Coordination through Local Communication



Wei-Chen Lee

Reuben College

University of Oxford

A thesis submitted for the degree of

DPhil in Computer Science

Michaelmas term, 2025

Supervised by Alessandro Abate and Michael Wooldridge

Abstract

Communication is a natural, practical and effective way for players in a game to coordinate their actions. In this thesis, I explore a particular form of localised communication where players are partitioned into coalitions, and can reach an agreement about their coordinated actions if the agreement is credible, efficient, and fair. I term the correlated strategies induced by such a communication partition 'partition equilibria', and study them in the context of singleton congestion games, where players have the incentive to disclose their strategy in order to avoid congestion. In the absence of communication, such games can lead to highly inefficient outcomes due to coordination failure.

Part I of this thesis studies the agreements that can be reached by a coalition of players under a range of credibility and fairness assumptions. I provide a full landscape of the computational complexity of finding an optimal agreement, and show that, in contrast with the unconstrained optimisation problem, many of the fairness assumptions impose a particular structure to the solution, termed 'contiguity', that enables a solution to be found in polynomial time.

Part II of this thesis characterises the 'optimal partition' - the partition that induces a partition equilibrium with the best worst-case outcome, for the cases of weighted and unweighted players. I also show that, for the general case of weighted players, using a simple partition can induce a worst-case outcome that is a 2-approximation to the socially optimal outcome.

I use a range of approaches to model player behaviour, including an axiomatic approach to credibility, efficiency, and fairness, and the use of epistemic uncertainty to model player beliefs. I show that, in contrast with the Nash equilibrium, such approaches provide an interpretation to the equilibrium selection problem. Such a framework can in turn be used to detect latent communication between players.

Acknowledgements

I am grateful to many people who have supported me over the past four years.

To my supervisors, Alessandro Abate and Michael Wooldridge, thank you for the opportunity to pursue this research, and for the guidance and support you have provided along the way. I would also like to thank Edith Elkind and Martin Bullinger, who have been so generous with their time, energy, and insights, and from whom I have learnt a great deal. I am grateful to David Hyland, with whom I wrote my first research paper, for the best introduction to collaborative research I could hope for. My thanks also go to Paul Goldberg, Elias Koutsoupias, and many other colleagues in the Computer Science Department and the Economics Department, who have provided helpful comments on my work. To my past and present colleagues at the University of Oxford - including Lewis Hammond, Anna Gautier, James Fox, Jiarui Gan, Paul Harrenstein, Daniel Jarne Ornia, Nick Bishop, Joel Dyer, Ani Calinescu, and Stella Simic - thank you for being a part of this journey over the past four years and for all the stimulating conversations we have had. I will remember this time fondly.

I am also grateful to the Ministry of Education of Taiwan (Oxford-Taiwan Graduate Scholarship) and to Google DeepMind (Oxford-DeepMind Graduate Scholarship) for their generous funding; this research would not have been possible without their support.

Finally, I am deeply thankful to my parents, Chih-I Lee and Li-Yun Lee, for their perpetual support in all my endeavours, however haphazard they appear; and to my partner, Victoria Iungerich, for her wonderful company.

Contents

1 Introduction **1**

1.1 The room-sharing dilemma 2

1.2 Key concepts 4

1.3 Overview 6

1.4 Publications 6

2 Background and Related Literature **8**

2.1 Fundamentals 8

2.1.1 Strategic-form games 8

2.1.2 Nash equilibrium 10

2.1.3 Multiplicity, refinements, selection 11

2.1.4 Correlated equilibrium 13

2.2 Cheap-talk Communication 14

2.2.1 Alternative uses of cheap talk 16

2.2.2 Communication partition 16

2.3 Epistemic Game Theory 17

2.4 Symmetries in Games 18

2.5 Principle of Indifference 20

2.6 Strategic Scheduling 21

2.6.1 Scheduling 21

2.6.2 Multi-way number partitioning 22

2.6.3 Congestion games 22

2.6.4 Selfish load balancing 24

3 Fair Coordination **26**

3.1 Introduction 26

3.2 Notation 27

3.3 Problem Definition 28

3.4 Properties and Relations 29

3.5 Contiguous Agreement 33

4 Complexity of Fair Coordination **35**

4.1 Introduction 35

4.2 Tractable Properties 36

4.2.1 Simple Satisfiability 38

4.2.2 Envy-Freeness 40

4.2.3 Weak Ordered Envy-Freeness 44

4.2.4 Strong Monotonicity 55

4.2.5 $WOE \wedge SM$ 63

4.2.6 Weak Monotonicity 63

4.3 Intractable Properties 64

4.4 Conclusion 67

5 Communication Partition **68**

5.1 Singleton Congestion Games 69

5.2 Social Cost 70

5.3 Player Partition and Communication 72

5.3.1 Inter-coalition beliefs 72

5.3.2 Intra-coalition agreements 75

5.3.3 Partition equilibrium 78

5.4 Worst-case Analysis 80

6 Optimal Partition **82**

6.1 Unweighted Players 82

6.1.1 Optimal outcomes 83

6.1.2 Optimal partition 84

6.1.3 Differential cost functions 87

6.2 Weighted Players 88

6.2.1 Existence and uniqueness of partition equilibrium 88

6.2.2 Optimal partition 90

6.2.3 Approximation 92

7 Conclusion **95**

7.1 Summary 95

7.2 Future Directions 96

References **98**

1 Introduction

Communication is a natural, practical and effective way for players in a game to coordinate their actions, but it also raises questions about the credibility or the fairness of the coordinated agreement. In this thesis, I explore a particular form of local communication, where players are partitioned into coalitions and can coordinate their actions with other players of the same coalition, while remaining uncertain about the behaviour of other coalitions. The resulting correlated strategy profile, termed the 'partition equilibrium', provides a lens for reasoning about the range of outcomes that can arise under such localised communication.

I apply the study of partition equilibrium to a class of congestion games known as the Singleton Congestion Games (SCGs). SCGs can be thought of as a strategic version of the identical machine scheduling (IMS) problem: tasks of various completion times can be assigned to one of many identical machines, with the objective of minimising the makespan (the time taken to complete all tasks) across all machines. While the IMS is a single-agent optimisation problem, the SCG is a strategic interaction where each task is owned by a different player, and each player is only concerned about the completion time of the specific machine to which its task is assigned. In this context, this thesis is a study of how local communication can improve the social efficiency of a strategic resource allocation problem.

In games with multiple pure-strategy Nash equilibria (such as a typical SCG), it is often not clear how players can achieve with certainty any of the pure-strategy equilibria: If players do not communicate before choosing their actions, then they risk mis-coordination; If players do communicate, then they may reach some mutually satisfactory agreement, if such an agreement exists. By modelling the players' strategies as the result of some localised communication, we can provide a descriptive explanation to certain observed outcomes, and further design communication partitions that induce certain outcomes.

1.1 The room-sharing dilemma

Consider the following problem. Adam, Bob, and Charlie are colleagues on a business trip. They booked two identical hotel rooms, each with a large bed, so that two of them would need to share a room. Each person prefers not to share a room, and chooses one of the two rooms, 101 and 102.

One natural assumption is that all the colleagues might discuss among themselves which rooms to choose. However, since each colleague prefers not to share, they might reach a deadlock in their discussion: any allocation where two people share a room while one person does not would lead to 'envy', where one of the sharing colleagues would prefer to swap their room allocation with the colleague that is not sharing.

There is, however, one envy-free agreement, where all colleagues share the same room! While this ensures that no colleague envies the allocated room of the others, this clearly comes with its own problems.

The first is that this is not a 'credible' agreement: if the agreement is not binding, then each of the three colleagues would have an incentive to switch their choice at the last minute to the empty room. Thus, for an agreement to be credible, it must be 'self-enforcing' in the sense that no colleague would benefit from a deviation. In other words, the agreement must be a Nash equilibrium.

The second is that, aside from being non-credible, this agreement is also not Pareto-optimal, meaning that all the players can reach an agreement where everyone

is better-off than the current one. Clearly, all colleagues would be better-off in any other agreement where there are at most two people in each room.

Therefore, it seems, that no agreement between the three colleagues can satisfy envy-freeness, credibility, and Pareto-optimality, when they can together discuss an allocation before making their choices.

Consider the alternative case where no colleague can communicate their intended choice between them. How should each colleague make their choice, given their lack of knowledge about the choices of others? Let us first consider how Adam might approach this problem. Adam would like to choose the room that is least likely to be chosen by Bob and Charlie, but because the two rooms are identical, Adam cannot reason why one room would be preferred by Bob or Charlie over the other, and thus he assigns equal probability to each room being chosen by Bob and by Charlie. These probability assignments reflect Adam's epistemic uncertainty about the choices of others, rather than any explicit aleatoric uncertainty inherent in Bob and Charlie's strategies (i.e., Adam does not necessarily assume that Bob and Charlie randomise over which room to choose). Given these uncertainties, Adam is indifferent between the two rooms, since each offers the same probability of not sharing the room.

Whichever room Adam chooses (and regardless of whether he chooses to use a coin flip to help him decide), Bob and Charlie each face the same epistemic uncertainty about the choice of Adam. From the perspective of an outsider, with the full epistemic uncertainty about the choice of each colleague, one could reason that each of the $2^3 = 8$ distinct outcomes carry the same probability. Thus, while 6 out of the 8 outcomes are Pareto-efficient (involving two colleagues sharing a room and the third colleague in the other room), the 2 outcomes where all colleagues choose the same room can also occur with positive probability.

Notice that the outsider's view is consistent with the only mixed-strategy Nash equilibrium of the game, where each colleague chooses each room with probability 1/2. In fact, this is the only equilibrium that can be achieved without communication, since the colleagues have no means of coordinating on any of the pure-strategy

Nash equilibria, which are also Pareto-efficient. The case where the colleagues do not communicate with each other is clearly sub-optimal.

Now, consider a third case where only Adam and Bob can communicate with each other about their intended choice. What agreement should they reach? Since neither knows what Charlie will choose, neither has a preference over either room. They essentially face two choices: choosing the same room, or choosing different rooms. The first guarantees that both will end up sharing the room (with at least one other person), while the second ensures that each player has some chance of not sharing a room. So naturally Adam and Bob would agree to choose different rooms. Notice that such an agreement is credible (neither would want to deviate), Pareto-optimal (no other agreement can improve the outcome for at least one colleague without making another worse-off), and envy-free (neither Adam nor Bob would prefer to swap their assigned room with the other).

Furthermore, regardless of the choice of Charlie, a Pareto-efficient outcome is achieved where no room is unused! In fact, it is now guaranteed that the realised outcome is one of the pure-strategy Nash equilibria where Charlie shares a room either with Adam or with Bob. From the perspective of the outsider, their epistemic uncertainty about the choices of each individual remain unchanged; however, the choices of Adam and Bob are now perfectly correlated: if Adam chooses room 101, then Bob would choose room 102, and vice versa. The outsider would therefore perceive the induced strategy profile as a probability distribution over four pure-strategy Nash equilibria, each with probability $\frac{1}{4}$.

1.2 Key concepts

This example illustrates that local communication, in this case between Adam and Bob, can lead to more efficient outcomes than no communication, and avoid the deadlock of full communication. This local communication can be characterised by a communication partition, where only players within the same coalition can coordinate on a joint action. This example highlights a number of crucial aspects of such coordination:

- **Credibility**: Since any agreement reached among players within a coalition is not binding, the agreement must be credible for it to be followed through by the players. In this thesis, I take credibility to mean that no single player can benefit from a unilateral deviation from the agreement.

- **Efficiency**: While players are self-interested, they are not malicious towards each other. As such, players within a coalition should favour agreements on the Pareto-frontier among agreements that satisfy other conditions.

- **Fairness**: Players within a coalition must perceive the agreement to be 'fair' or else they may not agree to it. Fairness can be defined as satisfying the 'envy-free' condition in the above example; I also explore other notions of fairness in this thesis.

- **Epistemic uncertainty**: Players model what others outside of their coalition would do as a probability distribution over the range of possible choices. This distribution does not necessarily represent the randomised choices of others; rather, it reflects the player's epistemic uncertainty over the choices of others.

- **Symmetry and the principle of indifference**: Throughout this thesis, we consider the case where all resources that can be chosen by the players are identical, or that players have no means of distinguishing between resources that are different. In the example of Adam, Bob and Charlie, we could apply the same reasoning to each colleague even if the two rooms are inherently different, but no player can distinguish between them. This symmetry allows us to deploy the principle of indifference (also known as the principle of insufficient reason), which says that 'in the absence of any relevant evidence, a rational agent will distribute their credence (or 'degrees of belief') equally amongst all the possible outcomes under consideration'.

1.3 Overview

The remainder of this thesis is structured as follows. In Chapter 2 I discuss the economic and computer science literature upon which this thesis was developed, including each of the key concepts that we saw in Section 1.2.

I then divide the thesis into two parts, each consisting of two chapters. Part I considers the case of 'global communication' between the players. It discusses how a set of players can reach a credible, efficient, and fair agreement, assuming that they can communicate freely with each other.

Chapter 3 sets out the various axiomatic properties that are desirable or essential for an agreement to be reached, and their relations with each other. It also introduces the basic scheduling problem and its makespan minimisation objective, as well as the concept of 'contiguity' for an agreement. This is followed by Chapter 4 which answers the research questions posed in Chapter 3, by proposing algorithms which answer these questions, or by proposing reductions from well-known problems.

Part II discusses the case of 'local communication', where players are partitioned into coalitions and can only communicate with others within the same coalition. I ask what is the optimal partitioning of players that results in the most efficient worst-case outcome. Chapter 5 introduces the concepts of a communication partition, the partition equilibrium, the worst-case outcome, and the optimality of an induced partition equilibrium. It also introduces the Singleton Congestion Game (SCG), a generalisation of the strategic scheduling problem. Chapter 6 then studies the optimal partition with respect to SCGs with unweighted and weighted players.

I conclude in Chapter 7 with a review of the thesis, and set out directions for further research.

1.4 Publications

This thesis is based on the following conference papers. The content of Chapter 3 and Chapter 4 are primarily based on [1], while the content of Chapter 5 and Chapter 6 are primarily based on [2, 3].

- [1]: Wei-Chen Lee, Martin Bullinger, Alessandro Abate, and Michael Wooldridge. 2026. To appear in Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026).
- [2]: Wei-Chen Lee, Alessandro Abate, and Michael Wooldridge. 2025. Equilibrium Selection via Cheap-talk Partition: Extended Abstract. In Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025).
- [3]: Wei-Chen Lee, Alessandro Abate, and Michael Wooldridge. 2025. Partition Equilibria in Weighted Singleton Congestion Games. In Proc. of the 28th European Conference on Artificial Intelligence (ECAI 2025).

2

Background and Related Literature

In this chapter, I discuss the relevant concepts in game theory upon which this thesis is built. This includes the strategic-form games and the solution concepts of the Nash equilibrium, its refinements and selection criteria, and the correlated equilibrium. I then discuss cheap-talk communication, a means through which players can exchange their strategic intentions, and epistemic game theory, which models how players handle epistemic uncertainty. Finally, I discuss the literature on congestion games and the scheduling literature, which are the core applications of this thesis.

2.1 Fundamentals

2.1.1 Strategic-form games

A *strategic-form game* is a common representation of strategic interactions in *non-cooperative* game theory (e.g., [4, 5]).[1] Such a game is defined by a tuple $(N, (S_i)_{i\in N}, (u_i)_{i\in N})$ where N is a set of n players, typically represented by $[n] = \{1, \ldots, n\}$; S_i represents the *strategy space* of player $i \in N$, and a *strategy profile* $\mathbf{s} \in S \coloneqq \prod_{i\in N} S_i$ is a tuple which specifies the strategies of all players. $u_i : \mathbf{s} \to \mathbb{R}$

[1] Non-cooperative games are games where players cannot make binding commitments. It contrasts with cooperative games, where players can form binding commitments, and the focus is on which coalitions will form among the players.

represents the utility function of player i, mapping each strategy profile $\mathbf{s} \in S$ to a real value representing the player's utility under $\mathbf{s}$.

A player can choose strategies that are deterministic ('pure'), or stochastic ('mixed'), i.e., a probability distribution over pure strategies. This distribution can come from (at least) two sources of uncertainty: aleatoric uncertainty and epistemic uncertainty. Aleatoric uncertainty relates to the deliberate randomisation of strategies by the player, perhaps using some randomisation device such as a die or a coin. In this case, even the player themself does not know the outcome of the randomisation at the time of playing the strategy.

