A strategy distribution or outcome $\boldsymbol{x} = (\boldsymbol{x}^i)_{i\in N}$ is *feasible* if $\sum_{s\in S_i} x_s^i = d_i$, $\forall i \in N$. Denote the *load* on e in an outcome $\boldsymbol{x}$ to be $f_e(\boldsymbol{x}) = \sum_{i\in N}\sum_{s\in S_i} x_s^i \mathbf{1}_s(e)$ where $\mathbf{1}$ is the indicator function. In $\boldsymbol{x}$, a player from population i receives a cost function $= C_i(s, \boldsymbol{x}, \boldsymbol{p}) = \sum_{e\in s} c_e(f_e(\boldsymbol{x}), p_e)$ when selecting strategy $s \in S_i$. The mixed strategy space for population i is denoted ΔS_i.

Definition 6.4.2 (Traffic light user equilibrium)**.** *A* ***traffic light user equilibrium*** *(TLUE) is a strategy distribution $\boldsymbol{x}$ such that all players choose a strategy yielding the minimum expected cost: $\forall i \in N$ and $s, s' \in S_i$ such that $x_s^i > 0$, we have $C_i(s, \boldsymbol{x}, \boldsymbol{p}) \leq C_i(s', \boldsymbol{x}, \boldsymbol{p})$.*

Note that a UE is a special case of a TLUE, where the traffic lights are always green upon arrival. As standard, we assume players have perfect information of the network, thus, also assume awareness of the probabilities p_e. This could be learned from past experience, through autonomous vehicles, etc. The next result follows, as with standard nonatomic congestion games (see e.g., Smith (1979)).

In the following proposition, we use the notation $\boldsymbol{x}C(\boldsymbol{x}, \boldsymbol{p})$ or $C(\boldsymbol{x}, \boldsymbol{p})\boldsymbol{x}$ as shorthand for

$$\sum_{i\in N}\sum_{s\in S_i} x_s^i C_i(s, \boldsymbol{x}, \boldsymbol{p}).$$

Proposition 6.4.3. *Feasible strategy distribution $\boldsymbol{x}$ is a TLUE solution if, and only if, for any feasible strategy distribution $\boldsymbol{x'}$,*

$$C(\boldsymbol{x}, \boldsymbol{p})(\boldsymbol{x} - \boldsymbol{x}') \leq 0.$$

Proof. Suppose that $\boldsymbol{x}$ is a TLUE. Let population i play strategy s in $\boldsymbol{x}$ with a positive probability: $\boldsymbol{x}_s^i > 0$. Then any strategy $s' \in S_i$, that has a higher cost than $C_i(s, \boldsymbol{x}, \boldsymbol{p})$, does not occur in a TLUE. That is,

$$C_i(s', \boldsymbol{x}, \boldsymbol{p}) > C_i(\boldsymbol{x}, \boldsymbol{p}) \Rightarrow \boldsymbol{x}_{s'}^i = 0$$

Hence, for any feasible strategy distribution $\boldsymbol{x}'$, the route cost is at least as high as $\boldsymbol{x}$,

$$\sum_{s\in S_i} x_s'^i C_i(s, \boldsymbol{x}, \boldsymbol{p}) \geq \sum_{s\in S_i} x_s^i C_i(s, \boldsymbol{x}, \boldsymbol{p}).$$

Since this is true $\forall i \in N$, we can sum over N to get

$$\boldsymbol{x}'C(\boldsymbol{x}, \boldsymbol{p}) \geq \boldsymbol{x}C(\boldsymbol{x}, \boldsymbol{p}) \ \forall \boldsymbol{x}',$$

as claimed.

Now suppose the converse is true, that $\boldsymbol{x}$ is not a TLUE. So, there exists a feasible strategy distribution $\boldsymbol{x}'$, a unilateral deviation from $\boldsymbol{x}$, that costs less than $\boldsymbol{x}$. Then there exists nonempty $M \subseteq N$ such that $i \in M$ if, and only if,

$$\exists s' \in S_i \text{ s.t. } x'^i_{s'} > 0,\ C_i(s', \boldsymbol{x}, \boldsymbol{p}) < C_i(\boldsymbol{x}, \boldsymbol{p}).$$

If $i \in M$ reroutes their flow along these cheaper routes, then it will reduce the total cost by

$$\sum_{s \in S_i} x^i_s C_i(s, \boldsymbol{x}, \boldsymbol{p}) - \sum_{s \in S_i} x'^i_s C_i(s, \boldsymbol{x}, \boldsymbol{p}) > 0.$$

Thus, $\sum_{s \in S_i} x'^i_s C_i(s, \boldsymbol{x}, \boldsymbol{p}) < \sum_{s \in S_i} x^i_s C_i(s, \boldsymbol{x}, \boldsymbol{p})$, for $i \in M$. For $j \in N \backslash M$, since $\forall s \in S_j$ where $x'^j_s > 0$, $C_j(s', \boldsymbol{x}, \boldsymbol{p}) \geq C_j(\boldsymbol{x}, \boldsymbol{p})$, we have that $\boldsymbol{x}'^j_s = \boldsymbol{x}^j_s$.

So, for any $\boldsymbol{x}$ where there is an incentive to unilaterally deviate to $\boldsymbol{x}'$ for population $i \in N$, we have

$$\sum_{s \in S_i} x'^i_s C_i(s, \boldsymbol{x}, \boldsymbol{p}) < \sum_{s \in S_i} x^i_s C_i(s, \boldsymbol{x}, \boldsymbol{p}),$$

and for all $j \in N$, where $j \neq i$,

$$\sum_{s \in S_j} x'^j_s C_j(s, \boldsymbol{x}, \boldsymbol{p}) = \sum_{s \in S_j} x^j_s C_j(s, \boldsymbol{x}, \boldsymbol{p}).$$

Consequently, we have that

$$\boldsymbol{x}' C(\boldsymbol{x}, \boldsymbol{p}) < \boldsymbol{x} C(\boldsymbol{x}, \boldsymbol{p}).$$

Hence, the inequality does not hold if $\boldsymbol{x}$ is not a TLUE, so the statements are equivalent, as claimed. □

Note that for fixed traffic light control, this is a user equilibrium, for which the results on the existence and computation trivially hold.

From Proposition 6.4.3, we can readily derive that for any solution $\boldsymbol{x}$ and any feasible $\tilde{\boldsymbol{x}}$, we have $C(\boldsymbol{x}, \boldsymbol{p})(\tilde{\boldsymbol{x}} - \boldsymbol{x}) \geq 0$, which can be rewritten as follows.

$$\sum_{i \in N} \sum_{s \in S_i} C_i(s, \boldsymbol{x}, \boldsymbol{p})(\tilde{\boldsymbol{x}}^i_s - \boldsymbol{x}^i_s) \geq 0$$

$$\sum_{i \in N} \sum_{s \in S_i} \sum_{e \in s} c_e(f_e(\boldsymbol{x}))(\tilde{\boldsymbol{x}}^i_s - \boldsymbol{x}^i_s) \geq 0$$

$$\sum_{i\in N}\sum_{s\in S_i}\sum_{e\in s}[\bar{c}_e(f_e(\boldsymbol{x}))+w_e(f_e(\boldsymbol{x}),p_e)](\tilde{\boldsymbol{x}}_s^i-\boldsymbol{x}_s^i)\geq 0$$

$$\sum_{e\in E}\sum_{i\in N}\sum_{s\in S_i}[\bar{c}_e(f_e(\boldsymbol{x}))+w_e(f_e(\boldsymbol{x}),p_e)](\tilde{\boldsymbol{x}}_s^i-\boldsymbol{x}_s^i)\geq 0$$

$$\sum_{e\in E}[\bar{c}_e(f_e(\boldsymbol{x}))+w_e(f_e(\boldsymbol{x}),p_e)](f_e(\tilde{\boldsymbol{x}})-f_e(\boldsymbol{x}_s^i))\geq 0$$

This is equivalent to the following minimisation problem, since $\boldsymbol{x}$ is a TLUE by Proposition 6.4.3,

$$\min_{\boldsymbol{x}}\sum_{e\in E}\int_0^{f_e(\boldsymbol{x})}\bar{c}_e(\boldsymbol{z})+w_e(z,p_e)dz,\quad \text{or}\quad \min_{\boldsymbol{x}}\sum_{e\in E}\int_0^{f_e(\boldsymbol{x})}c_e(\boldsymbol{z})dz.$$

This means that if cost functions are monotonic and differentiable, as they are in our case, a solution exists and is essentially unique, i.e., all TLUE have the same social cost.

We note at this point that a key difference of our framework with previous ones in the literature (see for instance Smith (1985)), is that here we do not assume that the cost functions must tend to infinity as the congestion or the proportion of red time reach their upper limit.

The model so far has involved the addition of a further cost function to edges, which has a key similarity with the literature on tolls (see eg. Karakostas & Kolliopoulos (2009); Sandholm (2002)). However, our model focuses on waiting functions w_e that correspond to their out direction node by the cycle parameters p_e. For a set of edges ending at v denoted E_v, $\sum_{e\in E_v}(1-p_e)=1$. Thus, we do not have the same independence properties as tolls. This difference implies that existing analyses of networks with tolls cannot be applied to our setting in a straightforward way. Additionally, we argue that optimising light cycles is more socially acceptable and more easily implemented than incurring tolls on all roads in traffic networks.

To this end, let us consider the realistic case of traffic lights with finite waiting times: $\max_{e\in E} w_e<\infty$. Suppose there exists $e\in E$ such that $p_e=1$, then there will be no green light time for edge e in order for players to move across the junction. Yet, $w_e(f_e(\boldsymbol{x}),p_e)<\infty$. Thus, for finite waiting times, there must exist some bounds $[p_-,p_+]$ on the values of p such that players are always able to cross junctions within a finite time, where $p_->0$ and $p_+<1$.

Suppose that the minimum period of time required for a car to be able to pass through a junction is t_{min}. Then there exists an upper bound p_+ for a car to be able to pass through the junction for a traffic light cycle of length t_r+t_g. The

upper bound on the proportion of red time is $p_+ := 1 - \frac{t_{min}}{t_r+t_g}$. For any $p \in (p_+, 1]$, there will not be sufficient time in the cycle for any cars to pass through, and thus all waiting times for this edge would be infinite.

Since p has an upper bound, it must also have a lower bound p_-, as there are at least two edges at each traffic light node and the proportions of green times associated with a node should sum to 1. This lower bound is defined as $p_- := \frac{t_{min}}{t_r+t_g}$. For example, if the total cycle time of a traffic light was 40 seconds and the minimum time required for a car to pass through a traffic light junction safely was 6 seconds, then we would require $p \in [0.15, 0.85]$. Besides being realistic, finite waiting times induce their own efficiency properties and will be compared against later.

A simple simulation in SUMO with a single population is included in the Appendix A. We include this to show compatibility with SUMO and that the assumptions we made in the model hold in the simple example included. Note that we mostly consider example networks with a single population throughout this chapter since they are easier to interpret. Yet, all results apply to multiple populations unless stated otherwise.

We will now continue by analysing the effect traffic lights have on routing inefficiency, namely, Braess' paradox.

6.5 Braess' Paradox and Traffic Lights

It is clear that changing traffic light cycles have a big impact on the routing choices of drivers. In this section, we show that biased traffic lights can force optimal or suboptimal solutions.

Suppose that there exists a traffic light at a node v if there at least two edges entering v. If we had a Wheatstone network, we would add a single traffic light. We will use this network as a motivating example to show how the traffic lights enable Braess' paradox. Figure 6.1 shows this example, where a square node represents the presence of a traffic light.

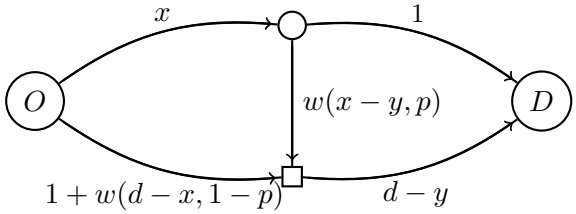

Figure 6.1: The Wheatstone network with a single traffic light.

Example 5. *Consider the directed Wheatstone graph with a traffic light as in Figure 6.1. We can write the cost functions for the edges as follows.*

$$\begin{aligned} c_{e_1}(x) &= \bar{c}_{e_1}(x) \\ c_{e_2}(x) &= \bar{c}_{e_2}(x) + w_{e_1}(x, 1-p) \\ c_{e_3}(x) &= \bar{c}_{e_3}(x) + w_{e_3}(x, p) \\ c_{e_4}(x) &= \bar{c}_{e_4}(x) \\ c_{e_5}(x) &= \bar{c}_{e_5}(x) \end{aligned}$$

Suppose that there exists a TLUE $\boldsymbol{x}$*, where* $f_{e_1}(\boldsymbol{x}) = x$*,* $f_{e_2}(\boldsymbol{x}) = d - x$*,* $f_{e_3}(\boldsymbol{x}) = x - y$*,* $f_{e_4}(\boldsymbol{x}) = y$ *and* $f_{e_5}(\boldsymbol{x}) = d - y$ *for some* $x, y \in [0, d]$ *where* $y \leq x$*.*

If we increase p*, it causes* w_{e_2} *to decrease, therefore, its edge load* $d - x$ *will increase. In addition,* w_{e_3} *increases, hence, it is less desirable. So,* $x - y$ *decreases. By increasing* p *to a large value, we can increase the costs of using the central edge to make its use undesirable to drivers. Combining large* p *with the congestion costs from the Braess example limits players to the subgame where the social optimum solution is the TLUE.*

In contrast, if we reduce p*, then* w_{e_3} *decreases, which causes the edge load* $x - y$ *to increase. Furthermore,* w_{e_2} *increases, hence, its load* $d - x$ *decreases. For very low values of* p *combined with the Braess example cost functions, the TLUE here is for the players to all use the strategy* $\{e_1, e_3, e_5\}$*.*

By adding only one traffic light into the Wheatstone network, we have shown that changing the proportions of red and green time of the traffic light can induce routing inefficiencies; worsening or reducing the effects of Braess' paradox can be caused by traffic lights. Consequently, the traffic light cycles can be used to force more socially optimal outcomes in addition to enabling poor routing choices. Example 5 shows the importance of traffic light cycles in urban areas as a tool to reduce congestion.

Now consider what happens when we add another edge and traffic light giving a slightly more complex network.

Example 6. *Suppose that we add another edge and traffic light into the Wheatstone network. The new network with 6 directed edges and 2 traffic lights is shown in Figure 6.2. We write the cost functions as follows.*

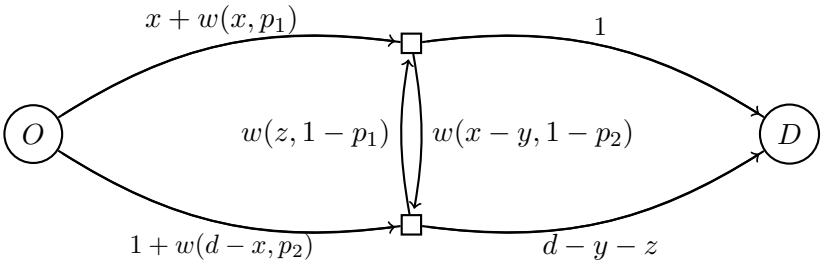

Figure 6.2: Avoiding or inducing Braess' paradox using traffic lights.

$$
\begin{aligned}
c_{e_1}(x) &= \bar{c}_{e_1}(x) + w_{e_1}(x, p_1) \\
c_{e_2}(x) &= \bar{c}_{e_2}(x) + w_{e_2}(x, p_2) \\
c_{e_3}(x) &= \bar{c}_{e_3}(x) + w_{e_3}(x, 1 - p_2) \\
c_{e_4}(x) &= \bar{c}_{e_4}(x) + w_{e_4}(x, 1 - p_1) \\
c_{e_5}(x) &= \bar{c}_{e_5}(x) \\
c_{e_6}(x) &= \bar{c}_{e_6}(x)
\end{aligned}
$$

Suppose that we have a TLUE where $f_{e_1}(\boldsymbol{x}) = x$, $f_{e_2}(\boldsymbol{x}) = d - x$, $f_{e_3}(\boldsymbol{x}) = x - y$, $f_{e_4}(\boldsymbol{x}) = z$, $f_{e_5}(\boldsymbol{x}) = y + z$ *and* $f_{e_6}(\boldsymbol{x}) = d - y - z$, *for some constants* $x, y, z \geq 0$. *If we have both* p_1 *and* p_2 *as near-zero values, then it makes using edges* e_3 *and* e_4 *very costly. In this case, we can force the social optimum solution as the user equilibrium using traffic lights. With other combinations of* p_1 *and* p_2 *you will get a TLUE that has higher social cost than the social optimum. Thus, the combinations of traffic light cycles can create immunity to Braess' paradox.*

Traffic lights have the capacity to act as a central decision-maker and remove the effects of noncooperative selfish routing. However, in Appendix A we showed that there are some restrictions on the maximum and minimum possible values of p. Therefore, in real-life scenarios we may not be able to achieve immunity to Braess' paradox through traffic light bias, only reduce it's impact.

A two-terminal network suffers from Braess' paradox if it is not series-parallel. In order to remove the affects of Braess' paradox, we must enforce the use of a subnetwork which does not suffer from Braess' paradox. In order to do that, we must find a set of "undesirable" edges to "remove" so that the new network does not suffer from Braess' paradox.

We have the additional constraint that the edges sharing the same end node

must have their proportions of green times summing to 1. Thus, an allocation of probabilities is *feasible* if $\sum_{e \in v_{in}} (1 - p_e) = 1$. For graphs with a maximum in-degree of two, we can easily set up a profile of feasible $\boldsymbol{p}$ to remove the occurrence of Braess' paradox.

Proposition 6.5.1. *For any two-terminal directed network $G = (V, E)$, where the maximum in-degree of a node is two, there exists a nonempty set of feasible allocations of $(p_e)_{e \in E}$ such that G is immune to Braess' paradox.*

Proof. Choose an allocation $\boldsymbol{p}$ such that $p_i = 1$ and $p_j = 0$ for every pair of edge e_i and e_j, where the in-degree of the common node is two. By construction, there exists only one path between the origin and destination, hence, the network cannot suffer from Braess' paradox. □

If we allow for multiedge networks (where there exists multiple edges connecting the same two nodes in the same direction) then the feasibility constraint restricts the values of p_e at edges. To specify general networks that are immune to Braess' paradox, define $\hat{E}$ to be a minimal set of edges such that $(V, E \backslash \hat{E})$ is series-parallel.