Epistemic uncertainty comes from the uncertainty of an observer about the strategy chosen by the player, possibly based on the observer's subjective belief about the player. The player in question may choose a fixed strategy, but the observer can only express their epistemic uncertainty as a probability distribution. This crucial distinction will be further discussed in Section 2.3 and Chapter 5.

A standard interpretation of the strategic-form game is that all players simultaneously choose a strategy from their strategy space, and each derive some utility based on the strategy profile that is chosen. The games that we consider in this thesis can all be represented in this form.

However, the strategic-form representation is also capable of capturing player interactions that are naturally sequential (i.e., extensive-form games)[2], where the game unfolds over a 'game tree' and some players are required to take actions at each node (stage) of the game.

When representing an extensive-form game in strategic-form, a strategy specifies a player's chosen action under all contingencies, that is, it specifies what a player should do at each stage in the game where they are required to take an action. Thus strategic-form games are capable of representing a significantly wider set of games than its standard interpretation. The first explicit use of the strategic-form game representation was formulated in [6].

[2]This holds under the standard assumption of 'perfect recall' where players remember the actions they have taken at each step of the game.

2.1.2 Nash equilibrium

Nash equilibrium is perhaps the most widely used and the most important solution concept in game theory, and certainly in strategic-form games [7]. First introduced in [8], it is defined as a strategy profile from which no player has the incentive to unilaterally deviate. Formally, given a strategic-form game, $\mathbf{s}^* \in S$ is a Nash equilibrium if, for all $i \in N$ and $s'_i \in S_i$,

$$u_i((s^*_i, \mathbf{s}^*_{-i})) \geq u_i((s'_i, \mathbf{s}^*_{-i}))$$

where $\mathbf{s}^* = (s^*_i, \mathbf{s}^*_{-i})$ explicitly expresses the strategy of player i and that of all other players, denoted by $-i \coloneqq N \setminus \{i\}$. $(s'_i, \mathbf{s}^*_{-i})$ denotes the strategy profile where player i chooses s'_i while all other players follow their strategies in s^*.

Given the formal definition, another way to think about the Nash equilibrium is in terms of 'best responses'. Given the strategies of all other players $\mathbf{s}_{-i}$, a best response s^*_i for player i is one that ensures player i can do no better than to play s^*_i (though there may be multiple best responses), i.e., $u_i((s^*_i, \mathbf{s}^*_{-i})) \geq u_i(s_i, \mathbf{s}^*_{-i})$ for all $s_i \in S_i$. Viewed in this light, a Nash equilibrium is a strategy profile where each player is best-responding to the strategies of all other players.

A subclass of Nash equilibrium where all players choose pure strategies is a pure-strategy Nash equilibrium (PSNE), while the general version, where players can choose mixed strategies, is known as the mixed-strategy Nash equilibrium (MSNE). Nash [8] established the important result that every finite (finite players, finite strategy spaces) non-cooperative game has at least one (mixed-strategy) Nash equilibrium, through the use of Kakutani's Fixed-Point Theorem.

MSNE are inherently unstable, requiring some players to be exactly indifferent about a number of alternatives. As we will see in the next section, this makes MSNE a very 'fragile' solution concept that can be easily eliminated under stricter conditions.

The computational complexity of finding a MSNE is $PPAD$-complete [9], which is a class of hard problems (believed to be not solvable in polynomial time) where the solution is guaranteed to exist.

2.1.3 Multiplicity, refinements, selection

A common feature of the Nash equilibrium is multiplicity, that is, a given game may admit multiple Nash equilibria. For example, consider a simple two-player game, where the two players would like to meet in town. Both players prefer to meet than not to meet, but each has a mild preference over the locations. The locations where they can meet include a cafe, a pub, two restaurants, a park, and the town square. There are six PSNE in this game (where both players choose the same location), and the solution concept does not provide any prediction or prescription on where they should meet.

This multiplicity gave rise to a large number of proposals for finding a subclass of Nash equilibrium with reasonable properties ('equilibrium refinement') or developing some criteria for selecting among a multitude of equilibria ('equilibrium selection').

In the refinement literature, a number of stricter requirements were proposed to eliminate Nash equilibria that are considered 'weak'. In games consisting of multiple stages (i.e., have a non-trivial extensive-form representation), *Subgame Perfect Equilibrium* [10] eliminates Nash equilibria that are not credible threats, by requiring that the strategy profile is a Nash equilibrium in every sub-tree of the game tree ('subgame'). *Trembling Hand Perfect Equilibrium* [11] eliminates Nash equilibria that consist of weakly dominated strategies, by assuming that players have a small chance of playing an unintended strategy. In games with incomplete information, players holds beliefs about the behaviour of others, *Perfect Bayesian Equilibrium* (see [12]) requires strategies to be optimal given such beliefs, and that beliefs are updated via Bayes rule along the equilibrium path; *Sequential Equilibrium* [13] further requires beliefs to be consistent even off the equilibrium path. These concepts are designed to eliminate beliefs that are inconsistent with the chosen strategy.

In the selection literature, the work of Harsanyi and Selten [14] attempted to resolve the equilibrium selection problem by considering payoff dominance and risk dominance. A payoff-dominant equilibrium is a Nash equilibrium that is strictly preferred by all players to any other Nash equilibrium. Formally, let $\varepsilon \subseteq S$ be the set of Nash equilibria of a game. Then $\mathbf{s}^* \in \varepsilon$ is payoff-dominant if $u_i(\mathbf{s}^*) > u_i(\mathbf{s})$ for

all $i \in N$, $\mathbf{s} \in \varepsilon$. The idea is that if there is a Nash equilibrium that is preferred by everyone, then it would be rational for all players to play the corresponding strategy.

A risk-dominant equilibrium is a Nash equilibrium that has the largest 'basin of attraction', which means that it is the best response for each player under the widest range of beliefs about the behaviour of other players. The two concepts are often illustrated using the Stag Hunt game, shown in Figure 2.1(c).[3]

This game has three NE: two PSNE, (S, S) and (H, H), and a MSNE. (S, S) is clearly payoff-dominant as it offers (strictly) the highest payoff for all players. However, (H, H) is the risk-dominant equilibrium: it is the optimal choice if each hunter assumes that the other will hunt stag with probability less than 1/3, and thus offers a 'safer' bet in the presence of uncertainty. Notably, the MSNE sits exactly at this boundary, where players are indifferent between the two choices if they believe that the other would hunt stag with probability exactly 1/3 (and for this reason the MSNE has a minimal basin of attraction, occupying a single point in the belief space).

Figure 2.1: Payoff profiles of well-known games

	A	B
A	1, 1	0, 0
B	0, 0	1, 1

(a) Pure Coordination

	B	S
B	2, 1	0, 0
S	0, 0	1, 2

(b) Battle of the Sexes

	S	H
S	3, 3	0, 2
H	2, 0	2, 2

(c) Stag Hunt

Unfortunately, neither payoff dominance nor risk dominance guarantees existence and uniqueness. In particular, consider the simple coordination game in Figure 2.1(a), where each of the two players has two symmetric strategies A and B. This game has two PSNE, (A, A) and (B, B), and a MSNE where players choose each strategy with probability 1/2. The two PSNE payoff-dominate the MSNE, but since they offer equal payoffs to each player, they do not payoff-dominate each other, and thus the game has no payoff-dominant Nash equilibrium. The two PSNE

[3]The narrative of the game is as follows: there are two hunters, and each can choose to hunt stag (S) or hare (H). Stag is valuable but hard to catch, and can only be caught if both hunters choose to hunt stag; otherwise the hunter choosing stag leaves empty-handed. Hare is easy to catch, and offers a lesser reward if chosen by a hunter, regardless of the choice of the other.

also have the exact same basin of attraction, and thus neither risk-dominates the other, and the game has no risk-dominant Nash equilibrium.

Interestingly, while in [14] Harsanyi and Selten found both considerations relevant for equilibrium selection, Harsanyi [15] later changed his mind based on Aumann's argument [16] and considered risk dominance to be the only relevant criterion. Aumann's argument is presented in Section 2.2 in the context of cheap-talk communication.

2.1.4 Correlated equilibrium

Correlated equilibrium (CE) [17, 18] is a generalisation of NE which allows players' strategies to be correlated. This contrasts with the standard definition of a strategy profile $\mathbf{s} \in \prod_{i \in N} S_i$ where players strategies are independent from one another.

Specifically, a correlated strategy σ is a probability distribution over all outcomes S, i.e., $\sigma \in \Delta S$, where ΔS represents the probability simplex over S, and $\sigma(\mathbf{s}) \in [0, 1]$ is the probability of $\mathbf{s}$ under σ.

A correlated equilibrium σ^* is a correlated strategy where a correlation device ('mediator') chooses some strategy $\mathbf{s}^* \in S$ according to σ^*, privately sends each player i their recommended strategy s_i^*, and no player has the incentive to deviate from the recommendation. Formally, for each $i \in N$ and $s_i \in S$,

$$\sum_{\mathbf{s}_{-i} \in S_{-i}} \sigma^*(\mathbf{s}_{-i}|s_i^*)[u_i((s_i^*, \mathbf{s}_{-i})) - u_i((s_i, \mathbf{s}_{-i}))] \geq 0$$

The definition assumes that the probability distribution is common knowledge (i.e., known to all players, and known by all players that it is known to all players, etc.) and that the mediator can be trusted to truthfully deliver the recommendations accordingly.[4]

This requirement of a trusted mediator can seem strong for the implementation of CE. As I will cover in Section 2.2, a large number of papers in the cheap-talk

[4]Aumann argues that the 'trusted mediator' is simply a mathematical construct to illustrate the idea, and in practice a CE can be implemented by players observing some random event in the world with a known probability distribution, and having a common prior about the interpretation of such signals.

literature that follow show that the mediator can (when $|N| \geq 4$) be replaced by non-binding pre-play communication among the players.

Note that every NE is a CE: If a mediator recommends a NE to all players, then by the definition of a NE, each player has the incentive to follow the recommendation. It follows that a CE always exists.

On the multiplicity of CE, Aumann [17] showed that the set of CE is a convex polytope. This has two important implications. One, the CE selection problem becomes even more challenging over a continuous space, which is why the trusted mediator is required to implement an arbitrary CE. Two, this makes the finding of a CE computationally easy, i.e., polynomial-time solvable using linear programming [19].

2.2 Cheap-talk Communication

While Nash [20] defined a non-cooperative game as one where each player 'acts independently, without collaboration or communication with any of the others', a natural response to the equilibrium selection problem is communication. This is how we as humans resolve many of our coordination problems: when do we meet, where do we meet, what do we eat, etc.. It is thus not surprising that pre-game communication, where players communicate their non-binding intentions with one another, is heavily studied in the game theory literature.

Farrell and Rabin [21, 22] study cheap-talk for coordination in complete information games with multiple Nash equilibria. These papers discuss the crucial concept of 'self-committing' signals: if a player declares that they intend to act according to one NE, and if other players believe the signal and best-respond to it, then the first player is incentivised to follow through with their declared intention, since doing so is a best response to others' responses. This is a crucial concept in the cheap-talk literature, and from which the concept of 'credible' agreement is derived in this thesis.

Nonetheless, debate on the credibility of cheap-talk communication persists in the literature. Consider again the stag hunt game shown in Figure 2.1(c). We have discussed that despite (S, S) being the payoff-dominant NE, players may choose to

play (H, H), the risk-dominant NE, if they were uncertain about the behaviour of the other player. Does cheap-talk communication alleviate such uncertainty?

Farrell and Rabin [22] note that the signal to play S is self-committing: if it is believed by the receiver (the other hunter), then the best response of the receiver is to play S, and thus the best response of the signal sender is to also play S, in accordance with the signal. They argue that the signal should act as a reassurance about the intention of the sender, and therefore reduce the uncertainty in the game and encourage the payoff-dominant NE.

Aumann [16] on the other hand argues that the cheap-talk signal should have no effect on the outcome of the game: regardless of whether the signaller intends to play S or H, signalling S is a dominant strategy, and therefore contains no information.[5]

Experimental evidence shows some nuanced results. In [23] the authors found that under no communication, players often converge to the risk-dominant equilibrium (H, H); under one-way communication, players are more likely to converge on the payoff-dominant equilibrium (S, S); under two-way communication, players mostly coordinate successfully on the (S, S) outcome. This work thus lends support to the claim that cheap-talk can act as an assurance and encourage players to coordinate, but also that it is most effective in two-way communication.

These results were corroborated by Charness [24] using the payoffs specified by Aumann, where there is a small benefit to hunting hare alone than in a pair. Interestingly, the experiment also shows that if a signal sender were to fix their action first before sending a signal (as suggested in [21]), then the receiver is more likely to be sceptical of the signal, and result in a similar level of coordination to the case of no communication, as Aumann conjectured. This suggests that when the known order of signal-action is reversed (which has, in theory, no impact on the players' strategies), the self-interest problem noted by Aumann becomes more prominent in the mind of the receiver. Clark et al. [25] found that while communication is effective in encouraging the payoff-dominant outcome, its efficacy is sensitive to the payoff structure of the game, and can fail when there is some benefit to hunting hare alone.

[5] In Aumann's model, there is a slight benefit to hunting hare alone than if both hunters hunt hare, and thus a hare hunter would strictly prefer the other hunter to hunt stag.

2.2.1 Alternative uses of cheap talk

In addition to being a coordination device, cheap-talk has also been studied in games where players have incomplete information about the game, and can share their private information through cheap-talk communication. In the seminal work of Crawford and Sobel [26] involving an informed sender and an uninformed receiver, where the action of the receiver affects both players, information transmission can occur in equilibrium.

There is also a rich literature exploring how a correlated equilibrium (or related concepts) can be implemented in a cheap-talk setting where all players can communicate bilaterally and also broadcast messages to all others (e.g., [27–34]). The key idea of this literature is to describe a communication scheme where any deviation from the prescribed strategy can be detected and credibly punished. The capacity for all players to communicate with each other is required so that deviations are publicly announced and trigger the punishment strategy.

Crucially, this literature does not address the equilibrium selection problem: which of the multitude of equilibria should be prescribed to the players? This literature also focuses on a specific communication network where all players can communicate bilaterally with each other, and broadcast to all players, which is distinct from this thesis, where I study the use of communication partitions as a mechanism for inducing particular outcomes.

More recently, Jamison [35] proposes a model describing how cheap-talk between players update their beliefs about the eventual action of each player, which converges to an efficient equilibrium.

2.2.2 Communication partition

Another idea presented in this thesis is localised communication, where players are partitioned into mutually exclusive and collectively exhaustive subsets ('coalitions') and can only communicate with those within the same coalition.

The idea of representing the communication network between players with a partition was first introduced in [36]. This work focuses on cooperative games with

transferable utility, and the role of communication as a means of negotiating for shares of the value of the coalition. In contrast to [36], this thesis relates to non-cooperative games with non-transferable utility, where the role of communication is to coordinate actions.

A related piece of work [37] studies the efficiency of a congestion game of coalitional players, where a coalition can act as a single player to minimise the maximum cost incurred by any player inside the coalition. This differs from the assumptions in this thesis where each player within a coalition is self-interested, and that any agreement reached by players within a coalition is non-binding.

2.3 Epistemic Game Theory

So far, I have discussed the use of cheap-talk for players within the same coalition to coordinate their strategies. This leaves open the question of how the players in each coalition should assess the strategies of players in other coalitions.

Harsanyi [38] first introduced the concept of player 'types' to capture a player's belief. A player can be one of multiple types, each with a belief which places a probability distribution on the types of all other players. The use of this type space neatly captures the infinite order of belief hierarchy underlying this problem.[6] Harsanyi's idea was formalised by Mertens and Zamir [39] into the *universal type space*, which is the space of all possible beliefs a rational player could have.

My treatment is based on another approach proposed by Aumann and Brandenburger [18, 40], which models a player's belief about the state of the world. This not only contains beliefs about the beliefs of other players, but also beliefs of other players' chosen strategies. This framework makes explicit the epistemic uncertainty that a player has about the chosen strategies of others, by capturing it within the true state of the world.

[6] Harsanyi's type also has a second component which is the type's private information. This aspect is not relevant to games of complete information and not further discussed here.

2.4 Symmetries in Games

Given the epistemic framework proposed by Aumann and Brandenburger, what belief should a player hold? This is a difficult question with no obvious solution. However, in the case where certain symmetry exists, we can make reasonable assumptions about such beliefs.