Proposition 6.5.2. *For any two-terminal network that suffers from Braess' paradox, there exists a nonempty set of traffic light cycles $(p_e)_{e \in E}$ which make the network immune to Braess' paradox.*

Proof. Let the maximal subgraph of G that is series-parallel be $\hat{G} = (V, E \backslash \hat{E})$. For any edge $\hat{e} \in \hat{E}$, there must exist a traffic light. (This is true since in order for the edge not to be added in series or in parallel, it must be formed from existing nodes. The in-degree of the node must already be one, so by adding a non-series-parallel edge we must have a traffic light at the end node.) For any such $\hat{e}$, set $p_{\hat{e}} = 1$. For any edge $e \in E \backslash \hat{E}$, set $p_e = \frac{1}{v_{in} - v_{\hat{e}}}$ where $v_{\hat{e}}$ is the number of edges that end at node v that belong to $\hat{E}$. By construction, the accessible network $\hat{G}$ is series-parallel, hence, it is immune to Braess' paradox. □

The proof of Proposition 6.5.2 works by essentially closing roads through enforcing infinite waiting times on edges which tempt players to choose suboptimal paths. We can extend this result to a nonatomic congestion game played on any general network using a similar method.

Theorem 6.5.3. *For any asymmetric network, there exists a nonempty set of traffic light cycles $(p_e)_{e \in E}$ which make the network immune to Braess' paradox.*

Proof. Let a subgraph of G that is immune to Braess' paradox be $\tilde{G} = (V, \tilde{E})$. To prove such a $\tilde{G}$ exists, we can use Theorem 2.3.5 which states that a nonatomic congestion game with matroidal strategy sets is immune to Braess' paradox.

Let $\tilde{G}_i \subset G$ be a directed spanning tree of population i. The network $\tilde{G} := \bigcup_{i \in N} \tilde{G}_i$ gives rise to the tuple $(\tilde{E}, (S_i)_{i \in N})$, which forms the base set of a matroid. Hence, $\tilde{G}$ is a subgraph of G such that all $|S_i| = 1$ and is immune to Braess' paradox. Note, this network has the minimum number of edges such that all populations have a strategy and does not embed any network which would make it vulnerable to Braess' paradox.

Next we claim that for any edge in $e \in E \backslash \tilde{E}$, there must exist a traffic light. To see this, first note that G is irredundant, so all edges must be used in at least one strategy set. Since $\tilde{G}$ contains a spanning tree for all populations, e must end at the same node as an edge in $\tilde{E}$. Hence, the in-degree of the node at which it ends is at least two. For any such e, set $p_e = 1$.

For any edge $\tilde{e} \in \tilde{E}$, set $p_{\tilde{e}} = 1 - \frac{1}{v_{in} - v_e}$, where v_e is the number of edges that end at v and belong to $E \backslash \tilde{E}$. By construction, the network $\tilde{G}$ is series-parallel, hence, it is immune to Braess' paradox. □

While this result has theoretical interest, real-world traffic lights rarely function on infinite waiting times and it is therefore necessary to study more realistic cases, when we may not be able to achieve immunity to Braess' paradox through traffic light bias, but only reduce its impact. For example, this improvement is restricted by the maximal in-degree of a node. To see this, consider the network in Figure 6.3. By setting "undesirable" edges at p_+ it means that the minimum value of p for "desirable" edges increases.

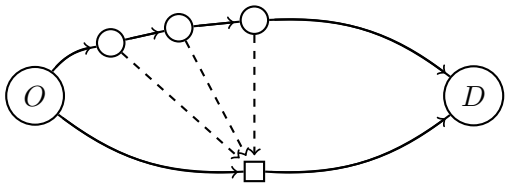

Figure 6.3: In this network, the dashed lines need to be removed in order for immunity to Braess' paradox. It is not possible to allocate $p_{\hat{e}} \leq p_+$ such that the network has immunity to Braess' paradox.

Nevertheless, we can find cases where immunity occurs. Define $\hat{G} = (V, E \backslash \hat{E})$ and $v_{\hat{e}}$ as in the proof of Theorem 6.5.3. Then we suggest the following condition on w_e that guarantees immunity to Braess' paradox for finite waiting times.

Proposition 6.5.4. *For any asymmetric network, there exists a nonempty set of bounded traffic light cycles $(p_e)_{e\in E}$ such that the game is immune to Braess' paradox if, $\hat{E} = \emptyset$, or $\forall \hat{e} \in \hat{E}$ and $\forall e \in E\backslash\hat{E}$*

$$c_e(\sum_{i\in N} d_i, p_e^-) < w_{\hat{e}}(0, p_+),$$

where $p_e^- := \frac{v_{\hat{e}}(1-p_+)-1}{(v_{in}-v_{\hat{e}})} + 1$.

Proof. For any $\hat{e} \in \hat{E}$, set $p_{\hat{e}} = p_+$. For any edge $e \in E\backslash\hat{E}$, let $p_e = p_e^-$. Thus, $\forall v \in V$, $\sum_{e\in E_v}(1 - p_e) = 1$. By construction, as $p_+ \to 1$, the network $\hat{G}$ becomes series-parallel.

For the equilibrium flow $\boldsymbol{x}$ to route along this series-parallel network, we need that $\forall \hat{e} \in \hat{E}$, and $\forall e \in E\backslash\hat{E}$, $c_e(f_e(\boldsymbol{x}), p_e) < c_{\hat{e}}(f_{\hat{e}}(\boldsymbol{x}), p_{\hat{e}})$. Therefore, for small $\epsilon > 0$,

$$\bar{c}_e(f_e(\boldsymbol{x})) + w_e(f_e(\boldsymbol{x}), p_e^-) < \bar{c}_{\hat{e}}(\epsilon) + w_{\hat{e}}(\epsilon, p_+).$$

This holds if $\max_{\boldsymbol{x}}\{c_e(f_e(\boldsymbol{x}), p_e^-)\} < w_{\hat{e}}(0, p_+)$. Thus, $\boldsymbol{x}$ exists in $\hat{G}$, i.e. is immune to Braess' paradox, for

$$c_e(\sum_{i\in N} d_i, p_e^-) < w_{\hat{e}}(0, p_+).$$

□

Theorem 6.5.3 and Proposition 6.5.4 specify instances where immunity to Braess' paradox can occur. However, the underlying assumptions may not hold in more realistic problems. To indicate how routing efficiency is effected by traffic light cycles in typical networks, we include the following examples on two-terminal and asymmetric networks.

Example 7 (Two-terminal game)**.** *Consider the Wheatstone network with one traffic light and cost functions as in Figure 6.1, where $d = 1$. To find the socially optimum solution we must solve the following minimisation problem.*

$$\min_{x,y}[x^2 + (1-x) + (1-x)w(1-x, p) + (x-y)w(x-y, 1-p) + y + (1-y)^2],$$

where $x \in [0, 1]$ and $y \leq x$. Suppose the waiting functions are of the simplest exponential form: $w(x, p) = x(e^p - 1)$. We use the Lagrangian method to solve for optimal x and y.

$$L(x, y, \lambda) = x^2 + (x-y)^2(e^{1-p} - 1) + (1-x)^2(e^p - 1) + y + (1-y)^2 - \lambda(x-y)$$

This amounts to the following Karush-Kuhn-Tucker conditions.

$$\begin{aligned}\frac{dL}{dx} &= 2x + 2(x-y)(e^{1-p}-1) - 2(1-x)(e^{p}-1) - \lambda = 0\\ \frac{dL}{dy} &= -2(x-y)(e^{1-p}-1) + 1 + 2(1-y) + \lambda = 0\\ \frac{dL}{d\lambda} &= y - x \leq 0 \qquad \lambda(y-x) = 0\end{aligned}$$

If $x \neq y$, the Karush-Kuhn-Tucker conditions are not solvable. When $x = y$, we solve then equations to get $x = y = \frac{e^p}{1+e^p}$. This tells us that at the social optimum, no players use the middle edge. For $p = 0$, we get the social optimum of $x = y = 1/2$ (the same solution as the case without traffic lights). Figure 6.4 shows that the social cost functions of user equilibria and social optima are strictly increasing in p, so by choosing p as low as possible, we still maximally reduce the costs of routing. There is no instance of p where the user equilibrium solution is equal to the social optimum. The TLUE never reaches the social optimum cost since the waiting time function is finite, so there are always some players using the middle edge.

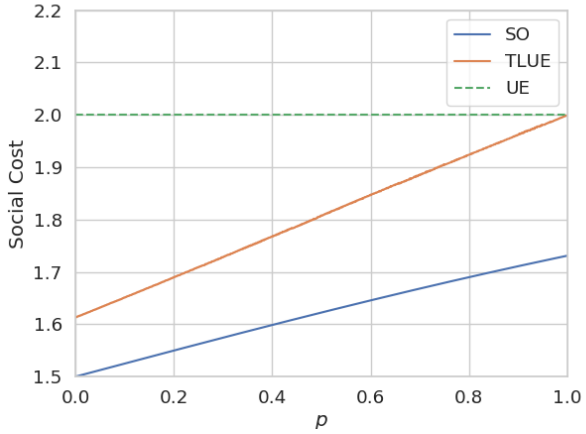

Figure 6.4: The social cost of the TLUE and social optimum on the Wheatstone network with a single traffic light. The equilibrium costs are strictly less than the same network without a traffic light due to the altered routing behaviours, here denoted UE.

The UE cost of this network without a traffic light (see Figure 2.1) is 2. Therefore, the traffic light is able to reduce the effects of selfish routing for all $p < 1$. By including a traffic light, the cost of the TLUE is up to 24% lower than without. Even with the additional constraint on p to adhere to its bounds $1 - p_+ < p < p_+$,

the social cost is reduced by 22% when $p_+ = 0.85$. This suggests that by positioning traffic lights at the end of edges which are not series-parallel, the costs of selfish routing are reduced. The Price of Anarchy belongs to $[1.07, 1.15]$ for this game, whereas once the traffic light and waiting time cost functions are removed, the Price of Anarchy is, as well known (see e.g., Roughgarden (2005)), strictly greater: $4/3$.

Thus, we have improved the efficiency of the Wheatstone network by including a traffic light.

Let us now work through a slightly more complex example to find the impact $\boldsymbol{p}$ has on Braess' paradox in a game with multiple populations.

Example 8 (Asymmetric game)**.** *To show that these effects scale in larger networks, consider the multipopulation game shown in Figure 6.5.*

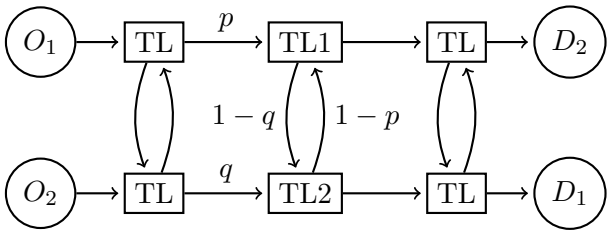

Figure 6.5: Consider a multipopulation game with 6 traffic lights. There are 4 static traffic lights (indicated as TL) with equally proportioned red and green cycle. The two middle traffic lights, TL1 and TL2 have changeable cycles of p and q respectively i.e. can set $p, q \in [p_-, p_+]$.

Here, there are two populations of players who both travel in similar direction across a grid. Suppose the cost of each edge is the same: $c_e(x, p) = xe^p$ for edges with a traffic light present; and, $c_e(x) = x$ otherwise. Let each population have a demand of 1. We assume that traffic lights are naturally set fairly, i.e. $p = 1/2$. Consider how changing the traffic light cycles, p for TL1 and q for TL2, affects the social cost of the game.

The results of the social cost to the populations at simulated TLUE are shown in Figure 6.6. A fair traffic light with $p = 0.5$ and $q = 0.5$ gives a social cost of 12.99. Compare this to the optimal traffic light phases of $p = 0.85$ and $q = 0.85$, which costs 12.14. If we consider the game where traffic light TL1 and TL2 do not exists and the middle edges are removed, this game has a social cost of 12.24 and is immune to Braess' paradox. Therefore, in order to have a game which is immune to Braess' paradox with TL1 and TL2, we want to choose a traffic light cycle which

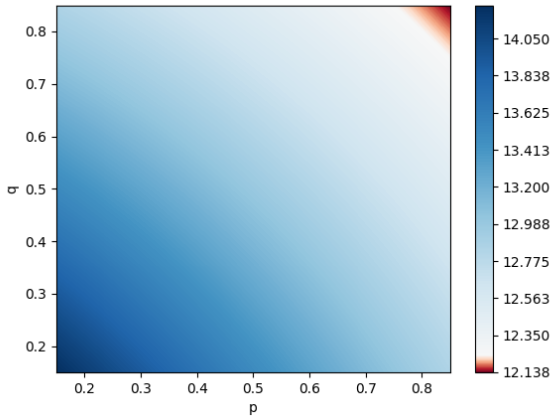

Figure 6.6: The social cost of the TLUE for varying p and q in the multipopulation game. The red region represent the values of p and q which would need to be set in order for the network to be immune to Braess' paradox and the blue region shows the region where adding the traffic light cycles allow for Braess' paradox to occur.

has a social cost of at most 12.24. The phases for which this occur are indicated in red in Figure 6.6.

Hence, we have found values of p and q that are compatible with immunity to Braess' paradox for an asymmetric network. Since many real-world networks have a similar grid-like road structure, these results could also occur in more realistic networks.

In conclusion, we have shown that traffic light cycles can create immunity to Braess' paradox in theory and reduce its impact in practice. To measure the impact of Braess' paradox on a network we could use the Braess ratio, which measures the largest reduction to the total cost of travel at equilibrium flow from the removal of an edge or a set of edges. However, the more common metric to compare network routing inefficiency is the Price of Anarchy. As such, we will now continue to address the effects of traffic lights in nonatomic congestion games by calculating their Price of Anarchy.

6.6 Price of Anarchy

In this section, we consider how the waiting time function affects the ratio of the Nash equilibrium social cost to the social optimum cost, i.e., the Price of Anarchy.

To begin, consider general $\bar{c}_e$ and w_e. Since $\bar{c}_e$ and w_e can be from different classes of functions, observe how (λ, μ)-smoothness will affect the Price of Anarchy.

Proposition 6.6.1. *Suppose $\bar{c}_e$ is (λ_1, μ_1)-smooth and w_e is (λ_2, μ_2)-smooth. Then*

$$PoA = \frac{\max(\lambda_1, \lambda_2)}{1 - \max(\mu_1, \mu_2)}.$$

Proof. Let the social optimum be $\boldsymbol{x}^*$. By smoothness properties, we can bound the social cost of the UE $\boldsymbol{x}$

$$SC(\boldsymbol{x}, p) \leq \lambda_1 \sum_{e \in E} \bar{c}_e(\boldsymbol{x}^*) + \mu_1 \sum_{e \in E} \bar{c}_e(\boldsymbol{x}) + \lambda_2 \sum_{e \in E} w_e(\boldsymbol{x}^*, p) + \mu_2 \sum_{e \in E} w_e(\boldsymbol{x}, p) \quad (6.1)$$

Now we can write equation 6.1 in terms of the social costs of $\boldsymbol{x}$ and $\boldsymbol{x}^*$,

$$SC(\boldsymbol{x}, p) \leq \max(\lambda_1, \lambda_2) SC(\boldsymbol{x}^*, p) + \max(\mu_1, \mu_2) SC(\boldsymbol{x}, p) \quad (6.2)$$

By rearranging equation 6.2 we see the ratio gives us the specified Price of Anarchy.

$$(1 - \max(\mu_1, \mu_2)) SC(\boldsymbol{x}, p) \leq \max(\lambda_1, \lambda_2) SC(\boldsymbol{x}^*, p)$$

$$\frac{SC(\boldsymbol{x}, p)}{SC(\boldsymbol{x}^*, p)} \leq \frac{\max(\lambda_1, \lambda_2)}{(1 - \max(\mu_1, \mu_2))}$$

□

We can specify an upper bound on the Price of Anarchy irrespective of traffic light cycles in this way. Proposition 6.6.1 finds the worst-case Price of Anarchy for any traffic light cycles. However, if we allow $p \in [0, 1]$, then w_e can be any value between 0 and ∞. As such, previous results from the tolls literature that state marginal cost tolls achieve social optimum equilibria hold Beckmann *et al.* (1956). If we have the additional constraint that $p_e \in (0, 1)$, then we can find a tighter bound dependent on the bounded function w_e.

To find the Price of Anarchy, we follow a similar argument to Correa et al. Correa *et al.* (2005) by using a function β such that the Price of Anarchy is equal to $(1 - \beta)^{-1}$. In order to calculate β, we must find a relationship between the social costs of the user equilibrium and social optimum as we have done for (λ, μ)-smooth functions.

First, let $\boldsymbol{x}$ be a TLUE and $\tilde{\boldsymbol{x}}$ be a feasible strategy distribution. Then, by Proposition 6.4.3, we have

$$\sum_{e \in E} c_e(f_e(\boldsymbol{x}), p_e) f_e(\boldsymbol{x}) \leq \sum_{e \in E} c_e(f_e(\boldsymbol{x}), p_e) f_e(\tilde{\boldsymbol{x}}) \quad (6.3)$$

We can expand inequality 6.3 on the right hand side as

$$\sum_{e\in E} c_e(f_e(\boldsymbol{x}),p_e)f_e(\boldsymbol{x}) \leq \sum_{e\in E} c_e(f_e(\tilde{\boldsymbol{x}}),p_e)f_e(\tilde{\boldsymbol{x}}) + \sum_{e\in E}[c_e(f_e(\boldsymbol{x}),p_e) - c_e(f_e(\tilde{\boldsymbol{x}}),p_e)]f_e(\tilde{\boldsymbol{x}}).$$

Now, let us define a function β, where we use the convention $0/0 = 0$ as in Correa *et al.* (2005), $\beta(\bar{c}_e, w_e, p_e) :=$

$$\sup_{f_e(\boldsymbol{x}),f_e(\tilde{\boldsymbol{x}})\geq 0} \frac{(\bar{c}_e(f_e(\boldsymbol{x})) + w_e(f_e(\boldsymbol{x}),p_e) - \bar{c}_e(f_e(\tilde{\boldsymbol{x}})) - w_e(f_e(\tilde{\boldsymbol{x}}),p_e))f_e(\tilde{\boldsymbol{x}})}{(\bar{c}_e(f_e(\boldsymbol{x})) + w_e(f_e(\boldsymbol{x}),p_e))f_e(\boldsymbol{x})}$$

And for sets of functions $\mathcal{C}_1$ and $\mathcal{C}_2$,

$$\beta(\mathcal{C}_1,\mathcal{C}_2,\boldsymbol{p}) := \sup_{\bar{c}_e\in\mathcal{C}_1, w_e\in\mathcal{C}_2} \beta(\bar{c}_e, w_e, p_e).$$

Then,

$$SC(\boldsymbol{x}) \leq SC(\tilde{\boldsymbol{x}}) + \beta(\mathcal{C}_1,\mathcal{C}_2,\boldsymbol{p})SC(\boldsymbol{x}).$$

Thus, the Price of Anarchy is

$$\frac{1}{1-\beta(\mathcal{C}_1,\mathcal{C}_2,\boldsymbol{p})}. \tag{6.4}$$

Now, let us define $\eta = \frac{f_e(\tilde{\boldsymbol{x}})}{f_e(\boldsymbol{x})}$ and $\zeta_e(p_e) = \frac{w_e(f_e(\boldsymbol{x}),p_e)}{\bar{c}(f_e(\boldsymbol{x}))}$. Then, the expression

$$\sup_{f_e(\boldsymbol{x}),f_e(\tilde{\boldsymbol{x}})\geq 0} \frac{(\bar{c}_e(f_e(\boldsymbol{x})) + w_e(f_e(\boldsymbol{x}),p_e) - \bar{c}_e(f_e(\tilde{\boldsymbol{x}})) - w_e(f_e(\tilde{\boldsymbol{x}}),p_e))f_e(\tilde{\boldsymbol{x}})}{(\bar{c}_e(f_e(\boldsymbol{x})) + w_e(f_e(\boldsymbol{x}),p_e))f_e(\boldsymbol{x})},$$

can be simplified using η and ζ:

$$\beta(\bar{c}_e, w_e, p_e) = \sup_{\eta\geq 0} \frac{1+\zeta_e(p_e) - \frac{\bar{c}_e(f_e(\tilde{\boldsymbol{x}}))}{\bar{c}_e(f_e(\boldsymbol{x}))} - \frac{w_e(f_e(\tilde{\boldsymbol{x}}),p_e)}{\bar{c}_e(f_e(\boldsymbol{x}))}}{1+\zeta_e(p_e)}\eta.$$

This becomes

$$\beta(\bar{c}_e, w_e, p_e) = \sup_{\eta\geq 0} \frac{1+\zeta_e(p_e) - \bar{c}_e(\eta) - \zeta_e(p_e)w_e(\eta,p_e)}{1+\zeta_e(p_e)}\eta.$$

Thus, we write β in the following form

$$\beta(\bar{c}_e, w_e, p_e) = \sup_{\eta\geq 0}\Big(1 - \frac{\bar{c}_e(\eta) + \zeta_e(p_e)w_e(\eta,p_e)}{1+\zeta_e(p_e)}\Big)\eta. \tag{6.5}$$

Suppose that $\bar{c}_e$ and w_e are polynomial of degree d_1 and d_2 respectively. So, we can write

$$\beta(\bar{c}_e, w_e, p_e) = \sup_{\eta \geq 0} \Big(1 - \frac{\eta^{d_1} + \zeta_e(p_e)\eta^{d_2}\bar{w}_e(p_e)}{1 + \zeta_e(p_e)}\Big)\eta \tag{6.6}$$

where $\bar{w}_e$ is some function such that $w_e(x, p) = x^{d_1}\bar{w}_e(p)$ and $0 < t_- \leq \bar{w}_e(p_e) \leq t_+ < \infty$ for $p_e \in [p_-, p_+]$.

To find an upper bound on the Price of Anarchy we need to find the maximum value of β. This maximum is achieved at

$$1 - \frac{(d_1 + 1)}{1 + \zeta_e(p_e)}\eta^{d_1} - \zeta_e(p_e)\frac{(d_2 + 1)}{1 + \zeta_e(p_e)}t_-\eta^{d_2} = 0.$$

Let $d = d_1 = d_2$, then the maximum occurs when

$$\eta^* = \Big(\frac{1 + \zeta_e(p_e)}{(d + 1)(1 + \zeta_e(p_e)t_-)}\Big)^{1/d}.$$

For $t_- \geq 1$, as ζ_e decreases β increases. Since we want the upper bound on β, we choose the minimum ζ_e and, since w_e is nondecreasing in p, we find the minimum value at the lower bound p_-:

$$\zeta = \min_{e \in E} \zeta_e(p_-).$$

Now we have the components to calculate the Price of Anarchy bound for polynomial cost functions.

Theorem 6.6.2. *Given an instance of the traffic light game with polynomial edge-cost functions and waiting time functions of degree d, the Price of Anarchy is bounded by*

$$\left(1 - \Big(\frac{d}{(d + 1)}\Big)\Big(\frac{1 + \zeta}{(d + 1)(1 + \zeta t_-)}\Big)^{1/d}\right)^{-1}.$$

Proof. We substitute η^* into equation 6.6 to find β:

$$\beta(\eta, \zeta) = \sup_{p_e} \Big(1 - \frac{(1 + \zeta) - \zeta(1 + \zeta)\bar{w}_e(p_e)}{(1 + \zeta)(d + 1)(1 + \zeta t_-)}\Big)\Big(\frac{(1 + \zeta)}{(d + 1)(1 + \zeta t_-)}\Big)^{1/d}$$

$$\beta(\zeta) = \Big(1 - \frac{(1 + \zeta) + \zeta(1 + \zeta)t_-}{(1 + \zeta)(d + 1)(1 + \zeta t_-)}\Big)\Big(\frac{(1 + \zeta)}{(d + 1)(1 + \zeta t_-)}\Big)^{1/d}$$

$$\beta(\zeta) = \Big(1 - \frac{1+\zeta t_-}{(d+1)(1+\zeta t_-)}\Big)\Big(\frac{(1+\zeta)}{(d+1)(1+\zeta t_-)}\Big)^{1/d}$$

$$\beta(\zeta) = \Big(1 - \frac{1}{(d+1)}\Big)\Big(\frac{(1+\zeta)}{(d+1)(1+\zeta t_-)}\Big)^{1/d}$$

$$\beta(\zeta) = \big(\frac{d}{(d+1)}\big)\Big(\frac{(1+\zeta)}{(d+1)(1+\zeta t_-)}\Big)^{1/d}$$

Combine this with

$$PoA = (1 - \beta(\zeta))^{-1},$$

and the result follows:

$$PoA = \left(1 - \big(\frac{d}{(d+1)}\big)\Big(\frac{1+\zeta}{(d+1)(1+\zeta t_-)}\Big)^{1/d}\right)^{-1}.$$

□

Thus, to find the Price of Anarchy for any $\boldsymbol{p}$, we can vary p_- and use Theorem 6.6.2. We can also use this result to give us a closed form condition on waiting times to guarantee a good Price of Anarchy.

In summary, we have classified the inefficiency of equilibria in traffic light games in terms of Price of Anarchy and Braess' paradox. In these games, we assumed that each of the traffic light cycles was fixed at some value p. Let us continue by analysing the case where traffic lights are responsive. In fact, we let the traffic lights be intelligent agents selecting light cycles in the traffic light game.

6.7 Adaptive Traffic Lights

Now that we have shown the possibility of handling Braess' paradox and the Price of Anarchy by setting traffic light cycles in networks off-line, we address the problem of learning their optimal values, in a decentralised, adaptive, manner. This coincides with intelligent traffic light research, where algorithms are able to directly respond to dynamic routing behaviours. Rather than considering fixed traffic light cycles, we adapt the nonatomic congestion game to allow for traffic lights to be agents who wish to set their cycle in response to the congestion on the edges for which the light sequences effect.

To this end, we consider a variant of the traffic light game where the light cycles are not fixed. Instead, there exists players, V, at each traffic light node whose goal is to choose the values of p_e, $\forall e \in E_v$, such that it minimises the total

costs of edges which terminate at the node, where E_v denotes the set of edges which terminate at node v. The cost function of these node players are $C_v(\boldsymbol{x}, \boldsymbol{p}_v) = \sum_{e \in E_v} c_e(f_e(\boldsymbol{x}), \boldsymbol{p}_e)$. A vector traffic light cycles $\boldsymbol{p}$ is *feasible* if $\forall v \in V$, $\sum_{e \in E_v}(1 - \boldsymbol{p}_e) = 1$ where E_v is the set of edges ending at v.

Definition 6.7.1 (Traffic control game)**.** *A* ***traffic control game*** *is a tuple* $\mathcal{M} = (N, V, E, (S_i)_{i \in N}, P, (c_e)_{e \in E}, (d_i)_{i \in N})$*, where* N *is the set of players,* V *is the set of traffic lights,* E *is the set of edges,* S_i *the strategy set of player* $i \in N$*,* P *the set of feasible traffic light cycles,* c_e *the edge-cost functions, and* d_i *the demand of population* $i \in N$*.*

The social cost of the game will be the sum of all journey times, i.e. we only consider the costs of players belonging to N.

Theorem 6.7.2. *Every traffic control game is an exact potential game.*

Proof. Define the potential function of the game as

$$\Phi(\boldsymbol{x}, \boldsymbol{p}) = \sum_{e \in E} \int_0^{f_e(x)} c_e(z, \boldsymbol{p}_e) dz.$$

For any player $i \in N$ and $s_i, s_i' \in S_i$,

$$\begin{aligned}
\Phi(\boldsymbol{x}', \boldsymbol{p}) - \Phi(\boldsymbol{x}, \boldsymbol{p}) &= \sum_{e \in E} \int_0^{f_e(\boldsymbol{x}')} c_e(\boldsymbol{z}, \boldsymbol{p}_v) d\boldsymbol{z} - \sum_{e \in E} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}, \boldsymbol{p}) d\boldsymbol{z} \\
&= \sum_{e \in E} \Big[\int_0^{f_e(\boldsymbol{x}')} c_e(\boldsymbol{z}, \boldsymbol{p}_v) d\boldsymbol{z} - \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}, \boldsymbol{p}) d\boldsymbol{z} \Big] \\
&= \sum_{e \in s_i'} \int_0^{f_e(\boldsymbol{x}')} c_e(\boldsymbol{z}, \boldsymbol{p}_v) d\boldsymbol{z} - \sum_{e \in s_i} \int_0^{f_e(\boldsymbol{x})} c_e(\boldsymbol{z}, \boldsymbol{p}) d\boldsymbol{z} \\
&= C_i(\boldsymbol{x}', \boldsymbol{p}) - C_i(\boldsymbol{x}, \boldsymbol{p})
\end{aligned}$$

Thus, the potential function is an exact potential for $i \in N$, following from previous results in the literature (see e.g., Monderer & Shapley (1996)). For any player $v \in V$,

$$\begin{aligned}\Phi(\boldsymbol{x}, \boldsymbol{p}'_v, \boldsymbol{p}_{-v}) - \Phi(\boldsymbol{x}, \boldsymbol{p}_v, \boldsymbol{p}_{-v}) &= \sum_{e \in E} \int_0^{f_e(\boldsymbol{x})} c_e(z, (\boldsymbol{p}'_v, \boldsymbol{p}_{-v})_e) dz \\ &\qquad - \sum_{e \in E} \int_0^{f_e(\boldsymbol{x})} c_e(z, (\boldsymbol{p}_v, \boldsymbol{p}_{-v})_e) dz \\ &= \sum_{e \in E_v} \Big[\int_0^{f_e(\boldsymbol{x})} c_e(z, (\boldsymbol{p}'_v)_e) dz - \int_0^{f_e(\boldsymbol{x})} c_e(z, (\boldsymbol{p}_v)_e) dz \Big] \\ &= C_v(\boldsymbol{x}, \boldsymbol{p}'_v) - C_v(\boldsymbol{x}, \boldsymbol{p}_v)\end{aligned}$$

Hence, the game has an exact potential function as required. □

Since the traffic control problem is now written as a potential game, there exists a pure Nash equilibrium which can be reached through myopic best response, where a best response strategy is defined as $s_i^{BR} := \arg\min_{s_i \in S_i} C_i(s, \boldsymbol{x}, \boldsymbol{p})$ for any player $i \in N$ and $p_v^{BR} := \arg\min_{p_v \in [p_-, p_+]} C_i(\boldsymbol{x}, p_v, \boldsymbol{p})$ for players $v \in V$, where $(\boldsymbol{x}, \boldsymbol{p})$ are strategies from all other players.

The equilibrium of the traffic control game is essentially unique, in that each equilibrium has the same social cost. The traffic light agents aim to minimise the social cost by using their local information and hence always find the user equilibrium with minimum social cost.

For practical purposes, we can alter the cost functions of the traffic light agents to be $C_v(\boldsymbol{x}, \boldsymbol{p}_v) = \sum_{e \in E_v} w_e(f_e(\boldsymbol{x}), \boldsymbol{p}_e)$, whilst still maintaining a potential game. The proof of this follows trivially from the proof of Theorem 6.7.2. In a real system, these more localised cost functions are more appropriate since they can use cameras to detect waiting times within range of the junction, whereas calculating journey times across the length of a road requires higher level data collection. We have shown that best response dynamics converges to optimal traffic light cycles, an improvement on previous literature which finds this a difficult problem usually solved with estimation algorithms Smith & Van Vuren (1993); Smith (1981) or reinforcement learning Kuyer *et al.* (2008); Wiering (2000). Finding optimal traffic light cycles with dynamic routing behaviours is equivalent to finding pure Nash equilibria in an exact potential game. In particular, improvement paths Monderer & Shapley (1996) converge at the equilibrium for all potential games. In practice, approximate equilibria can usually be reached in polynomial time. For increasing cost functions, there is a unique globally stable equilibrium Sandholm (2001).

However, remember that we made the simplifying assumptions that traffic light congestion at a junction is independent of the traffic lights at other junctions. Thus, in real-world application, the problem will be more complex due to the impact from coordination required between neighbouring junctions.

Since there exists an essentially unique equilibrium for all potential games, any equilibrium reached through a learning algorithm will have the same social cost. Thus, we expect that reinforcement learning traffic light agents will also converge in this game. The traffic control game provides a natural and realistic reward function for reinforcement learning traffic lights that require only local information to arrive at an equilibrium - cumulative waiting times of cars at junctions.

Theorem 6.7.3. *Any traffic light learning algorithm based on local improvements minimising cost C_v will converge to an essentially unique equilibrium.*

The proof of this follows directly from potential games literature, since all local minimisers of potential are equilibria (Sandholm, 2001, Theorem 4.4). Similarly, a strategy distribution is an ICUE if, and only if, it minimises the potential function Acemoglu *et al.* (2018). Since the potential function expresses all player's cost functions (including C_v), the convergence of a learning algorithm necessarily finds an essentially unique ICUE.

A further direct consequence of this is that multi-agent reinforcement learning algorithms will find the same optimal traffic light cycles as single agent reinforcement learning algorithms. This implies that the additional costs of cooperation between agents is not necessary to find the optimal traffic light sequences in nonatomic congestion games. In fact, the speed of individual learning will make it more preferable, since many survey papers for MARL highlight problems with scalability due to the exponential growth of state-action space Hernandez-Leal *et al.* (2018); Nowé *et al.* (2012). For very large or continuous state-action spaces, approximate solution methods aim to estimate the equilibrium strategies. Moreover, learning the opponents' behaviour might not be successful if the opponents do too much exploration Albrecht & Stone (2018); Hernandez-Leal *et al.* (2017); Panait & Luke (2005).

We include a simple simulation to show that the Theorem 6.7.3 holds in simulated nonatomic congestion games in Appendix B.

6.8 Biased Adaptive Traffic Lights

The focus of this chapter now directs to looking at the effects of reinforcement learning traffic lights in more detail. This includes outlining the implementation

of Q-learning traffic lights in SUMO and comparing simulated data of static and adaptive traffic lights to predict and mitigate undesirable consequences. To achieve this, we analyse an instance of a traffic control game on the Wheatstone network in further detail. Later, we will simulate this example in SUMO using adaptive traffic lights and consider the effects of intelligent traffic lights on routing and journey times.

To motivate the use of the Wheatstone network in the SUMO simulations, consider what happens when we include adaptive traffic lights to the underlying nonatomic congestion game. (The results predict there could be a problem with unfair or biased routing.)

Figure 6.7 shows the cost functions of the Wheatstone network, where a square node represents the presence of a traffic light. We add two traffic lights to the Braess example where p_1 is the parameter for traffic light 1 (TL1) and p_2 corresponds to the traffic light cycle for traffic light 2 (TL2).

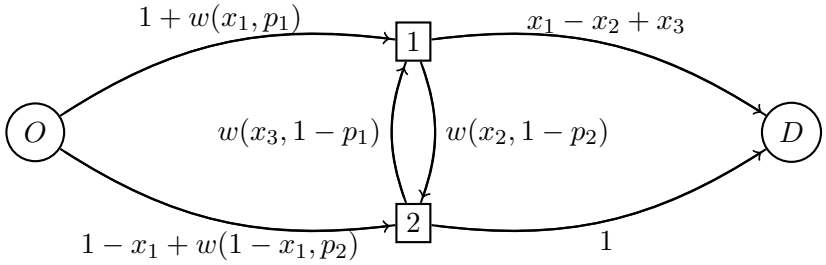

Figure 6.7: The Wheatstone network with two traffic lights and cost functions where the demand is 1.

We consider the more realistic case of the traffic light game where waiting time functions are finite. Suppose that the waiting times are of the form $w(x, p) = x(e^p - 1)$. We restrict $p \in [0.15, 0.85]$ to enable realistic routing patterns for finite waiting costs.

A simple atomic simulation was used to find the equilibrium flow and traffic light parameters for this setup. The unique equilibrium of this instance of the traffic control problem is $x_1 = 0.113$, $x_2 = 0$, $x_3 = 0.763$, $p_1 = 0.85$, and $p_2 = 0.15$. It has a social cost of 2.03. In our simulations, simple myopic best response dynamics, and a random initialisation of all values, converge to the equilibrium solution in less than 150 iterations with 200 players. We use this method to find the UE to confirm that the adaptive traffic lights converge to the correct soluation.

Now, we use reinforcement learning algorithms to find the optimal light cycles. Here, we test the independent implementation of the Q-learning algorithm on

each traffic light, where the actions are discretised values of p and the state is the UE edge flow of players' strategies. Specifically, TL1 and TL2 have a local state-space representation of (x_1, x_3) and $(1-x_1, x_2)$ respectively, representing congestion levels of the roads the end at each traffic light. The action space for both agents is $\{0.15, 0.5, 0.85\}$. The reward function is the negative sum of edge costs of the edges which face the traffic light. Further algorithm parameters are shown in Table 6.1. The results of these simulations are shown in Figure 6.8.