There are several distinct notions of symmetry in strategic-form games: player symmetry, strategy symmetry, and game automorphism. Here, we follow the framework of classifying different symmetries in [14]. They consider a tuple of permutations $\theta = (\pi, \phi_1, \ldots, \phi_n)$ where $\pi : N \to N$ is a bijection from the set of players to itself, and $\phi_i : S_i \to S_i$ is a bijection from the strategy set of player i to itself. A symmetry refers to the invariance of players' payoffs under certain permutations.

Player symmetry

Player symmetry refers to the case where the payoff structure remains unchanged under any permutation of the players. Formally, a game is player-symmetric if for any permutation π of players,

$$u_{\pi(i)}(s_1, \ldots, s_n) = u_i(s_{\pi(1)}, \ldots, s_{\pi(n)}).$$

Because the game requires payoff invariance under all permutations of players, this is sometimes described as being 'totally symmetric', in contrast with games where the payoff invariance holds only for some permutations (e.g., a cyclic permutation). We are concerned only with total symmetry in this thesis, which we simply refer to as player symmetry.

This notion was first introduced in [41] in the context of cooperative games, where the value of a coalition depends only on the number of players it contains, not the identities of the players. The definition of player symmetry in non-cooperative games was introduced in [20], where such games are simply referred to as *symmetric*

games. Nash proved that every finite symmetric game has at least one *symmetric Nash equilibrium* where every player plays the same (possibly mixed) strategy.

Anonymity and identical utility

A related but weaker concept to player symmetry is anonymity (e.g., [42–44]), where a player's utility does not depend on the identity of other players, but only the player's own strategy and the aggregate distribution of others' strategies.

A player-symmetric game is always anonymous, since if a player's payoff remains unchanged under all permutations of all other players, then it does not depend on the identity of the other players. An anonymous game may not be player-symmetric, since players in an anonymous game may have different utility functions. A game is player-symmetric if it is anonymous and that each player has identical utility function [45].

One class of game studied in this thesis, the unweighted singleton congestion game, is an anonymous game.[7] Milchtaich [42] showed that a PSNE always exists in such games by proving that there always exists a finite best-response path (see further discussion in Section 2.6.3).[8]

Strategy symmetry

Strategy symmetry refers to the invariance of a player's payoff under some permutation $(\phi_1, \ldots, \phi_n)$ of the strategy spaces of players. In this thesis, we are concerned with the special case where all players have the same strategy space, $S_i = S_1$, and that a single permutation ϕ is applied to all strategy spaces. A game is (totally) strategy-symmetric if for all permutations of the strategy space ϕ and for all $i \in N$,

$$u_i(s_1, \ldots, s_n) = u_i(\phi(s_1), \ldots, \phi(s_n)).$$

In other words, relabelling the strategies that are available to each player has no impact on the payoff of the player. For example, in congestion game settings (see

[7] Since we assume that players have identical utility function, the unweighted singleton congestion game we study here is also player-symmetric.

[8] A best-response path is a dynamic where, initiating from a strategy profile, players are offered, in turn, to change their strategy to the best-response strategy. If such a path terminates, then a PSNE is reached where all players are best-responding.

Section 2.6.3) where resources are identical, the sole determinant of a player's cost is the total number of players choosing that same resource, and does not depend on the identity of that resource.

Game automorphism

Game automorphism combines both player symmetry and strategy symmetry. Consider the tuple $\theta = (\pi, \phi_1, \ldots, \phi_n)$, where $\pi : N \to N$ is a permutation on the set of players, and $\phi_i : S_i \to S_{\pi(i)}$ is a bijection from the strategy set of player i to the strategy set of player $\pi(i)$. The tuple is termed a *game automorphism* if it preserves the payoff structure such that, for all $i \in N$ and for all $s_i \in S_i$,

$$u_i(s_1, \ldots, s_n) = u_{\pi(i)}(\phi_1(s_1), \ldots, \phi_n(s_n))$$

Let id be the identity function. Player symmetry can thus be seen as the special case where the payoff structure remains unchanged under the game automorphism $\theta = (\pi, id, \cdots, id)$ for all permutations $\pi : N \to N$. Similarly, strategy symmetry under identical strategy space is the special case where the payoff structure remains unchanged under the game automorphism $\theta = (id, \phi, \cdots, \phi)$ for all permutations $\phi : S_1 \to S_1$.

2.5 Principle of Indifference

There are two related but distinct concepts known as the principle of indifference.

The first is a fundamental concept in game theory and the rational basis for playing a mixed strategy: for a rational player to play a mixed strategy, they must be indifferent between the pure strategies that are played with positive probability. Harsanyi [46] considers the idea that a player would actively randomise over pure strategies that they find equally appealing to be highly contentious, and offers an alternative interpretation where the probability distribution stems from an observer's uncertainty over the exact utility that the player derives from each strategy, and represents the epistemic uncertainty that the observer has about the pure strategy that is being played (see earlier discussion in Section 2.3).

The second is a prescriptive concept from probability theory, originally termed the 'principle of insufficient reason', first articulated by Bernoulli [47] and later Laplace [48]. It is the principle that in the absence of any information distinguishing one event from another, one should treat such events as equally probable. It was renamed the 'principle of indifference' by Keynes [49] to emphasise the epistemic (rather than aleatoric) nature of such uncertainty. The principle was generalised into the 'principle of maximum entropy' [50] in the fields of information theory and physics, which states that the probability distribution that best represents the current state of knowledge is one with the largest entropy.

I adopt both concepts in this thesis. When a player (or an outside observer) places a probability distribution over the behaviour of players outside of their coalition, they do so due to their epistemic uncertainty over which set of strategies have been agreed by players in other coalitions; and, since the games we consider are strategy-symmetric, a player places equal probability over all such strategies.

2.6 Strategic Scheduling

The application that I consider in this thesis can be generally described as 'strategic scheduling'. The 'strategic' component highlights the tension between private interests and social outcomes, while the 'scheduling' component describes the type of problem that we are faced with. It connects a number of highly related (and in specific cases mathematically equivalent) problems in different fields of research.

2.6.1 Scheduling

Scheduling is a problem in operational research and computer science that concerns the allocation of tasks to different machines, with the typical objective of minimising the time required to complete all tasks.

In its most general form, the time taken to process each task is specified for each machine, which enables the modelling of machines that are specialised for processing specific tasks. A simpler model may assume that tasks are qualitatively similar but quantitatively different, and can thus be represented by weights, such that a task

of weight 4, say, takes twice as long to process as a task of weight 2, for any given machine. The simplest model, termed identical machine scheduling (IMS), assumes that all machines are identical and has the same process speed. That is, given a task of some weight w, all machines will process it in w units of time.

Early works in the scheduling literature established that finding the optimal schedule for IMS is NP-complete [51], and that a polynomial-time approximation of $4/3 - 1/(3m)$ can be achieved for the minimax problem using the *Longest Processing Time First (LPT)* algorithm [52].

2.6.2 Multi-way number partitioning

A closely related problem in combinatorics is the multi-way number partitioning (MNP) problem. A problem instance is given by a multiset of numbers and a number of bins. The objective is to distribute the numbers to the bins 'as evenly as possible'. Partitioning the numbers 'as evenly as possible' typically means one of three distinct objectives (e.g., [53]):

- minimax: minimise the size of the largest bin;
- maximin: maximise the size of the smallest bin; or
- min-gap: minimise the gap between the largest and the smallest bin.

Notice that under the minimax objective, the MNP problem is equivalent to the IMS problem. [51] showed that finding the optimal distribution is NP-complete for each of the three objectives.

2.6.3 Congestion games

In game theory, congestion games are a general class of games involving players and resources. Each player has a strategy space, where a strategy involves the use of a subset of resources. The cost incurred by a player is the sum of the costs that the player incurs across each of the resources chosen under the player's strategy, and the cost of using a resource is increasing in the number of players choosing that resource. One can think of a player's strategy space as a variety of ways in

which the player can achieve their objective by utilising different combinations of resources; however, each resource is congestible, and becomes increasingly costly to all players using it as more players choose to use it.

A standard example is the network congestion game which models road congestion. The game can be represented by a graph, where the vertices represent junctions and the edges represent roads. Each player has a source vertex and a destination vertex, and each path that goes from the source vertex to the destination vertex is a strategy of that player. The edges in the game are congestible, meaning that the time taken to traverse it increases as more players choose strategies that use that edge. The cost to each player is the total time it takes the player to reach their destination.

Congestion games are a type of 'anti-coordination' game where players benefit from playing dissimilar strategies to that of other players, in contrast with coordination game where players benefit from choosing the same strategy as others.

An important development in the study of congestion games is its connection with potential games. The work of [54] established that every congestion is an exact potential game, and [55] later established the converse, that every game with an exact potential function is equivalent to some congestion game.

An exact potential game is a game where there exists a potential function $\Phi : S \to \mathbb{R}$ such that, $\forall i \in N$, $\forall s_i, s_i' \in S_i$, and $\forall \mathbf{s}_{-i} \in S_{-i}$,

$$\Phi((s_i, \mathbf{s}_{-i})) - \Phi((s_i', \mathbf{s}_{-i})) = u_i((s_i, \mathbf{s}_{-i})) - u_i((s_i', \mathbf{s}_{-i}))$$

In other words, the potential function neatly captures the change in utility when any player changes their choice of action.

A potential game where each player has a finite number of strategies has what is known as the 'finite improvement property'. Consider the following process: start with any pure strategy profile $\mathbf{s} \in S$, and let a single player change their (pure) strategy for the better. Since doing so strictly increases their utility, the potential Φ strictly increases. Repeat this process until no player can benefit from changing their strategy. Because there are finitely many strategy profiles, this process must

terminate, at which point no player can improve on their strategy and we have reached a PSNE. And thus every finite congestion game has a PSNE.

Another important area of study for the congestion game is the 'price of anarchy' (PoA) [56–61], which is the ratio between the social cost of the worst Nash equilibrium and the social cost of the optimal outcome. It is a worst-case measure of the loss of efficiency when moving from a system of centralised control to a decentralised system where each player acts in accordance with their private interest. In congestion games, the social cost is measured as the (weighted) sum of the players' costs, i.e., the utilitarian social welfare.

For linear cost functions, [57, 59] show that the PoA is exactly 2.5. [60] show that for general polynomial cost functions of degree d, the PoA is

$$\frac{(k+1)^{2d+1} - k^{d+1}(k+2)^d}{(k+1)^{d+1} - (k+2)^d + (k+1)^d - k^{d+1}},$$

where $k = \lfloor \Psi_d \rfloor$ and Ψ_d is a generalisation of the golden ratio to higher dimensions such that $(\Psi_d + 1)^d = \Psi_d^{d+1}$.

The generalisation of congestion games to weighted players was given in [42]. Weighted congestion games with linear or exponential cost functions are still guaranteed to possess PSNE [62–64], but this is not true of general cost functions. The PoA of weighted congestion games with linear cost is $\frac{3+\sqrt{5}}{2} \approx 2.62$, marginally worse than the unweighted case.

Non-atomic congestion games are a class of congestion games where the number of players $n \to \infty$, where each player's own choice has no impact on the congestion level. We do not consider such games in this thesis; interested reader may consult the works of [65–69] on the PoA of such games.

2.6.4 Selfish load balancing

Finally, if we restrict a weighted congestion game to the case where each player chooses just one resource (as opposed to a subset of all resources), then we have a weighted singleton congestion game. Specifically in the case where the cost function

is linear, and that the social cost is the highest load on any resource (as opposed to the utilitarian objective), then we have the selfish load balancing problem.

The survey paper [70] shows that for m identical resources, the PoA of PSNE in this problem is $2 - \frac{2}{m+1}$ based on the Longest Processing Time algorithm [52], and that the PoA is $\Theta(\frac{\log m}{\log \log m})$ for the MSNE.

3
Fair Coordination

3.1 Introduction

I begin by considering a strategic variant of a classic problem known as the identical machine scheduling (IMS) problem, where each of the n weighted tasks is assigned to one of the m identical resources (machines). The load on a resource is the sum of the weights of the tasks assigned to it, and the objective of the assignment is to minimize the largest load on any resource, known as the 'makespan' of the assignment.[1]

In this strategic variant, each task is owned by a different player concerned only with the completion time of the resource to which their task is assigned. I assume explicitly that players can coordinate on their choice of resource through pre-play communication, and I refer to an *agreement* as the tuple $\mathbf{a} = (a_1, ..., a_n)$ where $a_i \in [m]$ is the agreed choice of resource of player i. Notice that because resources are identical (in the sense that they process each task at the same speed as all other resources), the exact identity of the resource is not important.

What kind of agreement do we expect players to reach? I take an axiomatic approach and consider the properties that an agreement might satisfy, such as credibility, efficiency and fairness. In this chapter, I introduce the problem

[1]The interpretation is to consider the weight of a task as the time taken to process it, and thus the makespan is the time required to process all tasks.

definitions and the relations of properties that are of interest to us; the study of the computational complexity of solving these problems are deferred to the next chapter.

3.2 Notation

For positive integers k and ℓ, I denote $[k] := \{1, \ldots, k\}$ and $[k : \ell] := \{k, \ldots, \ell\}$. Given a vector $\mathbf{v}$, I denote by $|\mathbf{v}|$ its length. We have a set $[n]$ of weighted players, and a set $[m]$ of identical resources. The weights of players are represented by a vector $\mathbf{w} = (w_i)_{i \in [n]} \in \mathbb{Z}^n_{>0}$ of positive integers. I denote by $\text{supp}(\mathbf{w})$ the set of unique weights in $\mathbf{w}$, and for $w \in \text{supp}(\mathbf{w})$, I set $\#(w) := |\{i \in [n] \colon w_i = w\}|$, which is the number of players of weight w with respect to $\mathbf{w}$. An agreement is an n-tuple $\mathbf{a} = (a_1, \ldots, a_n)$ where $a_i \in [m]$ specifies the resource that is assigned to player i.

We can also express $\mathbf{a}$ in terms of the *load distribution* on each resource. For example, consider the instance $\mathbf{w} = (4, 3, 3, 1)$, $m = 3$, and the agreement $\mathbf{a} = (1, 1, 2, 2)$ where the third resource is unused. The load distribution can be written as $l(\mathbf{a}) = \{\{4, 3\}, \{3, 1\}, \varnothing\}$, meaning that a player of weight 4 and a player of weight 3 choose the same resource, while a player of weight 3 and a player of weight 1 choose another resource, and the last resource is unused. This representation groups equivalent agreements, only differing between players of same weight, into the same load expression. Two agreements are equivalent if they have the same load distribution, i.e., $\mathbf{a} \equiv \mathbf{a}' \iff l(\mathbf{a}) = l(\mathbf{a}')$.

Given an agreement $\mathbf{a}$, the load on some resource $x \in [m]$ is the sum of all the players' weights assigned to it, denoted as $v_x(\mathbf{a}) := \sum_{i : a_i = x} w_i$, which is also the cost incurred by each player assigned to x.[2] The makespan of the agreement is the maximum load on any resource, i.e. $c(\mathbf{a}) := \max_{x \in [m]} v_x(a)$. For brevity and consistency, I use the variables $i, j, k \in [n]$ for player indices or counts, and the variables $x, y, z \in [m]$ for resource indices or counts, throughout this chapter. Additionally, $W := \sum_i w_i$ denotes the total weight of all players, $\mathbf{w}[i : j] := (w_i, \ldots, w_j)$ denotes a contiguous list of sorted weights, and $W[i : j] := \sum_{k=i}^{j} w_k$ denotes the sum of weights of such a contiguous set of players.

[2] Equivalently, this is a weighted singleton congestion game with a linear cost function.

3.3 Problem Definition

The problem I study can be formulated in a number of equivalent ways. I define it as a constrained identical machine scheduling problem to emphasise its connection with the classical identical machine scheduling problem, particularly on the computational complexity of finding an optimal agreement.

Given a property P (this can be a single property or a conjunction of properties), our main problem is to minimize makespan subject to satisfying this property.

Problem: P-Makespan
Input: A weight vector $\mathbf{w} = (w_1, ..., w_n)$ of positive integers, a positive integer m, and a makespan bound t.
Task: Find a P-satisfying agreement of makespan at most t, or decide that no such agreement exists.

I also investigate the problem of satisfying P in isolation. It can be formulated as a special case of P-Makespan with $t = \infty$.