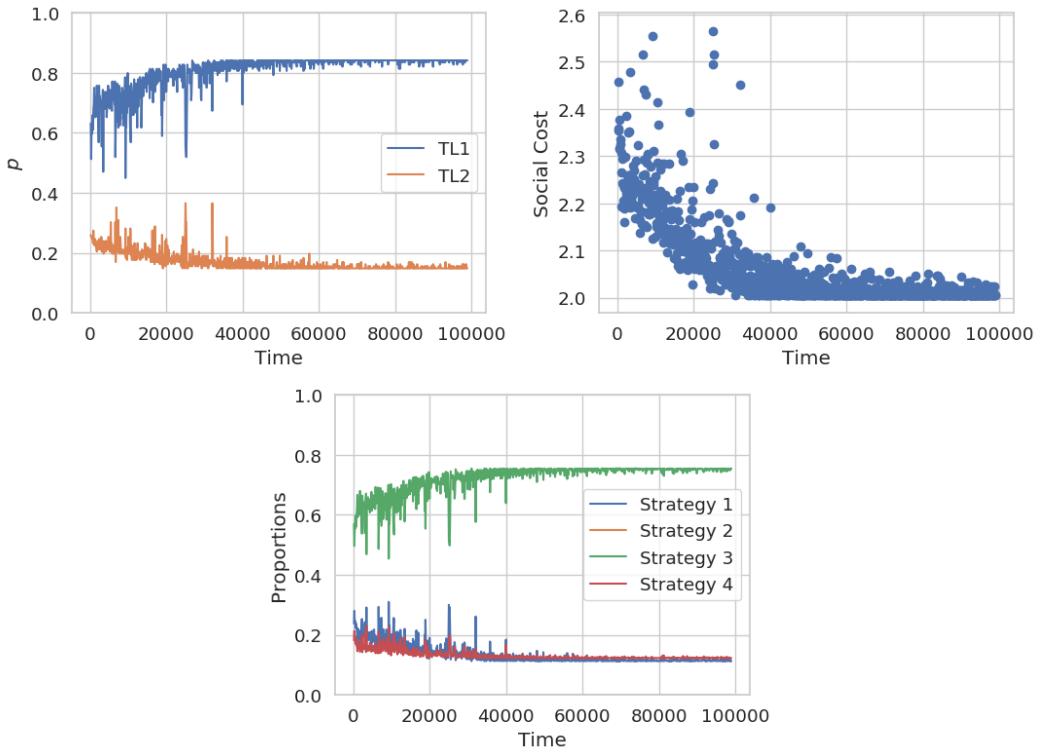

Figure 6.8: Here, we show the learning for Q-learning when the initial flow is at UE and BR dynamics determine the flow at each timestep. To the left, we see the p and q values learned from Q-learning. The convergence of social cost is snown on the right. Below is the best response dynamics of drivers, where strategy 1 corresponds to the journey through TL1 only, strategy 2 crosses TL1 then TL2, strategy 3 crosses TL2 then TL1, and strategy 4 passes TL2 only.

We see from Figure 6.8 that the Q-learning traffic lights converge to the same solution as the myopic best response equilibrium. Although this solution minimises social costs, having extreme values of p_1 and p_2 could cause disruptions in real-world traffic. The cost of using the middle edges will become large and could create inequality.

Table 6.1: Q-Learning Parameters

Parameter	**Value**
Learning rate	0.1
Exploration rate	$\max\{0.9 * 0.99^t, 0.005\}$
Discount factor	0.99

Biases need to be avoided in real-world applications to reduce unfairness and excessive waiting times. Nevertheless, RL algorithms will likely prefer a biased system due to its reduced average cost. Thus, we test the same Wheatstone network setup on a more complex and realistic simulator, SUMO, to quantify the potential issues of adaptive traffic lights optimising average journey times.

6.8.1 Traffic Lights in SUMO

Here, we use SUMO simulations to analyse a more realistic traffic system and try to quantify its inefficiencies. First, we outline the details of the simulation and then we analyse the results for bias and unfairness.

The simulations are produced in SUMO, which uses an iterative algorithm to determine a dynamic user equilibrium as its routing method Gawron (1998a). In general, dynamic user equilibrium approximates the user equilibrium. The route choice algorithm, used as default by SUMO, calculates the probabilities of choosing routes based on their expected journey times from previous simulation steps Gawron (1998b). Thus, the congestion game model is compatible with this software. Other traffic simulators also use congestion game based routing e.g., Vissim, Aimsun.

Implementation of MARL traffic lights in the literature is varied in terms of state space, action space and reward functions. In our simulation, we extend the traffic agents from the nonatomic congestion game to an RL agent implementable in SUMO and achievable in a real-world system. We choose to use independent Q-learning agents to represent adaptive agents due to their ease of implementation and fast convergence.

We define a traffic light cycle by a sequence of phases where green lights are showing for certain lanes. In general, there are many ways to such choose phases. Figure 6.9 shows our selected phases for the Wheatstone network. We use these for their simplicity and removal of intersecting flows. In our simulations, we compare a static cycle with an adaptive cycle controlled by Q-learning.

Firstly, let us describe the static cycles. A traffic light with a static cycle repeats the same sequence of phase lengths for fixed times. In our simulations, each

green phase lasts 32 seconds and is followed by an amber phase of 3 seconds to allow for adequate reaction and stopping time.

Now, we describe the implementation of the adaptive traffic lights. Contrary to the static cycles, an adaptive traffic light has varied phase lengths. Their aim is to use local congestion information to optimise traffic flow. To represent adaptive traffic lights, we use a Q-learning agent, where the state space corresponds to the traffic at the junction and the actions allow for varied phase lengths.

The state should be able to encode the environment to a sufficient level of detail for analysis, yet, it must be achievable through current technology such as cameras or signals for the model to be implementable in the real-world. Our chosen state space for each set of traffic lights is encoded as a vector of multiple variables. These are the current phase of the cycle, the proportion of elapsed time out of the maximum green time of the current phase, and each lane's density at the junction.

The actions are predefined by the possible configurations of green lights on lanes to minimising conflicting traffic flow. In this instance, a traffic light has two actions as there are two possible directions of flow (see Figure 6.9). A traffic light chooses which traffic light phase will be used for the next δ timesteps. Therefore, the action space of each set of traffic lights is the possible green light phases. Here, we choose $\delta = 5$ timesteps in between each decision. Similarly to the static lights, between green or red phases there is an amber light phase of 3 seconds to allow for safe stopping.

The reward functions should use locally collected measures in order to optimise flow. We choose the reward function compatible with the potential function in order to preserve the game's unique equilibrium (from the traffic control game, Theorem 6.7.2). Thus, the reward function for junction v is $r_v = -\sum_{e \in E_v} w_e(f_e(\boldsymbol{x}), \boldsymbol{p}_e)$.

We choose an exploration rate that is time-dependent, to allow for more exploration at the beginning of the simulation to reduce any early bias. The learning rate and discount factor are typical values found in the literature. Table 6.2 is a summary of these parameters.

Table 6.2: Q-Learning Parameters

Parameter	**Symbol**	**Value**
Learning rate	α	0.1
Exploration rate	ϵ	$0.9 * 0.99^t$
Discount factor	γ	0.99
Time	T	100000

Figure 6.9 shows a screenshot of the Wheatstone network implemented in SUMO. To reduce the cost of the middle edge, we change the speed limit of the edge from a low value (0.1ms^{-1}) to a high value (500ms^{-1}). This change should induce Braess' paradox. For a slight variation in the other edge costs, two of the edges have a single lane and the other two have two lanes each. The multiple lane edges also have a lower speed limit (10ms^{-1} compared to 20ms^{-1}). Thus, the edges with multiple lanes should be less sensitive to congestion than single lane edges.

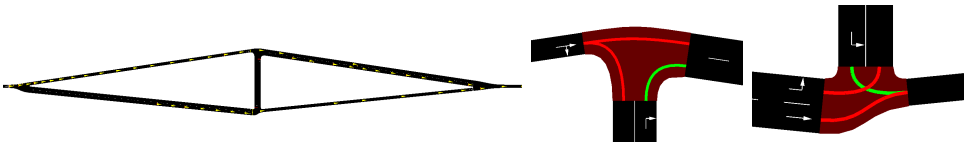

Figure 6.9: The Wheatstone network was set up with variable speed limit across the middle edge so that it either had a low speed limit (high cost) or high speed limit (low cost). The network (left) is taken from the simulation with a high-cost middle edge so that drivers choose to route around it. Each traffic light has two possible green phases. Here is one of the phases for traffic lights TL1 (middle) and TL2 (right).

The simulation has a discrete number of cars rather than the nonatomic flow of the congestion game, i.e., the set of players is defined $N = \{1, 2, ..., n\}$ where $n \in \mathbb{N}$. In this example, all players in N move between the same origin and destination. All simulations have the same number of cars overall, with cars randomly generated according to predefined distributions.

The environment transitions are determined by the SUMO simulation, which mimics realistic driving behaviours. Further information about how SUMO simulations work can be found in this paper Vinet & Zhedanov (2011).

6.8.2 Simulation Results

We will now compare the results of the simulations of static and adaptive traffic lights when changing the cost of an edge from high cost (before) to low cost (after).

Figure 6.10 shows the distribution of cars' travel times in the simulations for either static or adaptive traffic lights. It indicates that static cycles have a much wider deviation from the mean after the speed limit is increased. However, for the Q-learning traffic light agents, there is also a significant shift in the mean and deviation after the change. This means that the intelligent traffic lights are still vulnerable to Braess' paradox. There is a distinct change in the optimal policy of the Q-learning agents when the speed limit is increased, which is what enables Braess'

paradox. Note that because the routing algorithm in SUMO is probabilistic, we end up with a few cars choosing a route with very high travel times when Q-learning agents control the traffic lights.

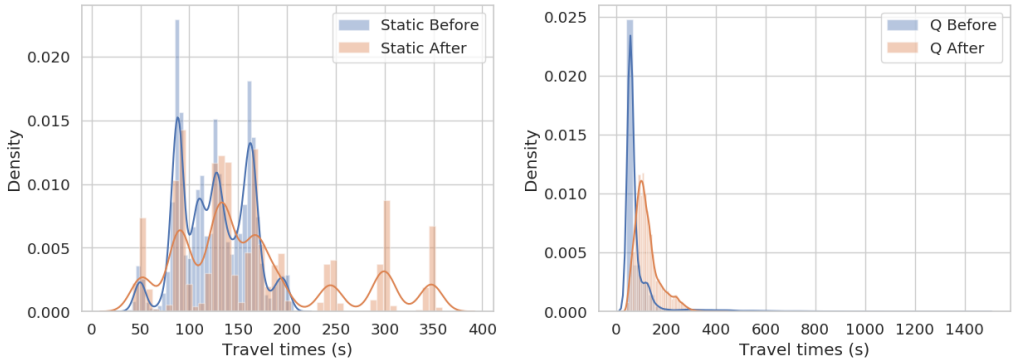

Figure 6.10: These figures are histograms of the travel times of all travel times of the Wheatstone simulation (ignoring the first 100,000 cars to allow for learning) averaged over 8 runs.

In Figure 6.11, we see that same data plotted as Figure 6.10; a bar chart allows an easier comparison between the static and adaptive light cycles.

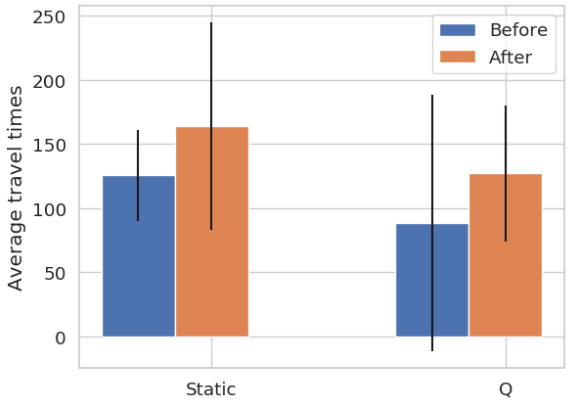

Figure 6.11: Average trip times with error bars of 95% confidence intervals for static and Q-learning traffic lights on the Wheatstone network (before and after the speed limit change).

Figure 6.12 shows the changes in the traffic light parameter p chosen for each traffic light by Q-learning agents for the simulation, before and after the speed limit change. In the before case, where the middle edge has a low speed limit, the Q-learning traffic light agent is able to learn the equilibrium solution quickly and,

therefore, reduce average travel times. After the speed limit increases, this road becomes more desirable to selfish drivers. It is then much more difficult for the Q-learning agent to find the equilibrium solution. Note that the static cycle traffic lights have a constant parameter of $p = 0.5$.

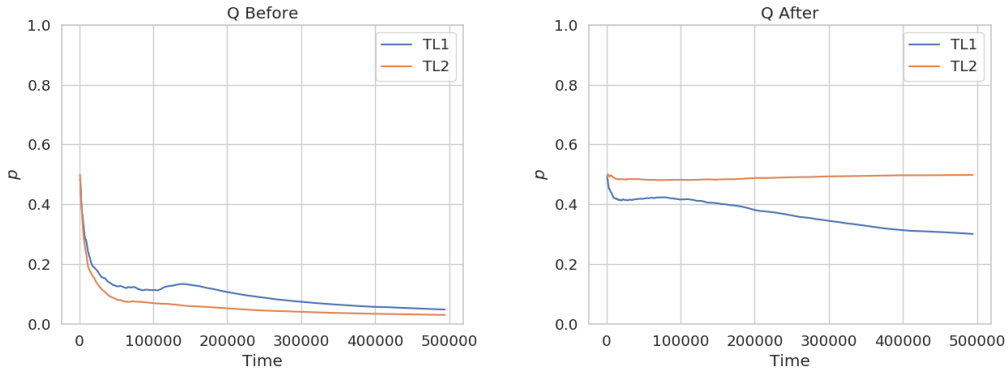

Figure 6.12: Convergence of p for Q-learning traffic lights on the Wheatstone network.

In the "before" case, the adaptive traffic light learns to choose values of the traffic light parameters close to 0. As such, it causes the inequality seen in Figure 6.10 where some drivers have very long journey times. To quantify these effects, we will now analyse the data for whether the adaptive traffic lights are unfair to drivers.

6.8.3 Fairness of Adaptive Traffic Lights

Since adaptive traffic lights only optimise average measures of waiting times, the variance of waiting times can be large. Therefore, there can be large inequalities between journey times of drivers based on their route choice. This is highly undesirable for drivers, and these effects should be further understood in order to mitigate them.

The near-zero values for p, found from adaptive traffic light agents, make the system unfair, since this massively increases journey times of certain edges. The Wheatstone simulation is one example of a traffic network where the values for traffic light parameters p are close to 0 or 1. Ergo, we continue to study the results of this simulation in the fairness context.

Let us first define some notions of fairness in terms of journey times, i.e. costs, to drivers for two-terminal atomic congestion games.

Definition 6.8.1 (k-Satisfied)**.** *A player i is* ***k-satisfied*** *if*

$$C_i(\boldsymbol{x}, \boldsymbol{p}) < \frac{k}{n} \cdot \sum_{j \in N} C_j(\boldsymbol{x}, \boldsymbol{p}).$$

Definition 6.8.2 (k-Envy-Free)**.** *A player i is* ***k-envy-free*** *if*

$$C_i(\boldsymbol{x}, \boldsymbol{p}) < k \cdot \min_{j \in N} C_j(\boldsymbol{x}, \boldsymbol{p}).$$

Satisfaction occurs when players compare their journey times with that of the average journey of players. Envy-free comes from when players compare their journey times with that of the shortest journey time of any player. Players who are satisfied have a more cooperative and egalitarian way of deciding whether their journey times were acceptable, when compared to those who want to be envy-free.

Some variation in journey times between players can be expected for those that have varying departure times. However, if drivers wish to travel in convoy, they prefer to have a similar journey time to those who have similar departure times.

Definition 6.8.3 (k-Satisfied in Convoy)**.** *A player i is* ***k-satisfied in convoy*** *if others, who leave within a time window of T seconds of their departure time t_0^i, have a similar journey time to them:*

$$C_i(\boldsymbol{x}, \boldsymbol{p}) < \frac{k}{\sum_{j \in N} \mathbf{1}_{\{t_0^i - T < t_0^j < t_0^i + T\}}} \sum_{j \in N} \mathbf{1}_{\{t_0^i - T < t_0^j < t_0^i + T\}} C_j(\boldsymbol{x}, \boldsymbol{p}).$$

These three measures are possible ways that a person might choose to decide whether they had an acceptable journey time. We use them to see what proportion of players in the simulation would approve of their journey and thus are satisfied with the implementation of intelligent traffic lights.

Figure 6.13 compares the satisfaction and envy-freeness of the Wheatstone network simulations for both static and adaptive light cycles. The adaptive traffic light before the speed limit change has the highest number of envy-free players for all values of k, but it also has the worst satisfaction rate. The high number of envy-free player is from the low average travel time, but the satisfaction does not reach 100%, since there are a small number of drivers who have very long journey times. The price of selfish routing in this instance is that in order to have a reduced average travel time, there must be a few players with very high expected journey times. For the static traffic lights, with both speed limits, 100% satisfaction of players is achieved for k> 3. However, the envy-freeness is worse than the adaptive lights for

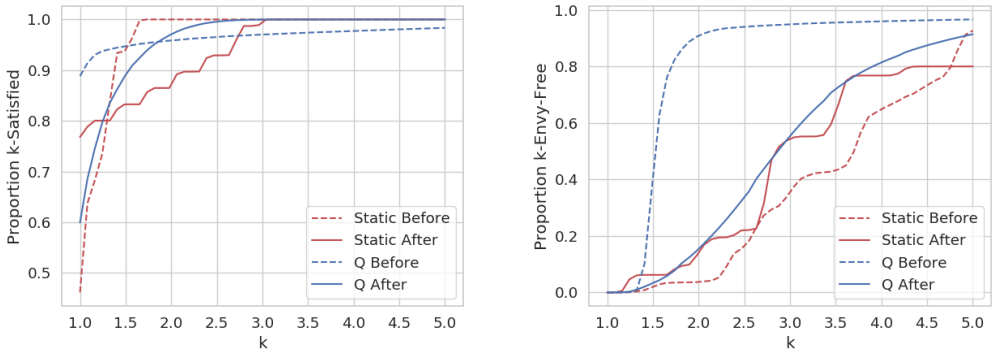

Figure 6.13: The k-satisfication (left) and k-envy-freeness (right) for drivers in the Wheatstone network simulation for varying parameter k. The four types of simulation are static and adaptive light cycles for both before and after the speed limit change.

all values of k. This is because envy-freeness is improved by reducing the average travel times whereas the same is not true for the satisfaction measure.

In Figure 6.14, we see the satisfaction of drivers travelling in convoy, i.e., those who depart within a timeframe of T seconds from each other. It is preferable for people leaving at the same time from the origin to have similar journey times. Otherwise, cars will struggle to travel in convoy.

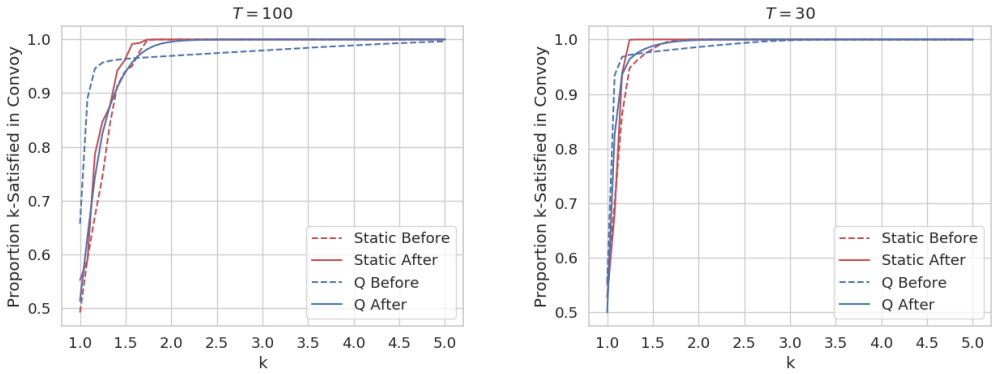

Figure 6.14: k-satisfaction in convoy for $T = 100$ (left) and $T = 30$ (right) for the Wheatstone network simulation.

For $T = 100$, the Q-learning adaptive traffic light before the speed limit is altered is not able to keep journey times within a factor of two, for cars departing in the same 100 second window. The other three simulations do not have this property. Again, we see that the adaptive traffic light struggles to distribute journey times

equitably due to the strong bias of keeping specific roads open to reduce average journey times. If an adaptive traffic light has the potential for cars leaving at the same time to have a journey time longer than three times the average of cars leaving at a similar time, then this has potential for some drivers to be very unhappy with the light sequence.

For the smaller time window of $T = 30$, the problem is reduced and the issues of the adaptive traffic lights unfairly altering journey times is mitigated. However, the parameter T is not controlled by the designer of the traffic lights, but rather represents driver preferences. Since we do not have any experimental results in the literature to choose T, considerations must be made for a range of T values.

It is essential that intelligent lights are useful for reducing the average travel times of cars and that users of the system trust it, so that they can benefit from its implementation. If there are cases where two drivers can leave at a similar time and have journey times differing by more than a factor of two, such as in the Q-learning based example from Figure 6.14, then it is possible that the public would prefer static light cycles.

To address these issues, we could use a different reward function to enforce behaviour that minimises the standard deviation of the travel times as well as the mean. For example, the rewards should punish large waiting times more than it punishes smaller waiting times. In the next section, we test out this theory by selecting reward functions that could improve the skewed distribution of travel times and use simulated data to compare them.

6.8.4 Fairness of Reward Functions

The natural cost function for intelligent traffic lights to converge to a Nash equilibrium in a nonatomic congestion game, as discussed in Section 6.7, was the sum of waiting times at junctions. However, the choice of reward function may affect its fairness and vulnerability to inefficient equilibria.

Suppose that we choose to square each cars waiting time before summation as the reward function. Then there should be an incentive to reduce the excessively long waiting times and improve average journey times. This should reduce the variance in journey times, thus, increasing fairness.

Similarly to squaring the waiting times, choosing the maximum waiting time as the reward function should reduce those cars with very long waiting times at junctions. Again, these rewards could improve the fairness of traffic lights in relation to journey times.

In Lujak *et al.* (2015), the fairness of route guidance is addressed using a

normalised mean path duration cost. This metric is used to encourage envy-free traffic assignment. Thus, we could use a similar expression to find normalised mean waiting times in the hope of encouraging fair traffic light behaviour.

To discover which reward function is the most successful at mitigating unfairness whilst optimising average jounrey times, we tested them using a Q-learning adaptive traffic light in the Wheatstone simulation. Let j be a traffic light agent, and E_j be the edges directed towards the node where j exists. The reward functions chosen to compare against the fairness measures were:

- Wait: The sum of waiting times at junctions. This is the baseline to compare other reward functions against.

$$r_j = \sum_{e \in E_j} w_e(f_e(\boldsymbol{x}), p)$$

- Wait^2: The sum of the squared waiting times at junctions.

$$r_j = \sum_{e \in E_j} (w_e(f_e(\boldsymbol{x}), p))^2$$

- Norm: Inspired by Lujak *et al.* (2015), we test a normalised mean strategy cost, as defined:

$$r_j = \sqrt[n_j]{\prod_{e \in E_j} w_e(f_e(\boldsymbol{x}), p)}$$

  where n_j is the number of cars waiting at the junction, i.e., the number of waiting times in the expression.

- Norm^2: The Norm reward function where the waiting time functions are squared.

$$r_j = \sqrt[n_j]{\prod_{e \in E_j} (w_e(f_e(\boldsymbol{x}), p))^2}$$

- max(Wait): The maximum wait time of any car at the junction.

$$r_j = \max_{e \in E_j} w_e(f_e(\boldsymbol{x}), p)$$

- max(Wait^2): The squared maximum wait time of any car at the junction.

$$r_j = \max_{e \in E_j} (w_e(f_e(\boldsymbol{x}), p))^2$$

There are, of course, many other possible ways of choosing reward functions based on the queue size or speed of vehicles. However, since Theorem 6.7.2 implies that waiting time functions enable convergence to equilibria, we chose to use waiting time based rewards.

Each of the rewards functions was simulated over three different random seeds, with the same simulation setup as Section 6.8.1. The results of these simulations are shown in Figures 6.15, 6.16, 6.17, and 6.18.

The convergence of traffic parameters p for TL1 and TL2 in the Wheatstone simulations are shown Figure 6.15. In general, the selection of p values by the different reward functions look quite similar in the before case and there is more variation seen between reward functions after the speed limit change. Nevertheless, the differences are more easily interpreted using the fairness metrics. In the before case, both traffic lights always choose p close to zero. As such, we expect there to be an unfair distribution of journey times since drivers that attempt to use the road connecting TL1 and TL2 will have a large journey time.

In terms of envy-freeness, Figure 6.16 shows the waiting time based reward functions perform the best in both speed limit versions of the simulation. However, as discussed previously, this measure is highest when the average journey time is minimised. This suggests there could be unfair distributions of players that could be picked up from the satisfaction measures. Furthermore, it shows that these simulated results substantiate the theoretical reasoning for choosing the sum of the waiting times as a reward function. Before the speed limit change, the different reward functions are very similar, except for max(Wait) which is slightly less favourable. After the speed limit change, there is a bigger difference between their performances. The summation of waiting times, Wait, rewards has the best satisfaction for all values of $k \leq 3.5$. The maximum waiting time squared, $\max(\text{Wait}^2)$, rewards give the worst satisfaction levels for $k \leq 3.5$. Thus, the results show that the waiting time function (Wait) is able to minimise average journey times the most successfully of the reward functions tested.

The k-satisfaction for each reward function is shown in Figure 6.17. Before the speed limit change, the different reward functions are very similar as all values of k. The only noticeable difference is that the Norm^2 reward is slightly worse. Note that no reward is able to achieve 100% satisfaction for $k \leq 5$. After the change, there is a bigger difference between the satisfaction of reward functions. The summation of waiting times squared, Wait^2, has the best satisfaction for all values of k. The maximum waiting time rewards give the worst satisfaction levels.

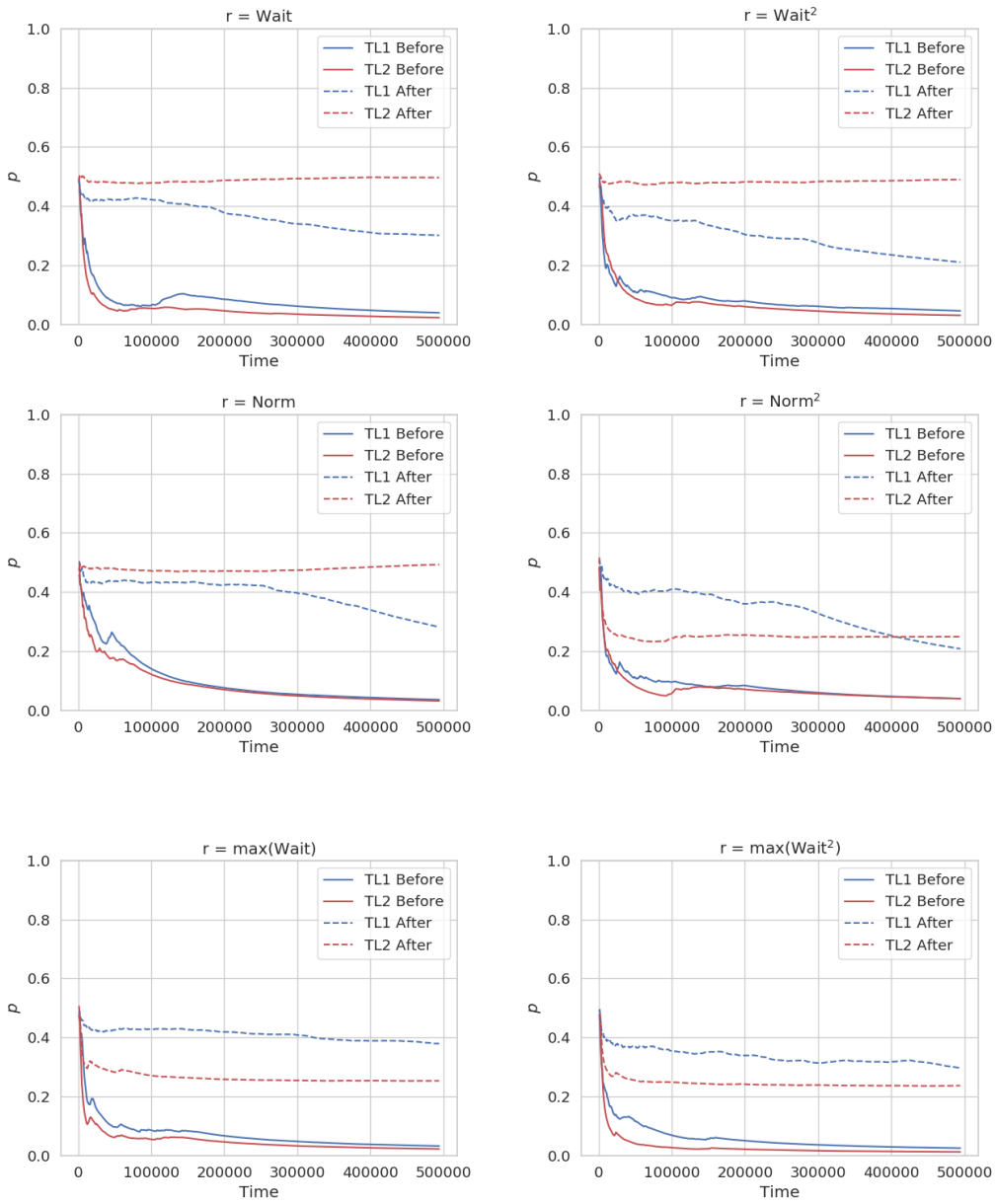

Figure 6.15: Convergence of traffic light parameters p for different reward functions in simulations of the Wheatstone network.

The best reward function to choose to improve the satisfaction of drivers would be the reward functions that sums the squared waiting times of cars. As predicted, this will reduce the number of drivers with exceptionally long waiting times. The norm function from Lujak *et al.* (2015) does not improve upon the sum of waiting times

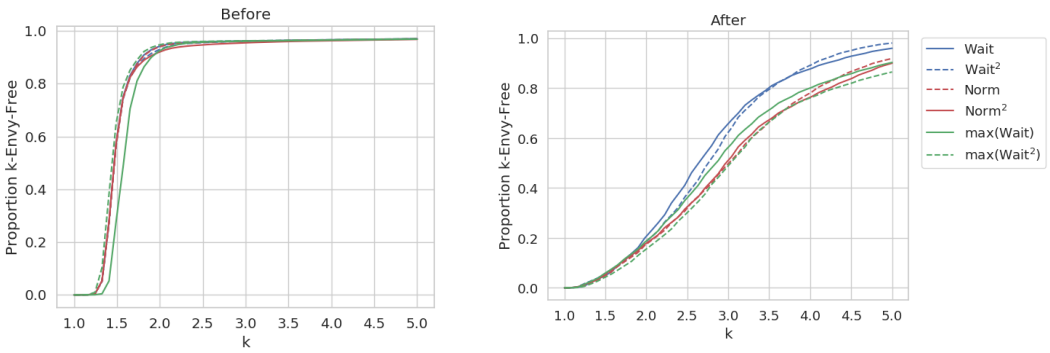

Figure 6.16: The k-envy-freeness for different reward functions in simulations of the Wheatstone network.

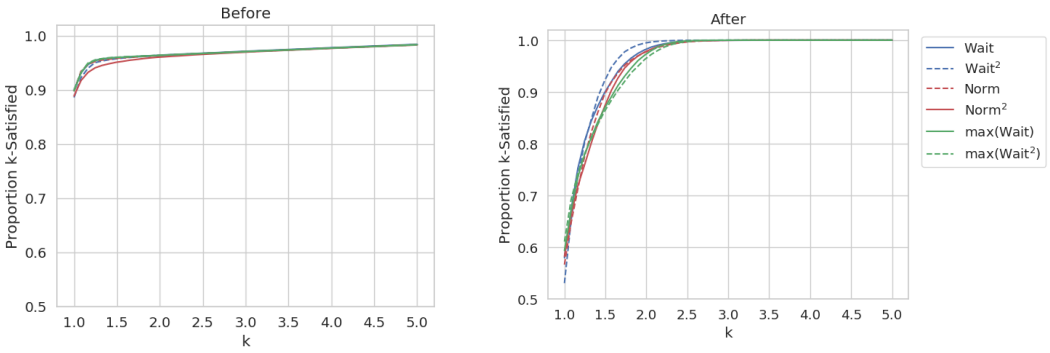

Figure 6.17: The k-satisfaction for different reward functions in simulations of the Wheatstone network.

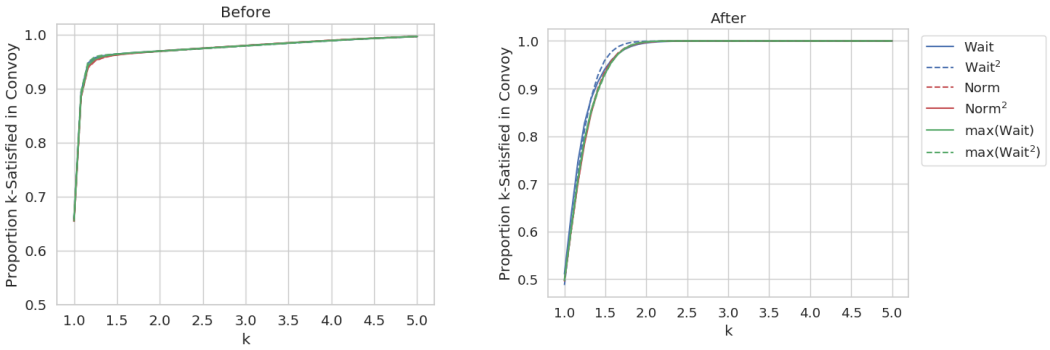

Figure 6.18: The k-satisfaction in Convoy for different reward functions in simulations of the Wheatstone network with $T = 100$.

rewards, which acts as our baseline.

The k-satisfaction in convoy for the different reward functions is shown in Figure 6.18. Here, the various reward functions give the same k-satisfaction in

convoy for all k in the before simulations. After the speed limit is increased, the sum of waiting times squared reward has a slight improvement of k-satisfaction in convoy against all the other reward functions.

From these results, it is clear that the sum of the waiting times is a satisfactory reward function to choose. However, to increase the fairness of the distribution of waiting times, squaring the waiting times before they are summed could lead to a fairer system. Further research on larger scale simulations should be done to confirm this, since we have only considered a simple Wheatstone network.

We have highlighted the need for further theoretical analysis on reinforcement learning traffic lights to avoid excessivley large journey times for drivers. We also proposed the extension of envy-freeness and satisfaction to the domain of travel times as a method of quantifying bias caused by adaptive traffic lights.

6.9 Discussion

In this chapter, we began by showing that it is possible to model traffic lights in a routing game using expected waiting time functions, where the equilibria of such games exist and are essentially unique. Traffic light cycles influence routing choice in relation to the proportion of red time, p, during a full cycle. Moreover, the waiting time functions do not depend on the total cycle time. The waiting time functions behave non-monotonically past an upper bound value of the red cycle time. Therefore, it is important to include a maximum red time length in intelligent traffic light models to reduce any unpredictable behaviour.

By altering the traffic light cycles, people's response can, in turn, significantly shift the social cost of the equilibria. Theoretically, we can use traffic lights to make a network immune to Braess' paradox for any network. However, real-world implementation is not compatible with the upper bounds for the proportion of red time in a traffic light cycle on general networks. Nevertheless, there are cases where immunity can be reached with bounded waiting times. Furthermore, the upper bound on the Price of Anarchy for any traffic light cycle can be written as a function of the traffic light parameter p and the lower bound on waiting times.

Then, we showed that an optimal equilibrium of cost-minimising traffic lights could be found using simple decentralised learning algorithms with a natural and straightforward reward function (Theorem 6.7.3). This result directs research in reinforcement learning traffic lights towards fully decentralised RL algorithms in selfish routing games, to remove the scalability problem that arises with centralised cooperative algorithms on large multi-agent networks. Although the RL algorithms

seen in this chapter are quite simple, the results will also impact the real-world development of AI-based traffic lights. It will improve the speed of learning as well as indicates some of the unfairness issues that could arise with their implementation without additional fairness restrictions.

Using the Wheatstone network simulated in SUMO as an example, we have shown that adaptive traffic lights can create a strong bias towards certain routes that unfairly increase journey times for a small number of users. We suggested three different criteria- envy-freeness, satisfaction, and satisfaction in convoy - to compare the fairness properties of adaptive traffic lights. We tested a selection of rewards functions and compared their fairness metrics to show that altering the reward function of RL traffic lights could improve their fairness. The best candidate for improving fairness whilst maintaining low average travel times was a reward of the negative sum of squared waiting times.

To conclude, we have designed a model of traffic lights compatible with nonatomic congestion games and traffic simulators, addressing essential properties of this system: convergence to equilibria; vulnerability to Braess' paradox; the Price of Anarchy bounds; and, fairness of journey times.