Problem: P-Satisfiability
Input: A weight vector $\mathbf{w} = (w_1, ..., w_n)$ of positive integers and a positive integer m.
Task: Find a P-satisfying agreement, or decide that no such agreement exists.

Throughout this chapter, I assume that $n > m > 1$ (i.e., there are more players than resources, and more than one resource), and that $\max_i w_i \leq t$. The case $n \leq m$ is rather trivial: an agreement where each player is assigned a different resource satisfies all of the conditions considered in this chapter, so we can answer P-Makespan by simply checking whether the largest weight in $\mathbf{w}$ exceeds the makespan bound t. Moreover, the case $m = 1$ allows only for a single agreement (all players assigned to the same resource), which can be checked to solve P-Makespan. Finally, the case $\max_i w_i > t$ clearly has no solution.

3.4 Properties and Relations

The properties that I consider in this chapter are defined as follows. Individual properties can be combined to form composite properties, e.g., **a** satisfies $WOE \wedge Eq$ means **a** satisfies WOE and **a** satisfies Eq.

Definition 1 (Single properties)**.** Given an agreement **a**, we say that **a** satisfies the following property if the corresponding conditions are met:

- **Credibility (*Cr*)**: $v_{a_i} \leq v_x + w_i$ for all $i \in [n], x \in [m]$.
- **Equality (*Eq*)**: $w_i = w_j \implies v_{a_i} = v_{a_j}$ for all $i, j \in [n]$.
- **Pareto-optimality (*PO*)**: $\nexists\, a' \in [m]^n$ such that $v_{a'_i} \leq v_{a_i}$ for all $i \in [n]$, and $v_{a'_k} < v_{a_k}$ for some $k \in [n]$.
- **Envy-freeness (*EF*)**: $v_{a_i} - w_i \leq v_{a_j} - w_j$ for all $i, j \in [n]$.
- **Weak ordered envy-freeness (*WOE*)**: $w_i < w_j, a_i \neq a_j \implies v_{a_i} - w_i \leq v_{a_j} - w_j$ for all $i, j \in [n]$.
- **Ordered envy-freeness (*OE*)**: $w_i \leq w_j, a_i \neq a_j \implies v_{a_i} - w_i \leq v_{a_j} - w_j$.
- **Strong monotonicity (*SM*)**: $w_i < w_j \implies v_{a_i} < v_{a_j}$ for all $i, j \in [n]$.
- **Weak monotonicity (*WM*)**: $w_i < w_j \implies v_{a_i} \leq v_{a_j}$ for all $i, j \in [n]$.
- **Monotonicity (*M*)**: $w_i \leq w_j \implies v_{a_i} \leq v_{a_j}$ for all $i, j \in [n]$.

Informally,

- Cr is the property that no player can benefit from a unilateral deviation, or equivalently the agreement is a pure-strategy Nash equilibrium, which ensures that the non-binding agreement is honoured by the players;
- EF is a fairness property that requires no player to want to switch their assigned resource with another;

- PO is an efficiency property which says that the agreement is on the Pareto frontier, i.e., there are no alternative agreements that can strictly lower the cost incurred by some players and not increase the cost incurred by any player;
- WOE is a weaker version of EF, which only requires the lighter player of any pair of players to not envy the assigned resource of the heavier player. This condition applies only to pairs of players that are assigned different resources, since it would make no sense for a player to envy another player assigned to the same resource.
- SM is another fairness property that says a heavier player should incur more cost than a lighter player, while WM says that a heavier player should incur no less cost than a lighter player;
- Since WOE, SM, and WM do not impose any fairness conditions on players of the same weight, I also introduce Eq, which requires players of the same weight to incur the same cost.
- Finally, OE and M combine the Eq condition with WOE and WM respectively, as I show in Proposition 4.

I use *P-agreement* to refer to an agreement satisfying P. I use the symbol $\top$ to denote the empty conjunction of properties, i.e., where there are no constraints to be satisfied.

Remark 2. Since Pareto optimality is implied by the stronger condition of credibility in our model, and is trivially satisfied by any agreement that uses all resources (given $n > m$), I do not further consider it in this chapter. Also notice that I do not define 'Strong ordered envy-freeness' (SOE), which would logically be the condition $v_{a_i} - w_i < v_{a_j} - w_j$ for all $i, j : w_i < w_j, a_i \neq a_j$. This is due to the fact that this is an unreasonably strong condition which cannot be satisfied even in trivial cases. For example, no SOE-agreement exists for the instance $\mathbf{w} = (2, 3), m = 2$; even the agreement where $l(\mathbf{a}) = \{\{3\}, \{2\}\}$ leads player of weight 2 to (strong ordered) envy the player of weight 3.

Remark 3 (Equivalent problems)**.** Notice that the $\top$-MAKESPAN problem, where no constraint is applied, is equivalent to the classical identical machine scheduling problem [52]. The Cr-SATISFIABILITY problem is equivalent to finding a pure-strategy Nash equilibrium in the singleton congestion game [57], and the Cr-MAKESPAN problem can be used to find the socially optimal pure-strategy Nash equilibrium via a binary search for the optimal t.

The relations between these properties are set out in Proposition 4. Figure 3.1 illustrates them with a Venn diagram.

Proposition 4. *[Property relations] The following relations hold:*

1. $Cr \implies PO$
2. $PO \wedge EF \implies Cr$
3. $EF \implies Eq$
4. $EF \implies WOE \implies WM$
5. $EF \implies SM \implies WM$
6. $WOE \wedge Eq \iff OE$
7. $WM \wedge Eq \iff M$

Proof.

- $(Cr \implies PO)$: I first show that an agreement is Pareto-optimal if and only if all resources are used (assuming $n > m$), i.e. $v_x(a) > 0 \; \forall x \in [m]$, a property that I call 'covering' (Co). $(Co \iff PO)$ In the forward direction, consider a covering agreement. Because the sum of all player weights, W, is constant, reduction in the cost of one player must result in some weight being redistributed to another resource, resulting in the increase in cost of another player. Thus there can be no Pareto-improvement on a covering agreement. Conversely, consider a non-covering agreement, where some resource is unused. If we move one of the players that is sharing a resource to the unused resource, then we have reduced the cost incurred by all players (the player that is

moved and all players that remain) on that shared resource, and thus found a Pareto-improvement. A non-covering agreement is therefore not Pareto-optimal. ($Cr \implies Co$) I prove the contrapositive. A non-covering agreement is clearly not credible, since any player that is sharing a resource (and some player must be sharing a resource, given $n > m$) would benefit from switching to the unused resource.

- ($PO \wedge EF \implies Cr$): I have shown above that $PO \iff Co$. I now show that $Co \wedge \neg Cr \implies \neg EF$: Consider a covering and non-credible agreement, so that there is a beneficial deviation by some player i from resource x to resource y, i.e. $v_x(a) > v_y(a) + w_i$. Because the agreement is covering, there is some player j that chooses y in a_C, so i must envy j and thus the agreement is not envy-free. The contrapositive of this statement is that $EF \implies \neg(PO \wedge \neg Cr)$, which implies $PO \wedge EF \implies Cr$ as required.

- ($EF \implies Eq$): In the case where $w_i = w_j$, $EF \implies v_{a_i} = v_{a_j} \implies Eq$.

- ($EF \implies WOE$): For any $i, j : w_i < w_j$, $EF \implies v_{a_i} - w_i = v_{a_j} - w_j$, which satisfies WOE.

- ($WOE \implies WM$): For any $i, j : w_i < w_j$, if $a_i \neq a_j$, then $WOE \implies v_{a_i} - w_i \leq v_{a_j} - w_j \implies v_{a_i} < v_{a_j}$, which satisfies WM; if $a_i = a_j$, then $v_{a_i} = v_{a_j}$, which also satisfies WM.

- ($EF \implies SM$): For any $i, j : w_i < w_j$, $EF \implies v_{a_i} - w_i = v_{a_j} - w_j \implies v_{a_i} < v_{a_j}$, which satisfies SM.

- ($SM \implies WM$): For any $i, j : w_i < w_j$, $SM \implies v_{a_i} < v_{a_j}$, which satisfies WM.

- ($WOE \wedge Eq \implies OE$): For any $i, j \in [n]$, consider two cases. If $w_i = w_j$, then $Eq \implies v_{a_i} = v_{a_j}$ which satisfies OE; if $w_i < w_j$, then $WOE \implies v_{a_i} - w_i = v_{a_j} - w_j$, which satisfies OE.

- $(OE \implies WOE \wedge Eq)$: OE implies that for any $i, j \in [n] : w_i = w_j$, $v_{a_i} - w_i \leq v_{a_j} - w_j$ and $v_{a_j} - w_j \leq v_{a_i} - w_i$, which together implies $v_{a_i} = v_{a_j}$, so $WOE \wedge Eq$ is satisfied. For any $i, j \in [n] : w_i < w_j$, $WOE \wedge Eq$ is clearly satisfied.

- $(WM \wedge Eq \implies M)$: For any $i, j \in [n]$, consider two cases. If $w_i = w_j$, then $Eq \implies v_{a_i} = v_{a_j}$ which satisfies M; if $w_i < w_j$, then $WM \implies v_{a_i} \leq v_{a_j} - w_j$, which also satisfies M.

- $(M \implies WM \wedge Eq)$: M implies that for any $i, j \in [n] : w_i = w_j$, $v_{a_i} \leq v_{a_j}$ and $v_{a_j} \leq v_{a_i}$, which together implies $v_{a_i} = v_{a_j}$, so $WM \wedge Eq$ is satisfied. For any $i, j \in [n] : w_i < w_j$, $M \implies v_{a_i} \leq v_{a_j}$, which satisfies $WM \wedge Eq$.

□

It might seem natural that $WOE \implies SM$, but this is not true, due to the fact that 'envy' does not exist between players assigned to the same resource (as specified in the pre-condition $a_i \neq a_j$ of WOE). Examples of agreements satisfying various properties are shown in Figure 3.2.

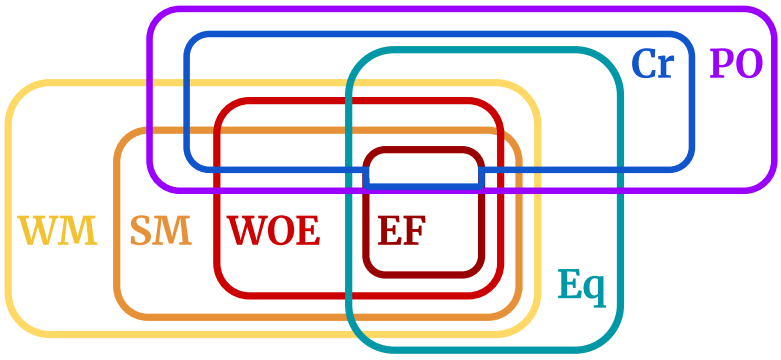
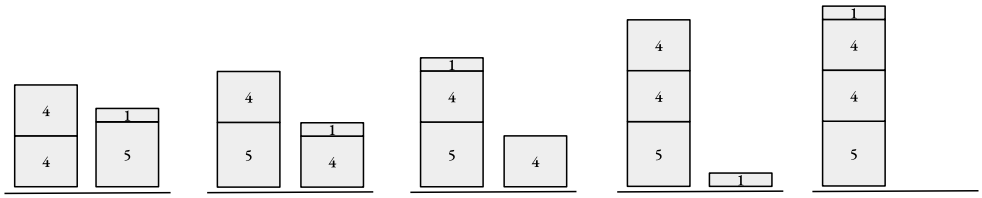


Figure 3.1: Venn diagram of relations between properties.

3.5 Contiguous Agreement

An important property that the satisfying agreements of many (conjunctions of) properties have is *contiguity*. This property allows us to quickly search for a satisfying agreement in its *canonical ordering*.



l(a), c(a)	{{4,4},{5,1}}, 8	{{5,4},{4,1}}, 9	{{5,4,1},{4}}, 10	{{5,4,4},{1}}, 13	{{5,4,4,1},∅}, 14
Contiguous?	No	Yes	No	Yes	Yes
Satisfied	Cr, Eq	Cr, WOE, WM	∅	WOE, WM, Eq	EF, WOE, WM, Eq
Not satisfied	EF, WOE, SM, WM	EF, SM, Eq	Cr, EF, WOE, SM, WM, Eq	Cr, EF, SM	Cr, SM

Figure 3.2: Examples of agreements for an instance where $\mathbf{w} = (5, 4, 4, 1)$, $m = 2$.

Definition 5 (Contiguous agreement)**.** An agreement is *contiguous* if under a non-increasing ordering of player weights where players are indexed such that $i < j \implies w_i \geq w_j$, we have $(i < j) \wedge (a_i = a_j) \implies a_i = a_k \; \forall k \in [i : j]$.

Definition 6 (Canonical ordering)**.** Assume we are given a contiguous agreement where players are indexed in a non-increasing order satisfying the contiguity definition. Then, we can further index the resources in a non-increasing order such that, for all $i, j \in [n], w_i \geq w_j \implies a_i \leq a_j$. I call this ordering of players and resources the *canonical ordering* of the contiguous agreement.

Thus, under the canonical ordering of a contiguous agreement, the first contiguous set of players are assigned to the first resource, the second contiguous set of players to the second resource, etc. The search for a contiguous agreement only has to find the right 'cuts' to make in order to assign players to resources, which significantly reduces the computational complexity of the problem.

Example 7. Consider the load distribution $l(\mathbf{a}) = \{\{2, 3\}, \{3, 3\}, \{1, 2\}\}$. This is a contiguous agreement since we can order the player weights as $(3, 3, 3, 2, 2, 1)$. We can then index the resources to express the agreement as $\{\{3, 3\}, \{3, 2\}, \{2, 1\}\}$ in its canonical ordering.

4
Complexity of Fair Coordination

4.1 Introduction

In this chapter, I explore the computational landscape of finding agreements that satisfy different combinations of the properties introduced in Chapter 3. While the base problem of identical machine scheduling is intractable, for a large number of (single and composite) properties, this problem can be solved in polynomial time.

I answer the following four questions for each (single or composite) property P, a summary of which is provided in Table 4.1.

- Existence: Does a P-satisfying agreement always exist?
- Uniqueness: If a P-satisfying agreement exists, is it unique?
- P-Satisfiability: What is the computational complexity of finding a P-satisfying agreement?
- P-Makespan: What is the computational complexity of finding a P-satisfying agreement of smallest makespan?

I find that, for certain classes of this decision problem, it is sufficient to search for a contiguous agreement which limits the search space and avoids the combinatorial

blow-up. However, when this is not possible, I provide a reduction to show that they are indeed NP-complete.

For the remainder of this chapter, I present tractable results (computable in polynomial time) in Section 4.2, and intractable results (NP-completeness) in Section 4.3. I then conclude in Section 4.4.

Property P	Exists	Unique	P-Satisfiability	P-Makespan
$\top$	Y	N	$O(n)$	NP-complete [51]
Eq	Y	N	$O(n)$	NP-complete
Cr	Y	N	$O(n \log n)$ [62]	NP-complete
$Eq \wedge Cr$	N	N	NP-complete	NP-complete
EF	N	N	$O(n\sqrt{n})$	$O(n\sqrt{n})$
$EF \wedge Cr$	N	Y	$O(n\sqrt{n})$	$O(n\sqrt{n})$
WOE	Y	N	$O(n)$	$O(n \log n)$
$WOE \wedge Eq \equiv OE$	Y	N	$O(n)$	$O(n^2)$
$WOE \wedge Cr$	N	Y	$O(n \log n)$	$O(n \log n)$
$WOE \wedge Eq \wedge Cr \equiv OE \wedge Cr$	N	Y	$O(n \log n)$	$O(n \log n)$
SM	N	N	$O(n \log n)$	$O(n \log n)$
$SM \wedge Eq$	N	N	$O(n \log n)$	$O(n \log n)$
$SM \wedge Cr$	N	Y	$O(n \log n)$	$O(n \log n)$
$SM \wedge Eq \wedge Cr$	N	Y	$O(n \log n)$	$O(n \log n)$
$WOE \wedge SM$	N	N	$O(n \log n)$	$O(n \log n)$
$WOE \wedge SM \wedge Eq$	N	N	$O(n \log n)$	$O(n \log n)$
$WOE \wedge SM \wedge Cr$	N	Y	$O(n \log n)$	$O(n \log n)$
$WOE \wedge SM \wedge Eq \wedge Cr$	N	Y	$O(n \log n)$	$O(n \log n)$
WM	Y	N	$O(n)$	NP-complete
$WM \wedge Eq \equiv M$	Y	N	$O(n)$	NP-complete
$WM \wedge Cr$	N	N	NP-complete	NP-complete
$WM \wedge Eq \wedge Cr \equiv M \wedge Cr$	N	N	NP-complete	NP-complete

Table 4.1: Summary of the satisfiability and complexity for combinations of various fairness and credibility conditions. In the third and fourth column, I either give a polynomial bound on the running time for computing a satisfying agreement with or without a makespan constraint, or state NP-completeness of deciding on existence.