CHAPTER 7

Conclusions

To summarise, in this thesis, we have considered the problem of reducing traffic congestion in several ways. Firstly, we found a belief system to encourage cooperation in social dilemmas whilst maintaining the safety property. We then analysed the impact of multiple route controllers aiming to reduce travel times of subpopulations through information design. Moreover, we found a class of networks that allow for efficient route information distribution. Finally, we designed a game whose cost functions represented traffic light phases and explored how traffic light cycles affect the equilibria of routing choices. Additionally, we showed that intelligent traffic lights admit undesirable properties in some networks.

One possible extension of the work would be to expand upon the ARCTIC algorithm (Chapter 3) for application to general multiplayer games. The main challenge in more complex games is to classify an action, or sequences of actions, as either cooperative or not. There may exist multiple types of cooperation, increasing the challenge of coordination and safe play in such games. Some of the example games used in the literature as more complex social dilemmas are Harvest and Cleanup. In these games, cooperation is required between groups of agents to achieve joint tasks with an additional temporal aspect that the cooperation efforts are rewarded in the long-term rather than short-term. A natural extension would be to expand the theory to these domains[1]. There is also further work needed to address the details of how the policy-conditioned beliefs should be defined for more than two players present. The simulated results indicate that ARCTIC has the desired properties of social dilemma strategies - nice, forgiving, provocable and clear. As such, applying ARCTIC to RL algorithms in complex semi-cooperative environments would be a valuable contribution to the literature.

One of the most unintuitive results was Theorem 6.7.3 in Chapter 6, which

[1]The application of ARCTIC to Harvest and Cleanup games was attempted but was not completed in time to be included in this thesis.

states that all learning algorithms setting traffic light cycles in a nonatomic congestion game will converge to an equilibrium with the same social cost. This suggests that the complex coordination mechanisms studied in the MARL traffic light literature are superfluous, if the traffic can be modelled through a nonatomic congestion game. Although real-world traffic is unlikely to abide by the assumptions of a nonatomic congestion game, it certainly has similar properties to observed routing behaviours. Many of the traffic simulators- SUMO[2], VISUM and VISSIM[3], Aimsun[4], MATSim[5] - use a congestion game in their route planning algorithms and a variant of user equilibrium (dynamic or stochastic).

One of the limitations of the fairness results in Chapter 6, could be the choice of simulation software used to produce the simulated traffic data. Nevertheless, as discussed, all major traffic simulation software use congestion game style routing with a dynamic user equilibrium. As such, the routing algorithms should create similar simulated routes, and this part of the simulator is the most important for our model. Additionally, research papers tend to use only one type of software to produce the simulated results, with the most popular choice being SUMO e.g. Laszka *et al.* (2016); Lopez *et al.* (2018); Mousavi *et al.* (2017); Pol & Oliehoek (2016). The confirmation of similar results using different simulation software is an area for future research.

It could also be argued that congestion games are too abstract to model real-world traffic and that atomic congestion games are more appropriate than nonatomic. Some of the underlying assumptions in a congestion game are that restrict its applicability to real world traffic networks are: (i) all players (vehicles) are of homogeneous size, weight, etc. (ii) the network is at an equilibrium state, (iii) origins and destinations of players are fixed, (iv) all players have the same travel costs, (v) players have identical preferences, (vi) players have full information about the network and congestion levels. Real-world traffic is a complex system that indeed

[2] https://sumo.dlr.de/docs/Demand/Dynamic_User_Assignment.html "The problem of determining suitable routes that take into account travel times in a traffic-loaded network is called user assignment." "The tool duaIterate.py can be used to compute the (approximate) dynamic user equilibrium."

[3] https://www.ptvgroup.com/en/contact-support/add-in-marketplace/traffic-realtime-equilibrium/ "The demand is propagated on the network accordingly with path choices previously calculated either with Dynamic User equilibrium procedure or with any other Visum assignment."

[4] https://www.aimsun.com/aimsun-next/ "Dynamic user equilibrium (DUE) techniques and stochastic/discrete route choice models are both available in Aimsun Next in combination with either mesoscopic or microscopic modeling."

[5] https://www.transitwiki.org/TransitWiki/index.php/MATSim "In the MATSim software, a co-evolutionary algorithm, as opposed to an evolutionary algorithm, is used to obtain equilibrium. This process leads to a stochastic user equilibrium."

expands beyond the scope of a congestion game, since the underlying assumptions do not hold true. However, it is an aphorism (attributed to George Box) that "all models are wrong, but some are useful". The benefits of using simplistic congestion games to model traffic are apparent through their prevalence in simulation software. Additionally, boundedly rational players are more likely to produce realistic routing traffic behaviours, such as the partial information games we considered in Chapters 4 and 5. Furthermore, we chose to focus on nonatomic congestion games over atomic games since their properties allow for easier mathematical analysis.

We also note that standard congestion games do not include time as a variant, which could be limiting in real-world application. However, the results from congestion games are useful for time-varying applications when the traffic reaches a steady-state. Moreover, the SUMO simulated results in Chapter 6 align with the nonatomic congestion game theory since it uses a dynamic user equilibrium that approximates the user equilibrium.

Despite the nonatomic congestion game being far from a perfect model of real-world traffic, there are many other applications of congestion games where the underlying assumptions are more realistic to the systems they represent. For instance, wireless communication networks Liu & Wu (2008), wireless network Quality of Service (QoS) Southwell *et al.* (2013), peer-to-peer computing (P2P) Suri *et al.* (2004), vehicle communication networks Yan *et al.* (2018), vehicular ad-hoc networks (VANETs) Chen *et al.* (2014), virtual drug screening Nikitina *et al.* (2018), and electrical power grids Ibars *et al.* (2010). All of the results posed throughout Chapters 4, 5, and 6 that are based on a congestion game model are also useful for these other applications. By modelling the networks using simplifying assumptions, the results generalise to applications outside of traffic networks. By relaxing the assumptions (i) to (vi), the work would be specifically tailored to traffic networks and no longer useful for the broad applications of congestion games.

One of the major research areas in RL is that of autonomous vehicles, including congestion-aware route planning Rossi *et al.* (2018). Survey research into people's beliefs about the ethics of autonomous vehicles, found that although people were in favour of a utilitarian approach of saving more lives over less, they also said that they would not purchase a utilitarian car themselves due to the risk of self-sacrifice Bonnefon *et al.* (2016). This phenomenon was coined "the social dilemma of autonomous vehicles". Perhaps people would have a similar perspective of socially optimal routing - desiring a utilitarian system that is beneficial for everyone, yet irrationally choosing the opposite. In which case, designing a routing system that drivers have no incentive to defect from, by choosing their own routes, would

be an important extension of the work in Chapter 4.

The research presented in this thesis impacts real-world traffic, particularly for reducing congestion using navigation applications and reinforcement learning traffic lights. Further lines of research motivated by these results include: (i) developing more understanding of fairness in intelligent traffic lights, (ii) improving information distribution efficient for network routing control where some vehicles do not use route planners, and (iii) classifying cooperative behaviours of path planning to encourage coordination of socially beneficial routing.

The key contributions of this thesis are a method for achieving safe cooperation in social dilemmas as well as mitigating the effects of selfish routing in variants of nonatomic congestion games. These variants cover networks where there exist multiple social planners, games with players that have heterogeneous network information, and those with delays at nodes. The results consider avoiding Braess' paradox, bounding the worst-case equilibrium using Price of Anarchy, and cooperating or coordination between players in a social dilemma or controlling delay functions at network junctions. These results are purely theoretical but are motivated by the behaviour of traffic networks and the impact intelligent transport systems have on routing. Each result is applicable to the wide-reaching applications of congestion games or social dilemmas. For instance, Chapter 5 could impact the design of peer-to-peer computing networks to improve information efficiency. The traffic light formulation could generalise to other areas such as VANETs to improve information flow. The network control game in Chapter 4 could be used improve the efficiency of energy grid usage. Additionally, the algorithm achieving safe cooperation in social dilemmas from Chapter 3 could influence automated negotiation strategies.

APPENDIX A

Reinforcement Learning Traffic Lights in SUMO

To show that the assumptions made in the model are reasonable, we show that it is compatible with SUMO[1], an open source microscopic traffic simulation software, popular in the transportation literature. A simulation is built using network information and route files and run using simulated data to mimic simple traffic movement. To verify the our simplifying assumptions over the cost functions hold in traffic simulations, we have set up a SUMO simulation to estimate the cost function of waiting at a traffic light.

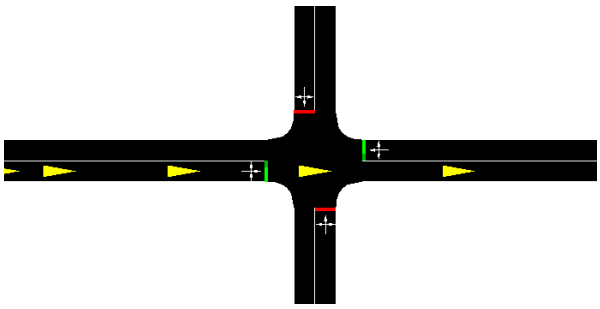

Figure A.1: Section of a SUMO simulation.

The simple network for the simulation tests is a crossroad junction with cars moving in only one direction across two edges. The left edge is 500m long and the right edge is 50m long. The cost of the journey is taken as the total travel time and the congestion, $x := f_e(\boldsymbol{x})$, on an edge is the average number of cars

[1]Note, the choice of SUMO as the simulation software is unlikely to effect the outcome of the results since other major simulation software (VISSIM, Aimsun, MATSim, SpaceWorks) use a similar congestion game based routing algorithm to estimate (dynamic) user equilibrium routing.

that are present during the time a car is on the edge. Each simulation ran for 20,000 timesteps with the same number of cars and distribution of departure times. The cars are homogenous (e.g. size and acceleration) with default SUMO vehicle parameters.

Firstly, we had to find a compatible cost function for the journey that would occur if there was no traffic light at the junction. Thus, here the waiting time function is always 0, as we have the standard nonatomic congestion game model. We choose affine cost functions due to their simplicity. The cost function[2] found from the simulation was $C(\boldsymbol{x}) = 0.4x + 44$. To check the dependency on road length, the first edge was extended to 1000m which gave an affine cost function of $C(\boldsymbol{x}) = 0.5x + 80$. The gradient of the cost function change is minor, it is the constant accounts for most of the change.

To confirm the hypothesis that congestion is independent of road length, we can add in a traffic light and compare the new cost functions. Therefore, we simulated a 1050m journey with a traffic light that has a cycle 20 seconds red light and 20 seconds of green light has cost function of $C(x) = 4.4x - 11$. We compared this to a simulation of a 550m journey with a traffic light that has a repeated cycle of 20 seconds of red light and 20 seconds of green has a cost of $C(x) = 4.4x - 9$. Again, we see that there is no change in congestion affected by the length of the road. Thus, we will make the simplifying assumption that the congestion coefficient is not dependent on a road length parameter. To find the relationship between the waiting time w_e and the traffic light cycle $\{t_r^e, t_g^e\}$, we use a 550m journey.

In these simulations, we set the amber phase of the cycle to be 3 seconds long. The amber light is to allow adequate reaction time for drivers to avoid collisions and emergency braking. Hence, we can reduce the phase to cycles of either red or green time by merging with the amber light phase. If the light is amber, this is added to the green cycle time, and if it shows red and amber, then this is added to the red cycle time. For example, a cycle that is red for 17 seconds, then red and amber for 3 seconds, green for 17 seconds and finally amber for 3 seconds would be written as $t_r^e = 20, t_g^e = 20$.

Any fixed (or static) traffic light cycle can be described through the total cycle time $T^e = t_r^e + t_g^e$ and the proportion of red time in a cycle $p_e = \frac{t_r^e}{t_r^e + t_g^e}$. We test the effects of changing these two variables on the cost functions and waiting times by assuming the form of $c_e(x, T_e, p_e) = \bar{c}_e(x) + w_e(x, T_e, p_e)$.

Figure A.2 plots the cost functions for a number of simulations. Each line

[2]Note we change the notation from C_i to C here since we only consider one population.

represents the cost function found from linear regression of the data points[3]. Lines are coloured using the p and T values to indicate the correlation between waiting time and red cycle proportion. Figure A.3 shows the combinations of p_e and T_e used in the simulations.

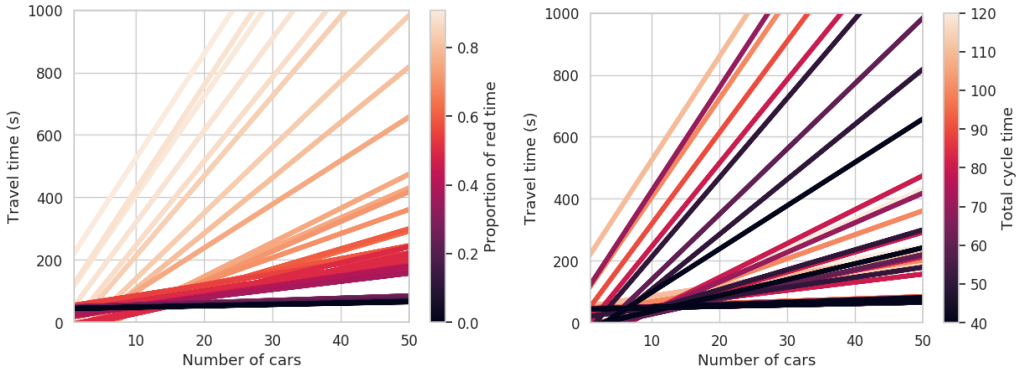

Figure A.2: Simulated affine cost functions are the expected journey time found from regressions of simulated data points. Left shows the relationship between the cost functions and values of $p \in [0, 1]$, and right shows with $T \in [40, 120]$. There is a clear correlation between the proportion of red light in a cycle p and the expected travel times (left). The length of the cycle time T showed no correlation to journey times (right).

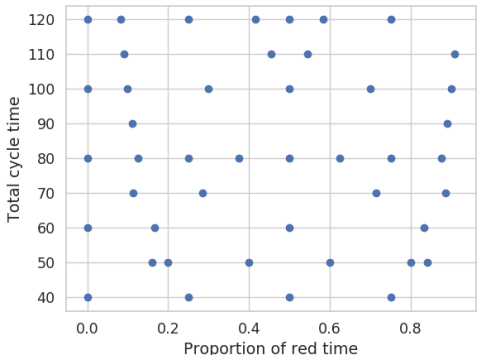

Figure A.3: Sampled values of T and p for the SUMO simulations.

The expected travel time is composed of travel time on the edge plus waiting

[3]Because we chose to use a linear regression and the relationship is nonlinear, there are some negative travel times predicted in Figure A.2.

time at a traffic light, i.e.,

$$c_e(f_e(\boldsymbol{x})) = \bar{c}_e(f_e(\boldsymbol{x})) + w_e(f_e(\boldsymbol{x}), t_r^e + t_g^e, \frac{t_r^e}{t_r^e + t_g^e}),$$

or

$$c_e(f_e(\boldsymbol{x})) = \bar{c}_e(f_e(\boldsymbol{x})) + w_e(f_e(\boldsymbol{x}), T_e, p_e).$$

Additionally, the system was simulated without a traffic light to find $\bar{c}$ to be $\bar{c}(x) = 0.4x + 44$. We find the form of the waiting time functions as follows. Take a linear regression to find function c, and then extract the waiting time function using $w(x)$=$c(x) - \bar{c}(x)$. For example, for the cycle of 20 seconds of red light and 20 seconds of green light, i.e. $T = t_r + t_g = 40$, $p = \frac{t_r}{t_r + t_g} = 1/2$, the simulation gives us $c(x) = 4.4x - 9$. So, we have $w(x) = 4x - 53$. For this example, we say that the waiting time coefficient dependent on congestion is 4, and the time independent of congestion is -53. Figure A.4 shows the relationship between these values for the simulations plotted against the proportion of red time p. This indicates how $\bar{c}$ and w depend on p.

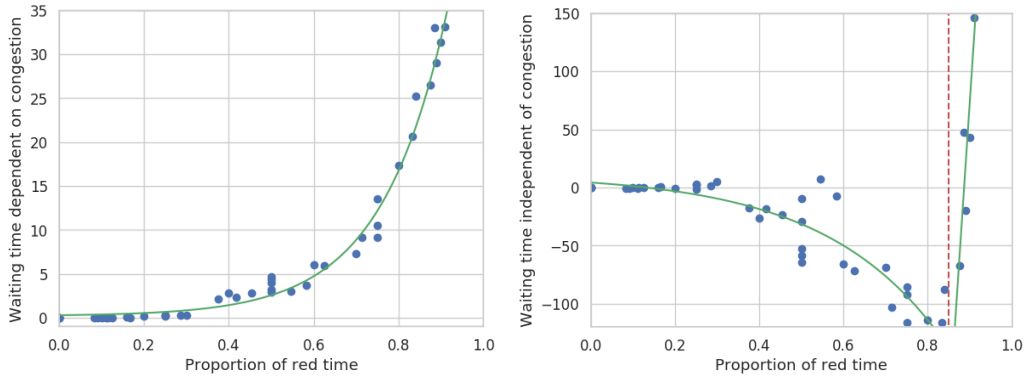

Figure A.4: Left: Waiting time dependent on edge congestion. Right: Waiting time independent of edge congestion.

From the figure, we see that the time dependent on congestion is exponential in p whereas the time independent of congestion only follows an exponential distribution until a threshold value of red proportion and is then linear. The relationship between $\bar{c}$ and p is monotonic and can be defined using an exponential function. However, the waiting time function w cannot be written as one function due to the change that occurs after p increases past a threshold value. If we restrict values of p to be lower than this value, then we can also use an exponential function to describe

their relationship.

After a threshold of 85% red time, the waiting time increases significantly. Yet, in real traffic light systems, it is unlikely any will have a cycle with more than 85% red time. Since each traffic light p is dependent on at least one other edge there must also exist a lower bound depending on the topology of the junction. This aligns with our theoretical model in Section 6.4, where we stated that for every $e \in E$ where there exists a traffic light, its cycle must be bounded by p_- and p_+ such that $0 < p_- \leq p_e \leq p_+ < 1$.

The total cycle time would be expected to affect the waiting times, however, we do not see a correlation between them in SUMO simulations. Figure A.5 shows that there is no correlation between the total cycle time and the waiting time functions. The correlations only exist between p and expected waiting times, so we can update our cost function to be $c_e(x, p_e) = \bar{c}_e(x) + w_e(x, p_e)$.