4.2 Tractable Properties

In this section, I aim at algorithmic feasibility. I present a general algorithmic framework based on Algorithm 1, a polynomial-time algorithm, which I term *Sequential Contiguous Assignment (SCA)*. It exploits the fact that a P-agreement

is necessarily contiguous for a wide range of properties. The algorithm does so by searching for a contiguous agreement in its canonical ordering. I start by setting out the general, modular structure of the algorithm, and then specify the subroutines corresponding to various properties P in subsequent sections.

Algorithm 1 Sequential Contiguous Assignment (SCA)

Input: A weight vector $\mathbf{w} = (w_1, w_2, \ldots, w_n)$, a number of resources m, a property P, and a makespan bound t.
Output: A contiguous P-agreement of makespan at most t, if it exists; None otherwise.

Sort and index $\mathbf{w}$ in non-increasing order of weights.
$k \leftarrow$ First_k$(P, t, \mathbf{w}, m)$
$a_j \leftarrow 1$ for all $j \in [1 : k]$ ▷ Assign players to 1st resource
$x \leftarrow 2$ ▷ Define resource counter
$i \leftarrow k + 1$ ▷ Define player counter
$t \leftarrow W[1 : k]$ ▷ Update load bound
while $x \leq m$ and $i \leq n$ **do**
    Compute relevant input $\mathcal{I}$
    $k \leftarrow$ Subsequent_k$(P, t, \mathbf{w}[i : n], m - x + 1, \mathcal{I})$
    $a_j \leftarrow x$ for all $j \in [i, i + k]$ ▷ Assign players to resource x
    $x \leftarrow x + 1$ ▷ Update resource counter
    $t \leftarrow W[i : i + k - 1]$ ▷ Update load bound
    $i \leftarrow i + k$ ▷ Update player counter
if $i < n$ **then** ▷ Some players remain unassigned
    return None
else ▷ All players assigned to resources
    $a \leftarrow (a_1, ..., a_n)$
    if Satisfies$(\mathbf{a}, P)$ **then** ▷ Test $\mathbf{a}$ for P
        return $a = (a_1, ..., a_n)$
    else
        return None

Players are first sorted in a non-increasing order. Then the algorithm iteratively finds the next k number of players to assign to a new resource. It proceeds until either a satisfying agreement is found, or it runs out of resources before all players are assigned, in which case no satisfying agreement is found. A final check is conducted for some cases of P to ensure that the candidate agreement does indeed satisfy P. The subroutines First_k, Subsequent_k and Satisfies differ for

different problems, and are specified in the dedicated sections below. Note that a subroutine might also return None, which causes termination of the algorithm.

I make a few general observations about the algorithm: First, the makespan bound t is updated after each iteration to be the load of the resource that was just assigned to ensure a non-increasing ordering of load, which is necessary for certain fairness conditions.

Second, after the agreement of the first k players to the first resource, Subsequent_k is repeatedly called with the remaining players and resources until all players or resources are assigned. An additional argument $\mathcal{I}$ is computed and passed to the Subsequent_k subroutine; $\mathcal{I}$ contains information about the players assigned to the previous resource (e.g., maximum weight, minimum weight, number of players), which is used by Subsequent_k to ensure the satisfaction of P in the current agreement.

Finally, the algorithm allows for only a single value of k to be returned by the First_k and Subsequent_k subroutines at each iteration. This is sufficient as the algorithm searches for the agreement that is 'closest' to satisfying P, so that if no agreement is found, then no satisfying agreement exists.

4.2.1 Simple Satisfiability

For a number of simple properties, one can always find a P-agreement (rendering the decision problem P-Satisfiability trivial) in polynomial time, while P-Makespan is NP-complete. I now discuss such properties P, and address the corresponding P-Makespan problems in Section 4.3.

$\top$-Satisfiability

Any agreement satisfies $\top$ when there is no makespan bound, and thus its existence is guaranteed, but not uniqueness.

Eq-Satisfiability

Consider an agreement where all players are assigned to the same resource. This satisfies Eq since all players incur the same cost W (which is the load of the resource to which they are assigned). Its existence is thus guaranteed, but not uniqueness.

Cr-Satisfiability

The Longest Processing Time algorithm [52] is known to find a Cr-agreement in time $O(n \log n)$ [62]. It is a greedy algorithm that first sorts players into a non-increasing weight order, then sequentially assigns them to the resource with the least current load, as shown in Algorithm 2.

Algorithm 2 Longest Processing Time first (LPT)

Input: A weight vector $\mathbf{w} = (w_1, \ldots, w_n)$ and a number of resources m.
Output: An agreement.

1: Sort and index $\mathbf{w}$ in non-increasing order of weights.
2: $v_x \leftarrow 0$ for all $x \in [m]$. ▷ Initialise weights on each resource
3: **for** $i \in \{1, \ldots, n\}$ **do**
4: $\quad x \leftarrow \arg\min_{y \in [m]} v_y$ ▷ Identify least-load resource; break ties arbitrarily
5: $\quad a_i \leftarrow x$ ▷ Assign players to the least-load resource
6: $\quad v_x \leftarrow v_x + w_i$ ▷ Update resource load
7: **return** $\mathbf{a} = (a_1, ..., a_n)$

By utilising a minimum heap structure, the operation to find the least-load resource can be conducted in constant time (line 4). The update operation, which requires removing the least-load resource from the heap, then re-insert its updated weight, can be done in $O(\log n)$ time. This means that the process of assigning all players to a resource can be conducted in $O(n \log n)$ time (lines 3-6). The sorting of player weights (line 1) also takes $O(n \log n)$ time, and thus the algorithm can be performed in time $O(n \log n)$.

Proposition 8. *The LPT algorithm outputs a Cr-agreement.*

Proof. To see this, consider the incentive to deviate from their assignment for each player when a new player is assigned to a resource.

When a new player i of some weight w_i is assigned to some resource x, no player that is already assigned to a different resource has the incentive to deviate to x, since the weight of x has increased. Moreover, all players that are already assigned to x has weight $\geq w_i$, and thus x remains the resource of least-load if they remove themselves from x, so they would not benefit from switching to another resource. And thus the credibility property is satisfied at every step of the assignment process, including the final assignment. □

Notice that a Cr-agreement always exists, since the LPT algorithm always returns an agreement. However, it may not be unique due to the arbitrary tie-breaking when there are multiple resources of least-load.

4.2.2 Envy-Freeness

I provide an algorithm for solving EF-Makespan and $EF \wedge Cr$-Makespan in time $O(n\sqrt{n})$.

EF-Makespan

EF is a strong condition that can only be satisfied under a range of specific conditions:

Lemma 9. *An agreement satisfies EF if and only if each resource is either unused, or shared by $b/w_i + 1$ players of the same weight w_i, for some value $b > 0$.*

Proof. ($\Longrightarrow$) If agreement **a** satisfies EF, then $v_{a_i} - w_i = v_{a_j} - w_j$ for all $i, j \in N$. This implies that there is some value b where $b = v_{a_i} - w_i$ for all $i \in N$. Notice also that $a_i = a_j \implies v_{a_i} - w_i = v_{a_i} - w_j \implies w_i = w_j$, and thus all players sharing a resource must have the same weight. Thus each resource is either unused, or shared by $b/w_i + 1$ players of the same weight w_i.

($\Longleftarrow$) Since each used resource is shared by $b/w_i + 1$ players of weight w_i, $v_{a_i} - w_i = b$ for all $i \in N$, and thus the agreement satisfied EF. □

Lemma 9 implies that in order to reach an EF-agreement, there are some strict conditions on the composition of player weights.

Corollary 10. *If an agreement satisfies EF, then there exists some value $b > 0$ such that*

$$w_i \mid b \ \forall w_i \in \text{supp}(\mathbf{w}) \qquad \text{(common multiple)}$$

$$\frac{b}{w_i} + 1 \mid \#(w_i) \ \forall w_i \in \text{supp}(\mathbf{w}) \qquad \text{(balanced loads)}$$

$$\sum_{w_i \in \text{supp}(\mathbf{w})} \frac{\#(w_i)}{b/w_i + 1} \leq m \qquad \text{(resource constraint)}$$

Conversely, given a set of players N with weights $\mathbf{w}$ and m resources, if there exists some value $b > 0$ such that the three conditions above are satisfied, then there exists an EF-agreement.

Proof. ($\implies$) Lemma 9 implies that such b exists and is divisible by w_i for all $w_i \in \text{supp}(\mathbf{w})$ (common multiple condition). Moreover, to ensure that $v_{a_i} - w_i = b$ for all w_i, $\#(w_i)$ must be divisible by $b/w_i + 1$ (balanced loads condition). Finally, the total number of resources that are shared across all players cannot exceed the number of resources available (resource constraint condition).

($\impliedby$) If the three conditions are satisfied, then we can construct an agreement where that each resource is either unused, or shared by $b/w_i + 1$ players of weight w_i. By Lemma 9 this agreement is a EF-agreement. □

Observe that each value of b corresponds to a unique EF-agreement. However, as the below example shows, an instance may support multiple feasible values of b, and therefore have multiple EF-agreements.

Example 11 (Non-uniqueness of EF-agreements)**.** Consider the instance where there are 6 players of weight 2 and 15 players of weight 1, and there are 10 available machines. Formally, $\text{supp}(\mathbf{w}) = \{2, 1\}$, $\#(2) = 6$, $\#(1) = 15$, and $m = 10$. The value $b = 2$ is a valid candidate, since we can easily check that all conditions are satisfied: $2 \mid 2$ and $1 \mid 2$, so the common multiple condition holds. $2/2 + 1 \mid 6$ and $2/1 + 1 \mid 15$, so the balanced loads condition holds. And finally $\frac{6}{2} + \frac{15}{3} = 8 \leq 10$, so the resource constraint holds. By Corollary 10, we know that there exists an EF-agreement corresponding to the

value $b = 2$. By Lemma 9, the EF-agreement must have load distribution $\{\{2,2\},\{2,2\},\{2,2\},\{1,1,1\},\{1,1,1\},\{1,1,1\},\{1,1,1\},\{1,1,1\},\emptyset,\emptyset\}$.

There is another valid candidate where $b = 4$, with the corresponding EF-agreement having the load distribution
$\{\{2,2,2\},\{2,2,2\},\{1,1,1,1,1\},\{1,1,1,1,1\},\{1,1,1,1,1\},\emptyset,\emptyset,\emptyset,\emptyset,\emptyset\}$.

Corollary 10 means that to determine whether an EF-agreement exists, we can search through the space of feasible values of b and check that these conditions are satisfied. Algorithm 3 is one such algorithm which checks all feasible values of b and return all EF-agreements.

Algorithm 3 EF-SATISFIABILITY

Input: A vector of n weights $\mathbf{w} = (w_1, w_2, \ldots, w_n)$, an integer number of resources m, and a makespan threshold t.
Output: All EF-agreement $\mathbf{a}$ of makespan at most t, if it exists; None otherwise.

1: Compute supp($\mathbf{w}$), and compute $\#(w_i)$ for each $w_i \in$ supp($\mathbf{w}$).
2: Initialise the candidate set $b_{cand} = \emptyset$.
3: Compute D, the set of all divisors of $\#(w_1)$
4: **for** $d \in D$ **do**
5: Set $b_{cand} = b_{cand} \cup \{(d-1)w_1\}$.
6: **for** $b \in b_{cand}$ **do**
7: **if** $\sum_{w_i \in \text{supp}(\mathbf{w})} \frac{\#(w_i)}{b/w_i+1} > m$ **then**
8: Set $b_{cand} = b_{cand} \setminus \{b\}$
9: **Continue**
10: **for** $w_i \in$ supp($\mathbf{w}$) **do**
11: **if** $w_i \nmid b$ or $(b/w_i + 1) \nmid \#(w_i)$ **then**
12: Set $b_{cand} = b_{cand} \setminus \{b\}$
13: Initialise the set of satisfying agreements $a_{set} = \emptyset$
14: **for** $b \in b_{cand}$ **do**
15: **for** $w_i \in$ supp($\mathbf{w}$) **do**
16: Assign $b/w_i + 1$ players to each resource, for a total of $\frac{\#(w_i)}{b/w_i+1}$ resources.
17: Add agreement associated with b to a_{set}
18: **return** a_{set}

The algorithm first generates a candidate set of values for b, based on the *balanced loads* condition for the largest weight w_1 (lines 3-5).

For each candidate, it then checks the resource constraint condition (lines 7-8), and for each distinct weight in supp($\mathbf{w}$), the balanced loads and common

multiple conditions (lines 10-11), and removes any candidate that fails these checks. This results in a set of feasible values of b. A set of unique agreements, one for each b value, is returned (lines 13-18). If there is no feasible value of b, then the algorithm returns an empty set.

Because the above algorithm returns all EF-agreements, we can simply look at the makespan of the optimal EF-agreement (i.e., the makespan of the agreement associated with the smallest feasible b, which is $b + w_1$ where w_1 is the largest weight) to answer the EF-Makespan problem for a given value of t.

Since $\#(w_1) \leq n$, finding all divisors of $\#(w_1)$ is in $O(\sqrt{n})$. Because also $\text{supp}(\mathbf{w}) \leq n$, the loops through all divisors of w_1 and $\text{supp}(\mathbf{w})$ (lines 6-12, 14-17) takes $O(n\sqrt{n})$, which is the running time of this algorithm.

$EF \wedge Cr$-Makespan

The $EF \wedge Cr$-Makespan requires a simple modification to Algorithm 3. The credibility condition requires that no player can benefit by unilaterally switching to another resource.

By construction, an EF-agreement that corresponds to some value b satisfies the condition $v_{a_i} - w_i = b < v_{a_j}$ for all $i, j \in [n]$. This means no player can benefit from switching to a resource that is already used. Thus, to additionally satisfy Cr, we simply require that there are no unused resource for a player to switch to.

I do this by changing the *resource constraint* condition to an equality, i.e. $\sum_i \frac{\#(w_i)}{b/w_i+1} = m$, which corresponds to changing the pre-condition in line 7 of the algorithm to $\sum_i \frac{\#(w_i)}{b/w_i+1} \neq m$.

Notice that at most one value of b can satisfy the condition $\sum_i \frac{\#(w_i)}{b/w_i+1} = m$, an $EF \wedge Cr$-agreement is unique if it exists.

Here, I provide an example of an instance that has an EF-agreement, but no $EF \wedge Cr$ agreement.

Example 12. Consider the instance $w = (3, 3, 3, 2, 2, 2, 2), m = 3$, where $\text{supp}(\mathbf{w}) = \{3, 2\}, \#(3) = 3, \#(2) = 4$. The only feasible value for b is 6, with the corresponding

EF-agreement where $l(\mathbf{a}) = \{\{3,3,3\},\{2,2,2,2\},\emptyset\}$, with a makespan of 9. Therefore the answer to EF-MAKESPAN is $\mathbf{a}$ if $t \geq 9$, and None otherwise. Because there is an unused resource in $\mathbf{a}$, $\mathbf{a}$ is not $EF \wedge Cr$-satisfying.

4.2.3 Weak Ordered Envy-Freeness

Since WOE does not place any restrictions on players assigned to the same resource (i.e., a player cannot be envious of another player who is assigned to the same resource), assigning all players to a single resource trivially satisfies WOE, and thus an WOE-agreement always exists, and it can be returned in time $O(n)$.

However, it may not be unique. For example, the agreements $\mathbf{a}$ and $\mathbf{a}'$ where $l(\mathbf{a}) = \{\{2,2\},\{2,1\}\}$ and $l(\mathbf{a}') = \{\{2,2,2\},\{1\}\}$ both satisfy WOE.

Moreover, we can make a stronger statement about WOE-agreements.

Lemma 13. *Every WOE-agreement is contiguous.*

Proof. Suppose on the contrary that there exists a non-contiguous WOE-agreement where, under any non-increasing ordering of players and resources, we can find some $i,j \in [n]$ such that $w_i < w_j$ and $a_i < a_j$ (i.e., there is no canonical ordering). Then $v_{a_i} \geq v_{a_j}$, and therefore $v_{a_i} - w_i > v_{a_j} - w_j$, which contradicts WOE. □

Thus if P contains WOE, it is sufficient to consider only contiguous agreements to answer P-MAKESPAN.

In addition, we can show that under the canonical ordering, if WOE holds between players in adjacent resources, then WOE holds globally.

Lemma 14. *Let $\mathbf{a}$ be a contiguous agreement in its canonical ordering. If the WOE condition holds between players assigned to any two adjacent resources, then $\mathbf{a}$ satisfies WOE.*

Proof. Given a contiguous agreement $\mathbf{a}$, consider three consecutive resources indexed x, $x+1$ and $x+2$ in its canonical ordering (i.e., $i < j \implies w_i \geq w_j$, and $v_x \geq v_{x+1} \geq v_{x+2}$), and assume that the WOE condition holds between players assigned to any two adjacent resources. I.e., $v_{x+1} - \underline{w}_{x+1} \leq v_x - \overline{w}_x$ and $v_{x+2} - \underline{w}_{x+2} \leq$

$v_{x+1} - \overline{w}_{x+1}$, where $\underline{w}_{x+1}$ and $\underline{w}_{x+2}$ are the lightest players assigned to $x+1$ and $x+2$ respectively, and $\overline{w}_x$ and $\overline{w}_{x+1}$ are the heaviest players assigned to x and $x+1$ respectively. Because players are in canonical ordering, $\overline{w}_{x+1} \geq \underline{w}_{x+1}$, and therefore $v_{x+2} - \underline{w}_{x+2} \leq v_{x+1} - \overline{w}_{x+1} \leq v_{x+1} - \underline{w}_{x+1} \leq v_x - \overline{w}_x$, which means that no player assigned to $x+2$ weakly envies anyone assigned to x. In other words, in a canonical ordering, the WOE condition is transitive, and therefore WOE holds between all players across all resources. □

Lemma 13 suggests that Algorithm 1 can be used to solve P-Makespan whenever WOE is entailed by P.

Hence, we obtain a correct algorithm by specifying the subroutines. I first discuss the subroutines and their intuition, then provide a formal proof to the following theorem.

Theorem 15. *P-Makespan can be solved in time $O(n \log n)$ for $P \in \{WOE, WOE \wedge Cr, WOE \wedge Eq \wedge Cr\}$, and in time $O(n^2)$ for $P = WOE \wedge Eq$.*

WOE-Makespan subroutines

I define three subroutines that are called by Algorithm 1 to solve the WOE-Makespan problem. The idea is to assign players a resource as early as possible as it does not harm envy to assign heavier players to a resource.

procedure First_k($P = WOE, t, \mathbf{w}, m$)
  return $\max\{i \in [n] \colon W[1:i] \leq t\}$

This subroutine assigns the maximum number of players to the first resource without exceeding the makespan bound t. Notice that for WOE-Satisfiability where $t = \infty$, the subroutine returns $k = n$, i.e., all players are assigned to the first resource.

For the agreement of players to subsequent resources, let $\mathcal{I}$ be the largest weight assigned to the previous resource. This information is used to determine the largest number of players that can be assigned to the next resource without triggering envy,

i.e., we need to ensure that the load on the current resource less its lightest player does not exceed the load on the previous resource less its heaviest player. The last min operation ensures that k does not exceed the number of remaining players.

procedure SUBSEQUENT_K($P = WOE, t, \mathbf{w}, m, \mathcal{I}$)
  Set $\hat{i} = \max\{i : W[1:i] \leq t - \mathcal{I}\}$
  return $\min(\hat{i} + 1, |\mathbf{w}|)$

Notice that, by construction, the WOE condition is satisfied between players assigned to any two adjacent resources. By Lemma 14, this means that the candidate agreement satisfies WOE. So no further check for WOE-satisfaction is required and the SATISFIES subroutine always returns True.

procedure SATISFIES($\mathbf{a}, P = WOE$)
  return True

$WOE \wedge Eq$-Makespan subroutines

I begin with a structural proposition about $WOE \wedge Eq$-agreements, which says that if players of the same weight are assigned to different resources, then it must be the case that all players sharing those resources are of the same weight.

Lemma 16. *Let* $\mathbf{a}$ *be a* $WOE \wedge Eq$*-agreement. If there exist* $i, j \in [n]$ *with* $w_i = w_j$ *and* $a_i \neq a_j$*, then* $w_k = w_j$ *for all* $k \in [n] : a_k = a_i \vee a_k = a_j$*.*

Proof. Suppose for contradiction there exists some player k s.t. $a_k = a_i$ and $w_k \neq w_i$, and let $v = v_{a_i} = v_{a_j}$ be the load on resources a_i and a_j, which must be the same due to Eq. If $w_k > w_i$, then player j would weakly envy k, since $v - w_k < v - w_j$, so WOE is not satisfied. If $w_k < w_i$, then player k would weakly envy j, since $v - w_j < v - w_k$, so WOE is again not satisfied. □

We thus need to modify our previous WOE subroutines to ensure Eq-satisfaction by changing the way k is determined in FIRST_K and SUBSEQUENT_K:

- If not all players assigned to the same resource have the same weight, then the next player to be assigned to the next resource (the $(k+1)$st player) cannot share the same weight as the last player of the current assignment (the kth player).
- If all players assigned to the same resource have the same weight, then all players of such weight must be evenly distributed among some number of resources so they incur the same cost. I.e., if k players of the same weight w_i are assigned to some resource, then it must be the case that $k \mid \#(w_i)$ and all assigned resources have the same load.

The subroutine First_k for $WOE \wedge Eq$-Makespan is as follows:

procedure First_k($P = WOE \wedge Eq, t, \mathbf{w}, m$)
  Set $n = |\mathbf{w}|$, $w_{n+1} = 0$
  if $\{i \in [n] \colon (W[1:i] \leq t) \wedge (w_{i+1} \neq w_i)\} \neq \emptyset$ **then**
    return $\max\{i \in [n] \colon (W[1:i] \leq t) \wedge (w_{i+1} \neq w_i)\}$
  else
    return $\max\{i \in [n] \colon (i \mid \#(w_1)) \wedge (i \cdot w_1 \leq t)\}$

For Subsequent_k, we need a number of additional information to ensure $WOE \wedge Eq$-satisfaction. Let $\mathcal{I}$ be the tuple $(\underline{w}, \overline{w}, k')$, representing the weight of the last player, the weight of the first player, and the number of players assigned to the previous resource, respectively. $\underline{w}$ and k' are used to ensure satisfaction of Eq by checking whether the current first player shares the same weight as the last player assigned to the previous resource; if so, then we simply need to assign k' players (of the same weight) to the current resource. $\overline{w}$ is used to ensure WOE satisfaction, as we have seen before.

procedure Subsequent_k($P = WOE \wedge Eq, t, \mathbf{w}, m, \mathcal{I}$)
  Let $\mathcal{I} = (\overline{w}, \underline{w}, k')$
  if $w_1 = \underline{w}$ **then** ▷ Continuation of players of equal weight
    return k'
  Set $n = |\mathbf{w}|$, $b = t - \overline{w}$, $w_{n+1} = 0$.

if $\{i \in [n] \colon (W[1:i-1] \leq b) \wedge (w_{i+1} \neq w_i)\} \neq \emptyset$ **then**
  return $\max\{i \in [n] \colon (W[1:i-1] \leq b) \wedge (w_{i+1} \neq w_i)\}$
else
  return $\max\{i \in [n] \colon (i \mid \#(w_1)) \wedge ((i-1) \cdot w_1 \leq b)\}$

Again, no further checks whether $WOE \wedge Eq$ is satisfied is required, so the SATISFIES subroutine always returns True.

procedure SATISFIES($(\mathbf{a}, P = WOE \wedge Eq)$)
  return True

$WOE \wedge Cr$-Makespan subroutines

While the WOE condition implicitly requires that resources assigned later have no greater load than the resources before it, the Cr condition requires that such differences are not so large that players assigned to earlier resources could benefit from deviating to later resources. The idea is that, in each iteration, the subroutines now assign a number of players k to a resource to ensure that the average load of the remaining resources, $\frac{W-W[1:k]}{m-1}$, can satisfy $WOE \wedge Cr$, i.e.,

$$W[1:k-1] \leq \frac{W - W[1:k]}{m-1} \leq W[1:k] \tag{4.1}$$

The first inequality is a necessary (but not sufficient) condition to satisfy Cr. The average load on the remaining resources must be sufficiently large to prevent player k from deviating to the least-load resource. We need only satisfy this for player k (and not others assigned to the same resource) to produce the largest lower-bound.

The second inequality is a necessary (but not sufficient) condition to satisfy WOE. It ensures that the average load on the remaining resources is sufficiently small to ensure a canonical ordering of contiguous agreement (Lemma 13). It's easy to see that at most one value of k satisfies these bounds:

Lemma 17. *At most one value of k satisfies Equation* (4.1)*.*

Proof. Suppose on the contrary that k and $k+l$ both satisfy the bounds, where l is a positive integer. Then by combining the two sets of bounds we have,

$$\begin{aligned} W[1:k-1] &\leq \frac{W - W[1:k]}{m-1} \leq W[1:k] \\ &\leq W[1:k+l-1] \leq \frac{W - W[1:k+l]}{m-1} \leq W[1:k+l] \end{aligned}$$

which cannot be true since $W[1:k] < W[1:k+1]$, which implies $\frac{W-W[1:k]}{m-1} > \frac{W-W[1:k+1]}{m-1}$. □

As we apply this argument to every iteration, we must reach a unique canonical ordering of resources if one exists.

Corollary 18 (Uniqueness)**.** *If* $\mathbf{a}$ *and* $\mathbf{a}'$ *satisfy* $WOE \wedge Cr$*, then* $\mathbf{a}$ *and* $\mathbf{a}'$ *are identical under their canonical orderings.*

The example below demonstrates that k may not exist.

Example 19. Consider the instance where $\mathbf{w} = (3, 2, 2)$ and $m = 2$. If $k = 1$, then we get $(W - W[1\colon 1])/(m-1) = 4$, which exceeds the upper-bound $W[1\colon 1] = 3$. If $k = 2$, then $(W - W[1\colon 2])/(m-1) = 2$, which falls below the lower-bound $W[1\colon 1] = 3$.

Explicitly, the agreement of players to resources is governed by the following subroutines:

procedure FIRST_K($P = WOE \wedge Cr, t, \mathbf{w}, m$)
  if $\exists k\colon W[1:k-1] \leq \frac{W-W[1:k]}{m-1} \leq W[1:k] \leq t$ **then**
    return k
  else return None

procedure SUBSEQUENT_K($P = WOE \wedge Cr, t, \mathbf{w}, m, \mathcal{I}$)
  if $m = 1$ **then** ▷ Final resource
    return $|\mathbf{w}|$

```
    if ∃k: W[1:k−1] ≤ (W−W[1:k])/(m−1) ≤ W[1:k] ≤ t then
        return k
    else return None
```

Finally, even if the iterations of assigning players to resources result in a unique candidate agreement $\mathbf{a}$, $\mathbf{a}$ has only satisfied the necessary, but not sufficient, conditions of $WOE \wedge Cr$. Thus a final test is required to check that the candidate agreement satisfies $WOE \wedge Cr$. For WOE, we need only check that the lightest player of each resource assignment does not envy the heaviest player assigned to the previous resource, and there are at most $m - 1 < n$ resources to check; for Cr, since resource loads are ordered non-increasingly, we need only check that the heaviest player of each resource does not benefit from deviating to the last and least-loaded resource, for which there are again $m - 1 < n$ resources to check.[1] These checks can thus be done in time $O(n)$.

```
procedure SATISFIES(a, P = WOE ∧ Cr)
    for i ∈ [n] do
        if v_{a_i} > v_m + w_i then                          ▷ Fails Cr
            return None
    for x ∈ [m − 1] do
        Set w̄ = max{w_i : a_i = x}
        Set w̲ = min{w_i : a_i = x + 1}
        if v_{x+1} − w̲ > v_x − w̄ then                        ▷ Fails WOE
            return None
    return True
```

$WOE \wedge Eq \wedge Cr$-Makespan subroutines

Since the previous algorithm for $WOE \wedge Cr$-MAKESPAN produces only a single feasible candidate, we need only conduct an additional check on whether this candidate satisfies Eq in the SATISFIES subroutine to answer this case. The subroutines FIRST_K and SUBSEQUENT_K remain unchanged:

[1] For simplicity we check that each player has no incentive to deviate to the least-load resource in the subroutine.

procedure FIRST_K($P = WOE \wedge Eq \wedge Cr, t, \mathbf{w}, m$)
  if $\exists k : W[1:k-1] \leq \frac{W - W[1:k]}{m-1} \leq W[1:k] \leq t\}$ **then**
    return k
  else
    return None

procedure SUBSEQUENT_K($P = WOE \wedge Eq \wedge Cr, t, \mathbf{w}, m, \mathcal{I}$)
  if $m = 1$ **then** ▷ Final resource
    return $k = |\mathbf{w}|$
  if $\exists k : W[1:k-1] \leq \frac{W - W[1:k]}{m-1} \leq W[1:k] \leq t\}$ **then**
    return k
  else
    return None

procedure SATISFIES($\mathbf{a}, P = WOE \wedge Eq \wedge Cr$)
  for $i \in [n]$ **do**
    if $v_{a_i} > v_m + w_i$ **then** ▷ Fails Cr
      return None
  for $i \in [n-1]$ **do**
    if $w_i = w_{i+1} \wedge v_{a_i} \neq v_{a_{i+1}}$ **then** ▷ Fails Eq
      return None
  for $x \in [m-1]$ **do**
    Set $\overline{w} = \max\{w_i : a_i = x\}$
    Set $\underline{w} = \min\{w_i : a_i = x+1\}$
    if $v_{x+1} - \underline{w} > v_x - \overline{w}$ **then** ▷ Fails WOE
      return None
  return True

Proof of Theorem 15

Having introduced the relevant subroutines for each property P that entails WOE, I restate Theorem 15 and provide a formal proof below.

Theorem 15. *P-MAKESPAN can be solved in time $O(n \log n)$ for $P \in \{WOE, WOE \wedge Cr, WOE \wedge Eq \wedge Cr\}$, and in time $O(n^2)$ for $P = WOE \wedge Eq$.*

Proof. I show that Algorithm 1, combined with the relevant subroutines, solves P-MAKESPAN for each property P stated in the theorem. I do so by showing

that the algorithm terminates, is sound (i.e., if it returns an agreement, then this agreement has the required property), is complete (i.e., if an agreement with the required property exists, then such an agreement is returned by the algorithm), and runs in time $O(n \log n)$ or $O(n^2)$ respectively.

Termination: In each case of P, each iteration of SUBSEQUENT_K assigns players to a resource, or terminates if no feasible k is found. Since there are finitely many players and resources, the algorithm always terminates.

For soundness, completeness, and running time, let us consider each case of P:

Case $P = WOE$:

Soundness: By construction, at each iteration of the assignment process, no player envies any player assigned to the previous resource. It is sufficient to check that the lightest player of the current assignment does not envy the heaviest player of the previous assignment. Notice also that the WOE condition implies that the current assignment is such that its load less its lightest player is no greater than the load less its heaviest player in the previous assignment. This implies that the load less than its heaviest player in the current assignment is no greater than the load less its heaviest player in the previous assignment, and therefore the WOE-satisfaction condition is transitive across iterations, and that it is sufficient to check for WOE-satisfaction between consecutive resources. Also notice that the sequentially assigned load on resources are non-increasing due to WOE, so it is sufficient to check that the load of the first resource is no greater than the makespan threshold.

Completeness: Suppose for contradiction that the algorithm does not return an agreement when a solution exists, which we call $\mathbf{a}$. We know that $\mathbf{a}$ must be contiguous due to Lemma 13. Since the algorithm does not return an agreement, it had run out of available resources to assign players to. This implies $\mathbf{a}$ must have assigned more players to some resources than the algorithm. However, no additional players could be assigned to the first resource due to the makespan constraint, and no more players could be assigned to each subsequent resource due to the WOE constraint.

Running time: Each iteration of assigning players to the next resource takes time $O(k)$ to find the maximum number of players that can be assigned at the makespan threshold or to satisfy the WOE condition, where k is the number of players assigned. The sum of k across all iterations is n, so the assignment part of the algorithm is in $O(n)$. This leaves the sorting of players as the dominant running time, at $O(n \log n)$.