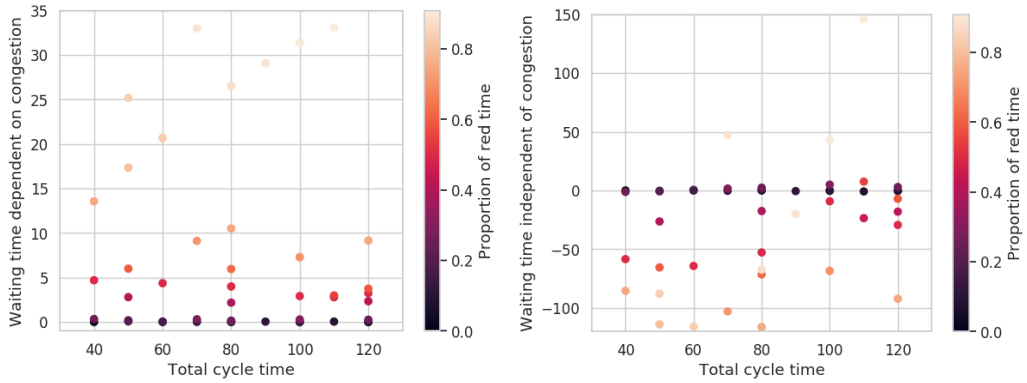

Figure A.5: Left: Waiting time dependent on congestion. Right: The waiting times seem independent of total cycle time. Any correlation seen between the total cycle time and waiting times are explained by red time proportions.

The expected cost of the edge in our SUMO simulations can be written as:

$$c(x, p) = \bar{c}(x) + (0.12e^{6.12p} + 0.16)x + f(p),$$

where x is congestion, $\bar{c}(x) = 0.4x + 44$, $p = \frac{t_r}{t_r + t_g}$, and f is the function

$$f(x) = \begin{cases} -10.51e^{3.05p} + 14.51 & \text{if } p < 0.85 \\ 5454.73p - 4834.45 & \text{otherwise} \end{cases}$$

This format is in line with the assumptions we made to formulate the traffic light game. Since the traffic light game is compatible with SUMO, later we will use

SUMO to create simulated data to combine with the theory.

APPENDIX B

Multi-agent Reinforcement Learning Traffic Lights Simulation

Here, we substantiate the result from Theorem 6.7.3, that myopic learning algorithms converge to the same equilibria, by simulating an example network with different RL algorithms. Each learns the optimal traffic light parameters p_e for edges e that end at the traffic light node.

The tested MARL algorithms are independent Q-learning, multi-agent Q-learning, and coordinated Q-learning as well as independent deep Q-learning and multi-agent deep Q-learning. Pseudocode for these algorithms is included in Chapter 2. We include this simulation to test our theory since the literature suggests that multi-agent and coordinated algorithms outperform independent implementation of RL traffic lights, see e.g., Bazzan & Klügl (2014); Prabuchandran *et al.* (2014).

We use the same multi-population example as before, shown in Figure B.1. Here, the waiting time functions are $w(x, p) = x(e^p - 1)$ and driving edge-costs are $\bar{c}(x) = x$. There are two traffic lights TL1 and TL2 that must learn the optimal value of p and q respectively. All other traffic lights have their p parameter set to 0.5.

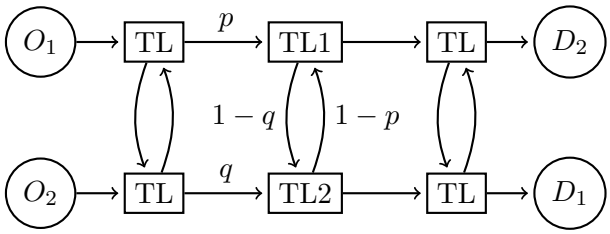

Figure B.1: TL1 and TL2 learn to play values p and q such that $p, q \in [0.15, 0.85]$.

The hyperparameters used for the MARL algorithms are included in Tables B.1 and B.2, selected for fast convergence.

Table B.1: MARL Algorithms - Q-Learning Parameters

Parameter	**Value**
Learning rate	0.2
Exploration rate	$\max\{0.8 * 0.99^t, 0.005\}$
Discount factor	0.9

Table B.2: MARL Algorithms - DQN Parameters

Parameter	**Value**
Learning rate	0.001
Exploration rate	$\max\{0.99^t, 0.01\}$
Discount factor	0.99
Batch size	64
Experience replay size	10000
Neural network	1 full connected layer, size 16
Activation function	rectified linear unit

Figure B.2 shows that each algorithm converged to the same value of p, as predicted from Theorem 6.7.3. The convergence rates are similar for all algorithms except independent Q, which converges the fastest. Another noticeable difference is that the multi-agent algorithms are noisier than the independent implementations.

Figure B.3 shows that the social costs of the algorithms while training. All algorithms converge to the same social cost.

From these figures, we see that the independent implementation of Q-learning converges to the solution fastest and is also the least noisy. This suggests that the simple implementation of independent local agents has advantages to the more complex cooperation mechanisms studied in intelligent traffic lights, such as coordinated reinforcement learning methods Pol & Oliehoek (2016). Further study on more complex simulators is needed to draw concrete conclusions from these results. However, the simulations highlight evidence of the counter-intuitive result that multi-agent cooperation mechanisms do not improve upon the optimal policy of simpler algorithms.

Figure B.2: Different MARL algorithms training curves for selecting a value for p.

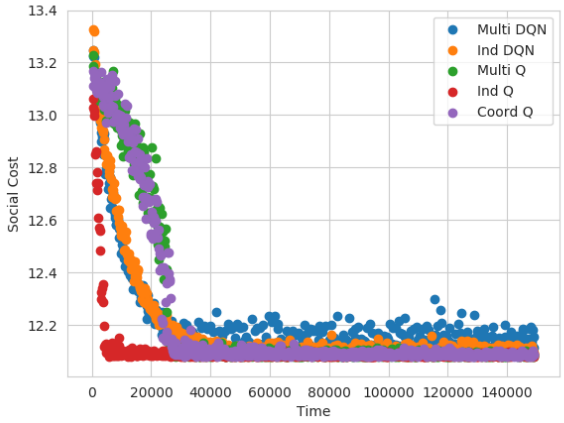

Figure B.3: The training curves showing the social cost for the different MARL algorithms.

Bibliography

Acemoglu, Daron, Makhdoumi, Ali, Malekian, Azarakhsh, & Ozdaglar, Asu. 2018. Informational Braess' paradox: The effect of information on traffic congestion. *Operations Research*, **66**(4), 893–917.

Ackermann, Heiner, Roglin, Heiko, & Vocking, Berthold. 2009. Pure Nash equilibria in player-specific and weighted congestion games. *Theoretical Computer Science*, **410**(17), 634–654.

Albrecht, Stefano V., & Stone, Peter. 2018. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence*, **258**, 66–95.

Altman, Eitan, Reiffers, Alexandre, Menasché, Daniel S., Datar, Mandar, Dhamal, Swapnil, & Touati, Corinne. 2019. Mining competition in a multi-cryptocurrency ecosystem at the network edge: A congestion game approach. *Performance Evaluation Review*, **46**(3), 114–117.

Alvarez, I, Poznyak, A, & Malo, A. 2008. Urban Traffic Control Problem a Game Theory Approach. *IEEE Conference on Decision and Control*, **41**(2), 2168–2172.

Anselmi, Jonatha, Ardagna, Danilo, & Passacantando, Mauro. 2014. Generalized Nash equilibria for SaaS/PaaS Clouds. *European Journal of Operational Research*, **236**(1), 326–339.

Arnott, Richard, Palma, Andre De, & Lindsey, Robin. 1991. Does providing information to drivers reduce traffic congestion? *Transportation Research Part A: General*, **25**(5), 309–318.

Axelrod, Robert, & Hamilton, William Donald. 1981. The evolution of cooperation. *Science*, **211**(4489), 1390–1396.

Bazzan, Ana L.C., & Klügl, Franziska. 2014. A review on agent-based technology for traffic and transportation. *Knowledge Engineering Review*, **29**(3), 375–403.

Beckmann, Martin, McGuire, C B, & Winsten, Christopher B. 1956. *Studies in the Economics of Transportation*. Yale University Press.

Beier, René, Czumaj, Artur, Krysta, Piotr, & Vöcking, Berthold. 2004. Computing equilibria for congestion games with (im)perfect information. *Pages 746–755 of: Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*. SIAM.

Bergemann, Dirk, & Morris, Steven. 2013. Robust Predictions in Games With Incomplete Information. *Econometrica*, **81**(4), 1251–1308.

Bertsimas, Dimitris, Farias, Vivek F., & Trichakis, Nikolaos. 2012. On the efficiency-fairness trade-off. *Management Science*, **58**(12), 2234–2250.

Bondy, John Adrian, Murty, Uppaluri Siva Ramachandra, *et al.* 1976. *Graph theory with applications*. Vol. 290. Macmillan London.

Bonifaci, Vincenzo, Harks, Tobias, & Schäfer, Guido. 2010. Stackelberg routing in arbitrary networks. *Mathematics of Operations Research*, **35**(2), 330–346.

Bonifaci, Vincenzo, Salek, Mahyar, & Schäfer, Guido. 2011. Efficiency of restricted tolls in non-atomic network routing games. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, **6982 LNCS**, 302–313.

Bonnefon, Jean-françois, Shariff, Azim, & Rahwan, Iyad. 2016. The social dilemma of autonomous vehicles. *Science*, **352**(6293), 1573–1576.

Braess, Dietrich. 1968. Uber ein Paradoxon aus der Verkehrsplanung. *Unternehmensforschung*, **12**(1), 258–268.

Bui, Khac-hoai Nam, & Jung, Jason J. 2017. Cooperative game-theoretic approach to traffic flow optimization for multiple intersections. *Computers and Electrical Engineering*, **71**, 1012–1024.

Buşoniu, Lucian, Babuška, Robert, & De Schutter, Bart. 2008. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, **38**(2), 156–172.

Capraro, Valerio, & Halpern, Joseph Y. 2019. Translucent players: Explaining cooperative behavior in social dilemmas. *Rationality and Society*, **31**(4), 371–408.

Capraro, Valerio, Venanzi, Matteo, Polukarov, Maria, & Jennings, Nicholas R. 2013. Cooperative equilibria in iterated social dilemmas. *Pages 146–158 of: International Symposium on Algorithmic Game Theory.*

Castiglioni, Matteo, Marchesi, Alberto, & Gatti, Nicola. 2019a. Be a leader or become a follower: The strategy to commit to with multiple leaders. *Pages 123–129 of: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence.*

Castiglioni, Matteo, Marchesi, Alberto, Gatti, Nicola, & Coniglio, Stefano. 2019b. Leadership in singleton congestion games: What is hard and what is easy. *Artificial Intelligence*, **277**, 103177.

Chakraborty, Doran, & Stone, Peter. 2014. Multiagent learning in the presence of memory-bounded agents. *Autonomous Agents and Multi-Agent Systems*, **28**(2), 182–213.

Chen, Chen, Li, Yajuan, & Pei, Qingqi. 2014. Avoiding Information Congestion in VANETs: A Congestion Game Approach. *Pages 105–110 of: 2014 IEEE International Conference on Computer and Information Technology.*

Chen, Ho Lin, Roughgarden, Tim, & Valiant, Gregory. 2010. Designing network protocols for good equilibria. *SIAM Journal on Computing*, **39**(5), 1799–1832.

Chen, Xujin, Diao, Zhuo, & Hu, Xiaodong. 2015. Excluding braess's paradox in nonatomic selfish routing. *Pages 309–318 of: International Symposium on Algorithmic Game Theory.* Berlin, Heidelberg: Springer.

Cheng, Aaron, Pang, Min-Seok, & Pavlou, Paul A. 2020. Mitigating traffic congestion: The role of intelligent transportation systems. *Information Systems Research*, **31**(3), 653–674.

Cheung, Man Wah, & Lahkar, Ratul. 2018. Nonatomic potential games: the continuous strategy case. *Games and Economic Behavior*, **108**(71601102), 341–362.

Chouhan, Aaditya Prakash, & Banda, Gourinath. 2018. Autonomous Intersection Management : A Heuristic Approach. *IEEE Access*, **6**, 53287–53295.

Cole, Richard, Lianeas, Thanasis, & Nikolova, Evdokia. 2018. When Does Diversity of Agent Preferences Improve Outcomes in Selfish Routing? *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 173–179.

Correa, J.R., Schulz, A.S., & Nicolás, E. 2005. On the Inefficiency of Equilibria in Congestion Games. *Pages 167–181 of: International Conference on Integer Programming and Combinatorial Optimization.* Springer Berlin Heidelberg.

da Silva, Bruno C., Basso, Eduardo W., Bazzan, Ana L. C., & Engel, Paulo M. 2006. Dealing with non-stationary environments using context detection. *In Proceedings of the 23rd international conference on Machine learning*, 217–224.

Dafoe, Allan, Hughes, Edward, Bachrach, Yoram, Collins, Tantum, McKee, Kevin R., Leibo, Joel Z., Larson, Kate, & Graepel, Thore. 2020. Open problems in cooperative AI. *arXiv preprint arXiv:2012.08630.*

Daganzo, Carlos F, & Sheffi, Yosef. 1978. Another "paradox" of traffic flow. *Transportation Research*, **12**(1), 43–46.

Das, Sanmay, Kamenica, Emir, & Mirka, Renee. 2017. Reducing congestion through information design. *55th Annual Allerton Conference on Communication, Control, and Computing, Allerton 2017*, **2018-Janua**, 1279–1284.

de Melo, Celso M, Khooshabeh, Peter, Amir, Ori, & Gratch, Jonathan. 2018. Shaping Cooperation between Humans and Agents with Emotion Expressions and Framing. *Pages 2224–2226 of: Proc. of the 17th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2018).*

Eccles, Tom, Hughes, Edward, Kramár, János, Wheelwright, Steven, & Leibo, Joel Z. 2019a. Learning Reciprocity in Complex Sequential Social Dilemmas. *arXiv preprint arXiv:1903.08082.*

Eccles, Tom, Hughes, Edward, Kramar, Janos, Wheelwright, Steven, & Leibo, Joel Z. 2019b. The Imitation Game: Learned Reciprocity in Markov games. *Pages 1934–1936 of: 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019).*

Elhenawy, Mohammed, Elbery, Ahmed A, Hassan, Abdallah A, & Tech, Virginia. 2015. An Intersection Game-Theory-Based Traffic Control Algorithm in a Connected Vehicle Environment. *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 343–347.

Epstein, Amir, Feldman, Michal, & Mansour, Yishay. 2009. Efficient graph topologies in network routing games. *Games and Economic Behavior*, **66**(1), 115–125.

Fabrikant, Alex, Papadimitriou, Christos, & Talwar, Kunal. 2004. The complexity of pure Nash equilibria. *Conference Proceedings of the Annual ACM Symposium on Theory of Computing*, 604–612.

Fisk, Caroline. 1980. Some developments in equilibrium traffic assignment. *Transportation Research Part B: Methodological*, **14**(3), 243–255.

Fujishige, Satoru, Goemans, Michel X, Harsk, Tobias, Peis, Britta, & Zenklusen, Rico. 2017. Matroids are immune to Braess' paradox. *Mathematics of Operations Research*, **42**(3), 745–761.

Ganzfried, Sam, & Sandholm, Tuomas. 2012. Safe opponent exploitation. *Proceedings of the Adaptive and Learning Agents Workshop 2012, ALA 2012 - Held in Conjunction with the 11th International Conference on Autonomous Agents and Multiagent Systems, (AAMAS 2012)*, **1**(212), 119–126.

Gardner, Roy, Ostrom, Elinor, & Walker, James. 1984. Social capital and cooperation: Communication, bounded rationality, and behavioral heuristics. *Pages 375–411 of: Social Dilemmas and Cooperation*. Springer.

Gawron, Christian. 1998a. An Iterative Algorithm to determine the Dynamic User Equlibrium in a Traffic Simulation Model. *International Journal of Modern Physics*, **9**(3), 393–407.

Gawron, Christian. 1998b. *Simulation-based traffic assignment: Computing user equilibria in large street networks*. Ph.D. thesis, Koln University.

Gers, Felix A., Schmidhuber, Jurgen, & Cummins, Fred. 1999. Learning to Forget: Continual Prediction with LSTM. *Pages 850–855 of: 9th International Conference on Artificial Neural Networks: ICANN '99*.

Gleave, Adam, Dennis, Michael, Wild, Cody, Kant, Neel, Levine, Sergey, & Russell, Stuart. 2019. Adversarial Policies: Attacking Deep Reinforcement Learning. *arXiv preprint arXiv:1905.10615*.

Glynatsi, Nikoleta E., & Knight, Vincent A. 2020. Using a theory of mind to find best responses to memory-one strategies. *Scientific Reports*, **10**(1), 1–9.

Guestrin, Carlos, Lagoudakis, Michail, & Parr, R. 2002. Coordinated Reinforcement Learning. *ICML*, **Vol. 2**, 227–234.

Haïkel Yaïche, Mazumdar, Ravi R, & Rosenberg, Catherine. 2000. A Game Theoretic Framework for Bandwidth Allocation and Pricing in Broadband Networks. *IEEE/ACM Transactions on Networking*, **8**(5), 667–678.

Hao, Yaqi, Pan, Sisi, Qiao, Yupeng, & Cheng, Daizhan. 2018. Cooperative Control via Congestion Game Approach. *IEEE Transactions on Automatic Control*, **63**(12), 4361–4366.

Helbing, Dirk, Schönhof, Martin, Stark, Hans Ulrich, & Holyst, Janusz A. 2005. How individuals learn to take turns: Emergence of alternating cooperation in a congestion game and the prisoner's dilemma. *Advances in Complex Systems*, **8**(1), 87–116.

Hernandez-Leal, Pablo, Kaisers, Michael, Baarslag, Tim, & de Cote, Enrique Munoz. 2017. A Survey of Learning in Multiagent Environments: Dealing with Non-Stationarity. *arXiv preprint arXiv:1707.09183*.

Hernandez-Leal, Pablo, Kartal, Bilal, & Taylor, Matthew E. 2018. Is multiagent deep reinforcement learning the answer or the question? A brief survey. *Learning 21*, 22.

Hoegen, Rens, Stratou, Giota, & Gratch, Jonathan. 2017. Incorporating emotion perception into opponent modeling for social dilemmas. *Pages 801–809 of: Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*.

Hu, Junling, & Wellman, Michael P. 1998. Multiagent Reinforcement Learning: Theoretical Framework and an Algorithm. *Pages 242–250 of: ICML*, vol. 98.