Case $P = WOE \wedge Eq$:

Soundness: By construction, if the algorithm outputs an agreement, then the agreement is of makespan at most t (the first and most-loaded resource satisfies this constraint), satisfies WOE through the transitivity of WOE-satisfaction across iterations, and satisfies Eq due to the fact that all players of the same weight are eventually distributed across resources.

Completeness: Suppose for contradiction that the algorithm does not return an agreement when a solution exists, which we call $\mathbf{a}$. Because the algorithm only fails to return an agreement if it runs out of resources while there are still players to be assigned, it must be the case that $\mathbf{a}$ assigns more players to some resource than the algorithm. This implies that one of the conditions for selecting k is breached, and thus $\mathbf{a}$ cannot satisfy $WOE \wedge Eq$.

Running time: Because each iteration of assigning players to the next resource takes up to $O(n)$ operations (may require checking all $i \in [n]$ to find the maximum feasible i), the process of finding a candidate takes time $O(n^2)$. This dominates the running time of sorting players ($O(n \log n)$).

Case $P = WOE \wedge Cr$:

Soundness: By construction, if the algorithm outputs an agreement, then its makespan is at most t (this is an explicit constraint on the first assigned resource, and holds implicitly through the WOE bound for subsequent assignments), and has been checked for $WOE \wedge Cr$-satisfaction by the subroutine $Satisfies(a, P = WOE \wedge Cr)$.

Completeness: Suppose for contradiction that the algorithm does not return an agreement when a solution exists, which we call $\mathbf{a}$. Then $\mathbf{a}$, in its canonical ordering, must assigned to some resource that does not correspond to the k players

prescribed by the algorithm. Consider the first resource (in canonical ordering) that fails this. The assignment of players to this resource fails to satisfy some necessary condition of WOE or Cr, or have makespan greater than t, and therefore $\mathbf{a}$ is not a solution to the problem.

Running time: Because each iteration of assigning players to the next resource takes $O(k)$ operations (by calculating, for each value up to k, $W[1:k-1]$, $\frac{W-W[1:k]}{m-1}$ and $W[1:k]$), and that the sum of ks across iterations is n, the assignment part of the algorithm runs in time $O(n)$. This leaves the sorting of players as the dominant running time, at $O(n \log n)$.

Case $P = WOE \wedge Eq \wedge Cr$:

Soundness and completeness: This is essentially the same as the argument for $WOE \wedge Cr$, but with the added check for Eq.

Running time: Compared to $WOE \wedge Cr$, the additional check for Eq is in time $O(n)$, which again leaves the sorting of players as the dominant running time, at $O(n \log n)$. □

Satisfiability Problems

Finally, I briefly consider the Satisfiability problems where P entails WOE.

Notice that for WOE-Satisfiability and $WOE \wedge Eq$-Satisfiability, assigning all players to the first resource is always a solution, which can be returned in time $O(n)$. Also, because there is at most one feasible agreement for the $WOE \wedge Cr$-Makespan and $WOE \wedge Eq \wedge Cr$-Makespan problems, a solution is unique if it exists. Theorem 15 implies that the corresponding satisfaction problem (i.e., with $t = \infty$) can be solved in $O(n \log n)$.

Corollary 20. *WOE-Satisfiability and $WOE \wedge Eq$-Satisfiability can be solved in time $O(n)$. $WOE \wedge Cr$-Satisfiability and $WOE \wedge Eq \wedge Cr$-Satisfiability can be solved in time $O(n \log n)$.*

4.2.4 Strong Monotonicity

Let us begin by making two observations about SM-agreements. First notice that an SM-agreement is always contiguous.

Lemma 21. *Every SM-agreement is contiguous.*

Proof. Suppose on the contrary that there exists an SM-agreement where, under any non-increasing ordering of players and resources, we can find some i, j such that $w_i < w_j$ and $a_i < a_j$. Then $v_{a_i} \geq v_{a_j}$, which contradicts SM. □

Another observation we can make is that in an SM-agreement, all players sharing the same resource must have the same weight. Since all such players incur the same cost, if some players are lighter than others, the SM condition would fail.

Lemma 22. *Let* $\mathbf{a}$ *be an SM-agreement. If $a_i = a_j$ for players $i, j \in [n]$, then $w_i = w_j$.*

Proof. Suppose on the contrary that there exists $i, j \in [n]$ where $a_i = a_j$ and $w_i < w_j$. Then $v_{a_i} = v_{a_j}$, which contradicts SM. □

In the remainder of this section, I discuss and present the relevant subroutines of the SCA algorithm for each property containing SM, to support the following theorem.

Theorem 23. *P-Makespan and P-Satisfiability can be solved in time $O(n \log n)$ for $P \in \{SM, SM \wedge Eq, SM \wedge Cr, SM \wedge Eq \wedge Cr\}$.*

SM-Makespan subroutines

Given Lemmas 21 and 22, a natural way of searching for an SM-agreement is to look for its canonical ordering, by first sorting players in non-increasing order of weight and sequentially assign all players of a given weight to as few resources as possible while ensuring SM-satisfaction.

In First_k, if $\#(w_1)$ is too large for all players of weight w_1 to be assigned to the first resource, then we first calculate the minimum number of resources required

to accommodate all $\#(w_1)$ players, then distribute these players as evenly as possible, to ensure that the last resource to which players of weight w_1 are assigned has the highest possible load. We do this to ensure that the new upper-bound on the load of the next resource is as high as possible.

procedure FIRST_K($P = SM, t, \mathbf{w}, m$)
  if $\#(w_1) \cdot w_1 \leq t$ **then**
    return $\#(w_1)$
  else
    Compute $\hat{k} = \max\{k : k \cdot w_1 \leq t\}$
    Set $r = \lceil \frac{\#(w_1)}{\hat{k}} \rceil$
    return $\lceil \#(w_1)/r \rceil$

I use a similar process for SUBSEQUENT_K, except that an additional check occurs on whether the current first player has the same weight as the players assigned to the previous resource. If so, then the load bound is a weak inequality, since players of the same weight can incur the same cost. If not, then the load bound is a strong inequality, to ensure that the players of the lighter weight incur strictly smaller cost than the heavier players.

procedure SUBSEQUENT_K($P = SM, t, \mathbf{w}, m, \mathcal{I}$)
  Set $\mathcal{I}$ to be the weight of players in previous resource
  if $\mathcal{I} = w_1$ **then** ▷ Load must be $\leq t$
    if $\#(w_1) \cdot w_1 \leq t$ **then**
      return $\#(w_1)$
    else
      Compute $\hat{k} = \max\{k : k \cdot w_1 \leq t\}$
      Set $r = \lceil \frac{\#(w_1)}{\hat{k}} \rceil$
      return $\lceil \#(w_1)/r \rceil$
  else ▷ Load must be $< t$
    if $\#(w_1) \cdot w_1 < t$ **then**
      return $\#(w_1)$
    else
      Compute $\hat{k} = \max\{k : k \cdot w_1 < t\}$
      Set $r = \lceil \frac{\#(w_1)}{\hat{k}} \rceil$
      return $\lceil \#(w_1)/r \rceil$

Finally, the subroutines above ensure that there are no additional checks to be made, so the SATISFIES subroutine simply returns true.

```
procedure SATISFIES(a, P = SM)
    return True
```

$SM \wedge Eq$-Makespan subroutines

To search for a solution to $SM \wedge Eq$-MAKESPAN, we need to additionally ensure that all players of the same weight incur exactly the same cost. This means that they must be distributed evenly across a number of resources (rather than the minimum number of resources), as shown in the following subroutines.

```
procedure FIRST_K(P = SM ∧ Eq, t, w, m)
    if #(w_1) · w_1 ≤ t then
        return #(w_1)
    else
        return max{k : (k · w_1 ≤ t) ∧ (k | #(w_1))}
```

```
procedure SUBSEQUENT_K(P = SM ∧ Eq, t, w, m, I)
    Let I be the weight of players in previous resource
    if I = w_1 then                                  ▷ Load must be ≤ t
        if #(w_1) · w_1 ≤ t then
            return #(w_1)
        else
            return max{k : (k · w_1 ≤ t) ∧ (k | #(w_i))}
    else                                             ▷ Load must be < t
        if #(w_1) · w_1 < t then
            return #(w_1)
        else
            return max{k : (k · w_1 < t) ∧ (k | #(w_i))}
```

No further checks are required, so the SATISFIES subroutine always returns True.

procedure SATISFIES($\mathbf{a}, P = SM \wedge Eq$)
  return True

$SM \wedge Cr$-Makespan subroutines

To satisfy Cr we cannot simply assign the maximal number of players to each resource, as doing so might result in a resource being unused or having a low load, and thus make it beneficial for some players to deviate to this resource. We need to ensure that, when choosing the number of players k to assign to each resource, that the remaining players can be distributed so that all subsequent resources have sufficiently high load.

To ensure that an agreement satisfies Cr, we must impose the same bounds shown in Equation (4.1) at each iteration of the assignment process. As I have shown in Corollary 18, at most one value of k exists at each iteration, so there is a unique candidate solution which we can find and check for the $SM \wedge Cr$-MAKESPAN problem.

Since players sharing the same resource must be of the same weight, we can re-write Equation (4.1) as $(k-1)w_1 \leq \frac{W-kw_1}{m-1} \leq kw_1$ where k is the number of players of the weight w_1 assigned to the resource. The only integer value that k can take is $\lceil \frac{W}{mw_1} \rceil$, if it exists, and we can check if it is less than or equal to $\#(w_1)$, the number of remaining players of weight w_1.[2]

Repeating this step for each resource eventually yields the result that either there is no satisfying agreement, or that all players are assigned, at which point we can check whether the candidate agreement satisfies Cr overall. This can be done by checking whether each player has the incentive to deviate to the least-loaded resource.[3] These steps are shown in the following subroutines to Algorithm 1.

[2]Rearranging the bounds gives $\frac{W}{mw_1} \leq k \leq \frac{W}{mw_1} + \frac{m-1}{m}$.

[3]It is in fact sufficient to check that the heaviest player assigned to each resource has no incentive to deviate to the least-loaded resource.

procedure FIRST_K($P = SM \wedge Cr, t, \mathbf{w}, m$)
  if $\exists k : (k-1)w_1 \leq \frac{W-kw_1}{m-1} \leq kw_1 \leq t \wedge k \leq \#(w_1)$ **then**
    return k
  else return None

procedure SUBSEQUENT_K($P = SM \wedge Cr, t, \mathbf{w}, m, \mathcal{I}$)
  Let $\mathcal{I}$ be the weight of players in previous resource
  if $\mathcal{I} = w_1$ **then** ▷ Load must be $\leq t$
    if $m = 1$ **then** ▷ Final resource
      if $w_1 = w_{|\mathbf{w}|}$ **then**
        return $|\mathbf{w}|$
      else return None
    if $\exists k : (k-1)w_i \leq \frac{W-kw_i}{m-1} \leq kw_i \leq t \wedge k \leq \#(w_1)$ **then**
      return k
    else return None
  else ▷ Load must be $< t$
    if $m = 1$ **then** ▷ Final resource
      if $w_1 = w_{|\mathbf{w}|} \wedge W[1 : |\mathbf{w}|] < t$ **then**
        return $|\mathbf{w}|$
      else return None
    if $\exists k : (k-1)w_1 \leq \frac{W-kw_1}{m-1} \leq kw_1 < t \wedge k \leq \#(w_1)$ **then**
      return k
    return None

procedure SATISFIES($\mathbf{a}, P = SM \wedge Cr$)
  for $i \in [n]$ **do**
    if $v_{a_i} > v_m + w_i$ **then** ▷ Fails Cr
      return None
  return True

$SM \wedge Eq \wedge Cr$-Makespan

Since the algorithm for $SM \wedge Cr$-MAKESPAN produces a single feasible candidate solution (if it exists), we simply need to additionally check for Eq in the final part of the algorithm. This can be done in $O(n)$ by checking whether all players with the same weight incur the same cost.

procedure FIRST_K($P = SM \wedge Eq \wedge Cr, t, \mathbf{w}, m$)
  if $\exists k : (k-1)w_1 \leq \frac{W-kw_1}{m-1} \leq kw_1 \leq t\}$ **then**
    if $k \leq \#(w_1)$ **then**
      return k
    else
      return None
  else
    return None

procedure SUBSEQUENT_K($P = SM \wedge Eq \wedge Cr, t, \mathbf{w}, m, \mathcal{I}$)
  Let $\mathcal{I}$ be the weight of players in previous resource
  if $\mathcal{I} = w_1$ **then** ▷ Load must be $\leq t$
    if $m = 1$ **then** ▷ Final resource
      if $w_1 = w_{|\mathbf{w}|}$ **then**
        return $k = |\mathbf{w}|$
      else
        return None
    if $\exists k : (k-1)w_i \leq \frac{W-kw_i}{m-1} \leq kw_i \leq t \wedge k \leq \#(w_1)$ **then**
      return k
    else
      return None
  else ▷ Load must be $< t$
    if $m = 1$ **then** ▷ Final resource
      if $w_1 = w_{|\mathbf{w}|}$ and $W[1 : |\mathbf{w}|] < t$ **then**
        return $|\mathbf{w}|$
      else
        return None
    if $\exists k : (k-1)w_i \leq \frac{W-kw_i}{m-1} \leq kw_i < t \wedge k \leq \#(w_1)$ **then**
      return k
    else
      return None

procedure SATISFIES($\mathbf{a}, P = SM \wedge Eq \wedge Cr$)
  for $i \in [n]$ **do**
    if $v_{a_i} > v_m + w_i$ **then** ▷ Fails Cr
      return None
  for $i \in [n-1]$ **do**
    if $w_i = w_{i+1} \wedge v_{a_i} \neq v_{a_{i+1}}$ **then** ▷ Fails Eq
      return None
  return True

Proof of Theorem 23

Having introduced all the relevant subroutines, I now provide a formal proof to Theorem 23.

Theorem 23. *P-Makespan and P-Satisfiability can be solved in time $O(n \log n)$ for $P \in \{SM, SM \wedge Eq, SM \wedge Cr, SM \wedge Eq \wedge Cr\}$.*

Proof. I show that Algorithm 1, combined with the relevant subroutines, solves P-Makespan for each property P stated in the theorem (we can consider P-Satisfiability to be a special case of P-Makespan where $t = \infty$). I do so by showing that the algorithm terminates, is sound (i.e., if it returns an agreement, then this agreement has the required property), is complete (i.e., if an agreement with the required property exists, then such an agreement is returned by the algorithm), and runs in time $O(n \log n)$.

Termination: In each case of P, each iteration of Subsequent_k assigns some players to a resource, or terminates if no feasible k is found. Since there are finitely many players and resources, the algorithm always terminates.

For soundness, completeness, and running time, let us consider each case of P:

Case $P = SM$:

Soundness: By construction, at each iteration of the assignment process, SM is maintained by ensuring that consecutive resources have non-increasing loads if players across resources have the same weights, or decreasing loads if players across resources have different weights. And since players are ordered in non-increasing order of weight, SM is transitive across resources. The makespan threshold is maintain by ensuring that the first (and thus most-loaded) resource has load at most t.

Completeness: Suppose for contradiction that the algorithm does not return an agreement when a solution exists, which we call $\mathbf{a}$. We know that $\mathbf{a}$ must be contiguous due to Lemma 21. Since the algorithm does not return an agreement, it had run out of available resources to assign players to. This implies $\mathbf{a}$ must have assigned more players to some resources than the algorithm. However, no additional

player could be assigned to the first resource due to the makespan constraint, and no additional player could be assigned to each subsequent resource due to the SM constraint.

Running time: Because each iteration of assigning players to the next resource takes time $O(k)$ to find the maximum feasible candidate, where k sums to n across iterations, the assignment part of the algorithm is in $O(n)$. This leaves the sorting of players as the dominant running time, at $O(n \log n)$.

Case $P = SM \wedge Eq$**:**

Soundness: Similar to case $P = SM$, by construction, if the algorithm outputs an agreement, then the agreement is of makespan at most t and satisfies SM. Moreover, the additional assignment rule ensures that all players of the same weight incur the same cost, and thus satisfying Eq.