Hughes, Edward, Leibo, Joel Z., Phillips, Matthew, Tuyls, Karl, Dueñez-Guzman, Edgar, Castañeda, Antonio García, Dunning, Iain, Zhu, Tina, McKee, Kevin, Koster, Raphael, Roff, Heather, & Graepel, Thore. 2018. Inequity aversion improves cooperation in intertemporal social dilemmas. *Advances in Neural Information Processing Systems*, **2018-Decem**(NeurIPS), 3326–3336.

Hughes, Edward, Anthony, Thomas W., Eccles, Tom, Leibo, Joel Z., Balduzzi, David, & Bachrach, Yoram. 2020. Learning to Resolve Alliance Dilemmas in Many-Player Zero-Sum Games.

Ibars, Christian, Navarro, Monica, & Giupponi, Lorenza. 2010. Distributed Demand Management in Smart Grid with a Congestion Game. *Pages 495–500 of: First IEEE International Conference on Smart Grid Communications.* IEEE.

Ikegami, Kei, Okumura, Kyohei, & Yoshikawa, Takumi. 2020. A Simple, Fast, and Safe Mediator for Congestion Management. *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**(02), 2030–2037.

Jaques, Natasha, Lazaridou, Angeliki, Hughes, Edward, Gulcehre, Caglar, Ortega, Pedro A., Strouse, D. J., Leibo, Joel Z., & de Freitas, Nando. 2019. Social influence as intrinsic motivation for multi-agent deep reinforcement learning. *36th International Conference on Machine Learning, ICML 2019*, **2019-June**, 5372–5381.

Karakostas, George, & Kolliopoulos, Stavros G. 2009. Edge pricing of multicommodity networks for selfish users with elastic demands. *Algorithmica (New York)*, **53**(2), 225–249.

Karush, W. 1939. *Minima of Functions of Several Variables with Inequalities as Side Constraints*. Ph.D. thesis.

Klein, Ido, & Ben-Elia, Eran. 2016. Emergence of cooperation in congested road networks using ICT and future and emerging technologies: A game-based review. *Transportation Research Part C: Emerging Technologies*, **72**, 10–28.

Kok, Jelle R., & Vlassis, Nikos. 2005. Using the max-plus algorithm for multiagent decision making in coordination graphs. *Belgian/Netherlands Artificial Intelligence Conference*, 359–360.

Konda, Vijay R., & Tsitsiklis, John N. 2000. Actor-critic algorithms. *Advances in neural information processing systems*, 1008–1014.

Korilis, Yannis A., Lazar, Aurel A., & Orda, Ariel. 1997. Achieving network optima using Stackelberg routing strategies. *IEEE/ACM Transactions on Networking*, **5**(1), 161–173.

Köster, Raphael, Hadfield-Menell, Dylan, Hadfield, Gillian K., & Leibo, Joel Z. 2020. Silly rules improve the capacity of agents to learn stable enforcement and compliance behaviors. *Pages 1887–1888 of: Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9-13, 2020, IFAAMAS, 2 pages.*

Koutsoupias, Elias, & Papadimitriou, Christos. 1999. Worst-case equilibria. *Pages 404–413 of: Annual Symposium on Theoretical Aspects of Computer Science*. Springer.

Kuhn, H. W., & Tucker, A. W. 1951. Nonlinear programming. *Pages 481–492 of: Proceedings of 2nd Berkeley Symposium.*

Kuyer, Lior, Whiteson, Shimon, Bakker, Bram, & Vlassis, Nikos. 2008. Multiagent reinforcement learning for Urban traffic control using coordination graphs. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, **5211 LNAI**(PART 1), 656–671.

Laszka, Aron, Potteiger, Bradley, Vorobeychik, Yevgeniy, Amin, Saurabh, & Koutsoukos, Xenofon. 2016. Vulnerability of Transportation Networks to Traffic-Signal Tampering. *Pages 1–10 of: ACM/IEEE 7th International conference on Cyber-Physical Systems (ICCPS).*

Leibo, Joel Z., Zambaldi, Vinicius, Lanctot, Marc, Marecki, Janusz, & Graepel, Thore. 2017. Multi-agent reinforcement learning in sequential social dilemmas. *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, **1**, 464–473.

Lerer, Adam, & Peysakhovich, Alexander. 2017. Maintaining cooperation in complex social dilemmas using deep reinforcement learning. *arXiv.*

Lerer, Adam, & Peysakhovich, Alexander. 2019. Learning existing social conventions via observationally augmented self-play. *AIES 2019 - Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, 107–114.

Liu, Jeffrey, Amin, Saurabh, & Schwartz, Galina. 2016. Effects of information heterogeneity in bayesian routing games. *arXiv preprint arXiv:1603.08853.*

Liu, Mingyan, & Wu, Yunnan. 2008. Spectum sharing as congestion games. *Pages 1146–1153 of: 2008 46th Annual Allerton Conference on Communication, Control, and Computing.* IEEE.

Lopez, Pablo Alvarez, Behrisch, Michael, Bieker-Walz, Laura, Erdmann, Jakob, Flotterod, Yun Pang, Hilbrich, Robert, Lucken, Leonhard, Rummel, Johannes, Wagner, Peter, & Wiebner, Evamarie. 2018. Microscopic Traffic Simulation using SUMO. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, **2018-Novem**, 2575–2582.

Lujak, Marin, Giordani, Stefano, & Ossowski, Sascha. 2015. Route guidance: Bridging system and user optimization in traffic assignment. *Neurocomputing*, **151**(P1), 449–460.

Macy, Michael W, & Flache, Andreas. 2002. Learning dynamics in social dilemmas. *Proceedings of the National Academy of Sciences*, **99**(suppl 3), 7229–7236.

Mak, Vincent, & Rapoport, Amnon. 2013. The price of anarchy in social dilemmas: Traditional research paradigms and new network applications. *Organizational Behavior and Human Decision Processes*, **120**(2), 142–153.

Mannion, Patrick, Duggan, Jim, & Howley, Enda. 2016. An Experimental Review of Reinforcement Learning Algorithms for Adaptive Traffic Signal Control. *Autonomic Road Transport Support Systems*, 47–66.

Marden, Jason R., & Wierman, Adam. 2013. Distributed welfare games. *Operations Research*, **61**(1), 155–168.

Maschler, Michael, Solan, Elion, & Zamir, Schmuel. 2013. *Game Theory*. Cambridge University Press.

Masuda, Naoki, & Ohtsuki, Hisashi. 2009. A theoretical analysis of temporal difference learning in the iterated Prisoner's dilemma game. *Bulletin of Mathematical Biology*, **71**(8), 1818–1850.

McKee, Kevin R, Gemp, Ian, McWilliams, Brian, Duéñez-Guzmán, Edgar A, Hughes, Edward, & Leibo, Joel Z. 2020. Social diversity and social preferences in mixed-motive reinforcement learning. *arXiv preprint arXiv:2002.02325*.

Meir, Reshef, & Parkes, David. 2018. Playing the Wrong Game: Bounding Externalities in Diverse Populations of Agents. *Pages 86–94 of: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*.

Meir, Reshef, & Parkes, David C. 2015. Congestion Games with Distance-Based Strict Uncertainty. *Proceedings of the AAAI Conference on Artificial Intelligence*, **Vol. 29**(No. 1), 986–992.

Meir, Reshef, & Parkes, David C. 2016. When are Marginal Congestion Tolls Optimal? *In: ATT@ IJCAI*.

Milchtaich, Igal. 1996. Congestion games with player-specific payoff functions. *Games and Economic Behavior*, **13**(1), 111–124.

Milchtaich, Igal. 2006. Network topology and the efficiency of equilibrium. *Games and Economic Behavior*, **57**(2), 321–346.

Mnih, Volodymyr, Kavukcuoglu, Koray, Silver, David, Graves, Alex, Antonoglou, Ioannis, Wierstra, Daan, & Riedmiller, Martin. 2013. Playing Atari with Deep Reinforcement Learning. *arXiv preprint arXiv:1312.5602*.

Mnih, Volodymyr, Badia, Adria Puigdomenech, Mirza, Lehdi, Graves, Alex, Harley, Tim, Lillicrap, Timothy P., Silver, David, & Kavukcuoglu, Koray. 2016. Asynchronous methods for deep reinforcement learning. *33rd International Conference on Machine Learning, ICML 2016*, **4**, 2850–2869.

Monderer, Dov, & Shapley, Lloyd. 1996. Potential Games. *Games and Economic Behaviour*, **14**, 124–143.

Mousavi, Seyed Sajad, Schukat, Michael, & Howley, Enda. 2017. Traffic light control using deep policy-gradient and value-function-based reinforcement learning. *IET Intelligent Transport Systems*, **11**(7), 417–423.

Murchland, John D. 1970. Braess's paradox of traffic flow. *Transportation Research*, **4**(4), 391–394.

Murphy, Ryan O., & Ackermann, Kurt A. 2015. Social preferences, positive expectations, and trust based cooperation. *Journal of Mathematical Psychology*, **67**, 45–50.

Nikitina, Natalia, Ivashko, Evgeny, & Tchernykh, Andrei. 2018. Congestion game scheduling for virtual drug screening optimization. *Journal of computer-aided molecular design*, **32**(2), 363–374.

Nowé, Ann, Vrancx, Peter, & De Hauwere, Yann Michaël. 2012. Game theory and multi-agent reinforcement learning. *Adaptation, Learning, and Optimization*, **12**, 441–470.

Orda, Ariel, Rom, Raphael, & Shimkin, Nahum. 1993a. Competitive Routing in Multiuse Communication Networks. *IEEE/ACM Transactions on Networking*, **1**(5), 510–521.

Orda, Ariel, Rom, Raphael, & Shimkin, Nahum. 1993b. Competitive routing in multiuser communication networks. *IEEE/ACM Transactions on networking*, **1**(5), 510–521.

Oxley, James G. 2006. *Matroid theory*. Vol. 3. Oxford University Press, USA.

Panait, Liviu, & Luke, Sean. 2005. Cooperative Multi-Agent Learning: The State of the Art. *Autonomous agents and multi-agent systems*, **11**(3), 387–434.

Pas, Eric I, & Principio, Shari L. 1997. Braess' paradox: Some new insights. *Transportation Research Part B: Methodological*, **31**(3), 265–276.

Pigou, Arthur Cecil. 1920. *The Economics of Welfare*. Palgrave Macmillan.

Pol, Elise van der, & Oliehoek, Frans A. 2016. Coordinated deep reinforcement learners for traffic light control. *30th Conference on Neural Information Processing Systems (NIPS 2016)*.

Prabuchandran, K. J., Hemanth Kumar, A. N., & Bhatnagar, Shalabh. 2014. Multi-agent reinforcement learning for traffic signal control. *2014 17th IEEE International Conference on Intelligent Transportation Systems, ITSC 2014*, 2529–2534.

Press, William H., & Dyson, Freeman J. 2012. Iterated Prisoner's Dilemma contains strategies that dominate any evolutionary opponent. *Proceedings of the National Academy of Sciences of the United States of America*, **109**(26), 10409–10413.

Ramos, Gabriel de Oliveira, Rădulescu, Roxana, Nowé, Ann, & Tavares, Anderson Rocha. 2020. Toll-Based Learning for Minimising Congestion under Heterogeneous Preferences. *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020)*, 1098–1106.

Rand, David G., & Nowak, Martin A. 2013. Human cooperation. *Trends in Cognitive Sciences*, **17**(8), 413–425.

Rapoport, Amnon, Kugler, Tamar, Dugar, Subhasish, & Gisches, Eyran J. 2009. Choice of routes in congested traffic networks: Experimental tests of the Braess Paradox. *Games and Economic Behavior*, **65**(2), 538–571.

Rapoport, Anatol, Chammah, Albert M., & Orwant, Carol J. 1965. *Prisoner's Dilemma: A Study in Conflict and Cooperation*. University of Michigan Press.

Roman, Charlotte, & Turrini, Paolo. 2019. Multi-Population Congestion Games with Incomplete Information. *Pages 565–571 of: Proceedings of the 28th International Joint Conference on Artificial Intelligence*. AAAI Press.

Roman, Charlotte, Dennis, Michael, Critch, Andrew, & Russell, Stuart. 2021. Accumulating Risk Capital Through Investing in Cooperation. *Pages 1073–1081 of: AAMAS '21: Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*.

Rosenthal, Robert W. 1973. A Class of Games Possessing Pure-Strategy Nash Equilibria. *International Journal of Game Theory*, **2**(1), 65–67.

Rossi, Federico, Zhang, Rick, Hindy, Yousef, & Pavone, Marco. 2018. Routing autonomous vehicles in congested transportation networks: Structural properties and coordination algorithms. *Autonomous Robots*, **42**(7), 1427–1442.

Roughgarden, Tim. 2003. The price of anarchy is independent of the network topology. *Journal of Computer and System Sciences*, **67**(2), 341–364.

Roughgarden, Tim. 2004. Stackelberg Scheduling Strategies. *SIAM Journal on Computing*, **33**(2), 332–350.

Roughgarden, Tim. 2005. *Selfish Routing and the Price of Anarchy*. The MIT Press.

Roughgarden, Tim, & Schoppmann, Florian. 2015. Local smoothness and the price of anarchy in splittable congestion games. *Journal of Economic Theory*, **156**, 317–342.

Rozenfeld, Ola, & Tennenholtz, Moshe. 2007. Routing mediators. *IJCAI International Joint Conference on Artificial Intelligence*, 1488–1493.

Sandholm, William H. 2001. Potential Games with Continuous Player Sets. *Journal of Economic Theory*, **97**(1), 81–108.

Sandholm, William H. 2002. Evolutionary implementation and congestion pricing. *Review of Economic Studies*, **69**(3), 667–689.

Schmeidler, David. 1973. Equilibrium points of non-atomic games. *Journal of Statistical Physics*, **7**(4), 295–300.

Sekar, Shreyas, Zheng, Liyuan, Ratliff, Lillian J, & Zhang, Baosen. 2018. Uncertainty in Multi-Commodity Routing Networks: When does it help? *Annual American Control Conference (ACC)*, 6553–6558.

Sharon, Guni, Hanna, Josiah P., Rambha, Tarun, Levin, Michael W., Albert, Michael, Boyles, Stephen D., & Stone, Peter. 2017. Real-Time adaptive tolling scheme for optimized social welfare in traffic networks. *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, **2**(May), 828–836.

Sheffi, Yosef. 1985. *Urban Transportation Networks*. Vol. 6. Prentice-Hall, Englewood Cliffs, NJ.

Sheffi, Yosef, & Daganzo, Carlos F. 1977. On stochastic models of traffic assignment. *Transportation Science*, **11**(3), 253–274.

Shoham, Yoav, & Leyton-Brown, Kevin. 2008. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press.

Smith, M. J. 1979. The existence, uniqueness and stability of traffic equilibria. *Transportation Research Part B*, **13**(4), 295–304.

Smith, M. J. 1981. Properties of a traffic control policy which ensure the existence of a traffic equilibrium consistent with the policy. *Transportation Research Part B*, **15**(6), 453–462.

Smith, M. J. 1985. Traffic signals in assignment. *Transportation Research Part B*, **19**(2), 155–160.

Smith, M.J., & Van Vuren, T. 1993. Traffic Equilibrium with Responsive Traffic Control. *Transportation Science*, **27**(2), 118–132.

Southwell, Richard, Chen, Xu, & Huang, Jianwei. 2013. QoS satisfaction games for spectrum sharing. *Pages 570–574 of: 2013 Proceedings IEEE INFOCOM*.

Stark, Hans-Ulrich, Helbing, Dirk, Schoenhof, Martin, & Holyst, Janusz A. 2008. Alternating cooperation strategies in a Route Choice Game: Theory, experiments, and effects of a learning scenario. *Games, Rationality, and Behaviour*, **Houndmills**, 256–273.

Suri, Subhash, Tóth, Csaba D, & Zhou, Yunhong. 2004. Uncoordinated load balancing and congestion games in p2p systems. *Pages 123–130 of: International Workshop on Peer-to-Peer Systems*. Springer.

Sutton, Richard, & Barto, Andrew. 2018. *Reinforcement Learning: An Introduction*. The MIT Press.

Sutton, Richard S. 1988. Learning to Predict by the Methods of Temporal Differences. *Machine Learning*, **3**(1), 9–44.

Tavafoghi, Hamidreza, & Teneketzis, Demosthenis. 2017. Informational incentives for congestion games. *Pages 1285–1292 of: 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE.

Tumer, Kagan, & Wolpert, David. 2000. Collective Intelligence and Braess' Paradox. *AAAI*, 104–109.

Van Segbroeck, Sven, de Jong, Steven, Nowe, Ann, Santos, Francisco C., & Lenaerts, Tom. 2010. Learning to coordinate in complex networks. *Adaptive Behavior*, **18**(5), 416–427.

Viedma, José María. 1999. Building a Network Theory of Social Capital. *Connections*, **22**(1), 28–51.

Vinet, Luc, & Zhedanov, Alexei. 2011. A 'missing' family of classical orthogonal polynomials. *Journal of Physics A: Mathematical and Theoretical*, **44**(8), 1–8.

Wardrop, John Glen. 1952. Some theoretical aspects of road traffic research. *Inst Civil Engineers Proc*, **Part II**(1), 325–378.

Watkins, Christopher John Cornish Hellaby. 1989. *Learning from Delayed Rewards*. Ph.D. thesis.

White, Neil, Rota, G-C, & White, Neil M. 1986. *Theory of matroids*. Cambridge University Press.

Wiering, Marco. 2000. Multi-agent reinforcement learning for traffic signal control. *Machine Learning: Proceedings of the Seventeenth International Conference (ICML'2000)*, 1151–1158.

Wu, Yuhu, Cheng, Daizhan, Ghosh, Bijoy K., & Shen, Tielong. 2019. Recent advances in optimization and game theoretic control for networked systems. *Asian Journal of Control*, **21**(6), 2493–2512.

Yan, Xiaoyun, Dong, Ping, Du, Xiaojiang, Zheng, Tao, Zhang, Hongke, & Guizani, Mohsen. 2018. Congestion game with link failures for network selection in high-speed vehicular networks. *IEEE Access*, **6**, 76165–76175.

Yao, Jia, Cheng, Zhanhong, Dai, Jingtong, Chen, Anthony, & An, Shi. 2019. Traffic assignment paradox incorporating congestion and stochastic perceived error simultaneously. *Transportmetrica A: Transport Science*, **15**(2), 307–325.

Yu, Hao, Ma, Rui, & Zhang, H. Michael. 2018. Optimal traffic signal control under dynamic user equilibrium and link constraints in a general network. *Transportation Research Part B: Methodological*, **110**, 302–325.

Zhao, Chunxue, Fu, Baibai, & Wang, Tianming. 2014. Braess paradox and robustness of traffic networks under stochastic user equilibrium. *Transportation Research Part E*, **61**, 135–141.