Completeness: Suppose for contradiction that the algorithm does not return an agreement when a solution exists, which we call $\mathbf{a}$. Because the algorithm only fails to return an agreement if it runs out of resources while there are still players to be assigned, it must be the case that $\mathbf{a}$ assigns more players to some resource than the algorithm. This implies that one of the conditions for selecting k is breached, and thus $\mathbf{a}$ cannot satisfy $SM \wedge Eq$.

Running time: For each iteration of players assignment, the candidate k can be computed by finding the largest divisor of $\#(w_i)$ that is below the threshold, which takes at most $O(\sqrt{\#}(w_i))$ steps. Since $n = \sum_{i \in \mathrm{supp}(\mathbf{w})} \#(w_i)$, the total number of operations across all iterations is at most $O(n)$ (if $\#(w_i) = 1$ for all $w_i \in \mathrm{supp}(\mathbf{w})$). This leaves the sorting of players as having the dominant running time, at $O(n \log n)$.

Case $P = SM \wedge Cr$**:**

Soundness: By construction, if the algorithm outputs an agreement, then its makespan is at most t (this is an explicit constraint on the first assigned resource, and holds implicitly through the SM bound for subsequent assignments), and has been checked for $SM \wedge Cr$-satisfaction by the subroutine $\textsc{Satisfies}(\mathbf{a}, P = SM \wedge Cr)$.

Completeness: Suppose for contradiction that the algorithm does not return an agreement when a solution exists, which we call $\mathbf{a}$. Then $\mathbf{a}$, in its canonical

ordering, must assigned to some resource that does not correspond to the k players prescribed by the algorithm. Consider the first resource (in canonical ordering) that fails this. The assignment of players to this resource fails to satisfying some necessary condition of SM or Cr, or have makespan greater than t, and therefore $\mathbf{a}$ is not a solution to the problem.

Running time: Each iteration of assigning players to the next resource takes $O(1)$ operations, by checking if the candidate $k = \lceil W/mw_1 \rceil$ satisfies the required bounds, and thus the process of generating a candidate agreement takes time $O(n)$. This leaves the sorting of players as the dominant running time, at $O(n \log n)$.

Case $P = SM \wedge Eq \wedge Cr$**:**

Soundness and completeness: This is essentially the same as the argument for SM, but with the added check for Eq.

Running time: Compared to $SM \wedge Cr$, the additional check for Eq is in time $O(n)$, which again leaves the sorting of players as the dominant running time, at $O(n \log n)$. □

4.2.5 $WOE \wedge SM$

For completeness, I briefly discuss properties that entail both WOE and SM.

For $WOE \wedge SM$, we can base our subroutines on those for SM, with the added requirement of WOE between players in consecutive resources.

A similar argument applies for $WOE \wedge SM \wedge Eq$.

For $WOE \wedge SM \wedge Cr$ and $WOE \wedge SM \wedge Eq \wedge Cr$, we can use the corresponding First_k and Subsequent_k of either the WOE version or the SM version, and simply check for satisfaction of the other condition in the subroutine Satisfies.

This is sufficient due to the uniqueness of a satisfying agreement, if one exists. In all cases, a solution is found in time $O(n \log n)$.

4.2.6 Weak Monotonicity

P-Makespan when P contains WM is generally not tractable, except for the special cases of WM-Satisfiability and $WM \wedge Eq$-Satisfiability, where the simple

agreement of placing all players on the same resource trivially satisfies WM and Eq. Discussion on the intractability of WM-Makespan is contained in the next section.

4.3 Intractable Properties

Finally, I discuss properties P for which P-Makespan is intractable. In contrast to the previous section, these problems are more difficult because a satisfying agreement of the lowest makespan may not be contiguous, as the example below shows for the case of WM.

Example 24. Consider the instance with $\mathbf{w} = (4, 3, 2, 1)$ and $m = 2$. The agreement $\mathbf{a}$ given by $l(\mathbf{a}) = \{\{4, 1\}, \{3, 2\}\}$ is of the lowest makespan of 5, and satisfies WM since all players incur the same cost. It is, however, not contiguous. The makespan-optimal, contiguous WM-agreement has the load distribution $\{\{4, 3\}, \{2, 1\}\}$, with a makespan of 7.

Our reductions are from the known NP-hard decision problem, Partition [71], which asks whether a multiset of positive integers can be evenly divided into two sets.

Problem: Partition
Input: A vector $\mathbf{s} = (s_1, ..., s_n)$ of positive integers
Task: Find a subset $S \subset [n]$ such that $\sum_{i \in S} s_i = \sum_{i \in [n] \setminus S} s_i$, or decide that no such subset exists.

The key of our reduction is to set $S = \sum_i s_i$, and to define the reduced instance by $\mathbf{w} = (\mathbf{s}, S + 1, S + 1) \in \mathbb{Z}_{>0}^{n+2}$ and $m = 2$. For makespan minimization, I additionally set $t = \frac{3}{2}S + 1$. Note that hardness for satisfiability of P immediately implies hardness of makespan minimization (by setting t to be sufficiently large) but our second result also covers properties for which we can efficiently solve P-Satisfiability.

Theorem 25. *Deciding whether a P-agreement exists is NP-complete for any property $P \in \{Eq \wedge Cr, WM \wedge Cr, WM \wedge Eq \wedge Cr\}$.*

Proof. Consider the following reduction from PARTITION to the P-SATISFIABILITY problem. Let $S = \sum_i s_i$, and let the instance of the P-SATISFIABILITY problem be $\mathbf{w} = (\mathbf{s}, S+1, S+1)$, and $m = 2$. I show that if a partition exists (i.e., s can be partitioned into s' and $s \setminus s'$ of equal sums), then an agreement with load distribution $\{s' \cup \{S+1\}, s \setminus s' \cup \{S+1\}\}$ satisfies P, and conversely, if an agreement with load distribution $\{s' \cup \{S+1\}, s \setminus s' \cup \{S+1\}\}$ satisfies P, then a partition exists.

Case $P = Eq \wedge Cr$**:**

($\implies$) Clearly, if a partition exists, then the agreement places the same load $\sum_{i \in s'} s_i + (S+1)$ on both resources. This implies that all players incur the same cost, so that no player has a credible deviation, and all players of the same weight incur the same cost, so the agreement is $Eq \wedge Cr$-satisfying. ($\impliedby$) Consider two cases: if the two players of weight $S+1$ share the same resource, then the total load on the other resource cannot exceed S, so that both players of weight $S+1$ would prefer to be assigned to another resource, which fails Cr, so this cannot be the case. If the two players of weight $S+1$ are assigned to different resources, then they must incur the same cost, meaning that the other players are assigned evenly between the two resources. And thus a partition exists.

Case $P = WM \wedge Cr$**:**

($\implies$) Clearly, if a partition exists, then the agreement places the same load $\sum_{i \in s'} s_i + (S+1)$ on both resources. This implies that all players incur the same cost, which is $WM \wedge Cr$-satisfying. ($\impliedby$) Consider two cases: If the two players of weight $S+1$ share the same resource, then by the same argument as above, Cr fails, so this cannot be the case. If the two players of weight $S+1$ are assigned different resources, then the load on the two resources must be equal, otherwise WM would not hold between any player $i \in S$ from the heavy-load resource and the $S+1$ player on the light-load resource. This means that players in S are divided evenly between the two resources. And thus a partition exists.

Case $P = WM \wedge Eq \wedge Cr$**:**

($\Longrightarrow$) Same argument as above. ($\Longleftarrow$) Consider two cases: If the two players of weight $S+1$ share the same resource, then by the same argument as above, Cr fails, so this cannot be the case. If the two players of weight $S+1$ are assigned different resources, then the load on the two resources must be equal in order to satisfy Eq. This means that players in S are divided evenly between the two resources. And thus a partition exists.

In each of the above cases, the P-SATISFIABILITY problem is also in NP: given a candidate solution, we can check the polynomial number of constraints imposed by these conditions, each of which takes polynomial time. □

Theorem 26. *Deciding whether a P-agreement satisfying a given makespan bound exists is NP-complete for any property $P \in \{\top, Eq, Cr, Eq \wedge Cr, WM, WM \wedge Eq, WM \wedge Cr, WM \wedge Eq \wedge Cr\}$.*

Proof. Consider the following reduction from PARTITION to a P-MAKESPAN problem. Let $S = \sum_i s_i$, and let the instance to the P-MAKESPAN problem be $\mathbf{w} = (\mathbf{s}, S+1, S+1)$, $m = 2$, with parameter $t = \frac{3}{2}S + 1$. We want to show that if a partition exists, then the agreement with load distribution $\{s' \cup \{S+1\}, s \backslash s' \cup \{S+1\}\}$ has makespan at most t and satisfies P, and conversely, if the agreement with load distribution $\{s' \cup \{S+1\}, s \setminus s' \cup \{S+1\}\}$ has makespan $\leq t$ and satisfies P, then a partition exists.

($\Longrightarrow$) If a partition exists, then the agreement places the same load $S/2+(S+1)$ on both resources, so the makespan is at the decision threshold t. Moreover, all player incur the same cost, so it is easy to verify that $P \in \{\top, Eq, Cr, Eq \wedge Cr, WM, WM \wedge Eq, WM \wedge Cr, WM \wedge Eq \wedge Cr\}$ is satisfied.

($\Longleftarrow$) If an agreement meets the threshold, then it must be the case that the two players of weight $S+1$ are assigned to different resources (otherwise the threshold would be breached), and that the players in s are split evenly across the two resources, which means that a partition exists.

Since each of the conditions above can be checked in polynomial time, each of the corresponding P-SATISFIABILITY problems are also in P. □

Note that an agreement need not be unique if it exists. For example, the two agreements $\mathbf{a}$ and $\mathbf{a}'$ where $l(\mathbf{a}) = \{\{2,1,1\},\{2,1,1\}\}$ and $l(\mathbf{a}') = \{\{2,2\},\{1,1,1,1\}\}$ are clearly not equivalent, and both satisfy $WM \wedge Eq \wedge Cr$.

Finally, let us reflect on why the same reduction for NP-hardness does not work for properties containing WOE, SM, or EF. These properties are not satisfied whenever players of different weights (say $w_i < w_j$) are assigned to different resources $a_i \neq a_j$ of the same load $v_{a_i} = v_{a_j}$, because this implies that $v_{a_i} \not< v_{a_j}$ (which fails SM) and $v_{a_i} - w_i > v_{a_j} - w_j$ (which fails WOE and EF). A Yes-instance of Partition would, therefore, not result in a Yes-instance of P-Makespan under the same reduction.

4.4 Conclusion

In the context of equilibrium refinement, the problem of finding a 'fair' Nash equilibrium can be phrased as solving the P-Satisfiability problem where P contains Cr and some additional fairness properties. This problem turned out to be tractable for a large number of fairness classes. Beyond equilibrium refinement, this chapter shows that the intractability of scheduling on identical machines is lifted when imposing additional, desirable constraints. The reason is that these constraints impose important structure to the solutions (i.e., contiguity), which sufficiently reduces the search space.

5
Communication Partition

In Part I of this thesis (Chapter 3 and Chapter 4), I discussed the conditions for a set of players to reach a coordinated agreement that is credible, efficient, and fair (for a range of definitions for fairness). We also found that for a given set of players, such an agreement may not exist. How might we resolve this problem, when no satisfying agreement can be reached between a set of players?

In the following two chapters (Part II of this thesis), I depart from the assumption of global communication, where all players communicate together to reach an agreeable outcome, and investigate the case of local communication, where players are partitioned into coalitions, and can only form agreements with players within the same coalition. As I will show, while a solution to the P-SATISFIABILITYproblem may not exist for some property P, we can always find a partition that ensures that all players reach a satisfying agreement within their coalition. As we have already seen, the room sharing dilemma introduced in Chapter 1 can be resolved when two out of the three colleagues coordinate their choices.

To narrow the range of conditions considered, I focus on the case where the properties of credibility (Cr), Pareto-optimality (PO) and envy-freeness (EF) are jointly satisfied. These conditions are collectively termed the *CrEF condition*, and an agreement satisfying these conditions is termed a *CrEF agreement*. (I have omitted PO in the label since, as I will show in Proposition 31, PO is implied by

Cr in SCGs.) I focus on the fairness condition of EF as it is the strongest condition that we considered in Part I (see Figure 3.1). Any existence guarantee unde EF therefore applies to other fairness conditions, namely Eq, WOE, SM and WM.

The remainder of this chapter sets out the formal model for investigating this problem, and provide some intermediary results concerning its properties. I formally introduce the SCG and discuss two standard measures of social cost, the Utilitarian social cost and the Rawlsian social cost. These are generalisations of the strategic scheduling problem that we discussed in Part I.

I then introduce the communication partition, and discuss assumptions about players' behaviour within each coalition and their beliefs about the behaviour of others outside of their coalition. Based on these assumptions, I define the partition equilibrium induced by a communication partition. Finally, I discuss the worst-case analysis for a partition equilibrium, i.e., what is the worst outcome that can occur with positive probability under a partition equilibrium.

5.1 Singleton Congestion Games

A (weighted) singleton congestion game (SCG) is a game with m identical resources M, and n weighted players N, represented by $w = (w_1, ..., w_n)$, where $w_i \in \mathbb{N}_{>0}$ is the weight of player i sorted in non-increasing order. Players each choose a resource, resulting in an outcome $\mathbf{a} = (a_1, ..., a_n)$ where $a_i \in M$ is the resource choice of player i. For illustrative purposes, I also express an outcome in accordance with the load distribution on all the resources, e.g., $\{\{4, 1, 1\}, \{3, 2\}\}$ corresponds to the outcome of a 5-player, 2-resource game, where the players of weights 4, 1 and 1 choose one resource, and while the players of weights 3 and 2 choose another.

I use $v_x(\mathbf{a}) \coloneqq \sum_{i \in N: a_i = x} w_i$ to denote the total weight of players that choose resource x in outcome $\mathbf{a}$. Each player incurs a cost $f : \mathbb{N}_{>0} \to \mathbb{Q}_{>0}$ that is strictly increasing in the total weight of players that choose the same resource. That is, given some outcome $\mathbf{a}$, the cost to player i is $c_i(\mathbf{a}) \coloneqq f(v_{a_i}(\mathbf{a}))$. The game can be fully specified by $G = (w, m, f)$. Notice that the game is *strategy-symmetric* since all resources have the same cost function.

An SCG in which all players have the same weight reduces to an unweighted SCG. In this case we can normalise all player weights to 1, so that the weights placed on a resource is equivalent to the number of players choosing that resource.

5.2 Social Cost

There are two common measures of the social cost of an outcome $\mathbf{a}$: the total weighted cost incurred by all players $\bar{c}(\mathbf{a})$, and the highest cost incurred by any player $\hat{c}(\mathbf{a})$, where

$$\bar{c}(\mathbf{a}) = \sum_{i \in N} w_i \cdot c_i(\mathbf{a}) = \sum_{x \in M} v_x(\mathbf{a}) \cdot f(v_x(\mathbf{a})),$$

and

$$\hat{c}(\mathbf{a}) = \max_{i \in N} f(v_{a_i}(\mathbf{a})) = \max_{x \in M} f(v_x(\mathbf{a})).$$

They correspond to the utilitarian social welfare function [72] and the Rawlsian social welfare function [73] respectively.

Minimising $\bar{c}(a)$ or $\hat{c}(a)$ closely relates to the well-studied problem of identical machine scheduling (IMS) [52], where the goal is to partition a multi-set of natural numbers (time required to complete each task) as evenly as possible into a given number of bins (machines). The player weights correspond to the multi-set of natural numbers, and the resources correspond to the bins. Partitioning the numbers 'as evenly as possible' typically means one of three distinct objectives [53]:

- 'minimax': minimise the size of the largest bin;
- 'maximin': maximise the size of the smallest bin;
- 'min-gap': minimise the gap between the largest and the smallest bin.

To see that these are distinct objectives, consider the following example taken from [74].

Example 27. Consider the problem of partitioning the set of numbers $\{46, 39, 27, 26, 16, 13, 10\}$ into three bins. For the minimax objective, the solution is $\{\{27, 26\}, \{46, 16\}, \{39, 13, 10\}\}$ where the maximum bin size is 62. For the maximin objective, the solution is $\{\{46, 10\}, \{27, 16, 13\}, \{39, 26\}\}$, with the minimum bin size of 56. For the min-gap objective, the solution is $\{\{39, 16\}, \{46, 13\}, \{27, 26, 10\}\}$, with a gap size of 8.

Finding the optimal partition is NP-complete for each of these objectives [75]. A fourth objective that is less commonly discussed in the literature, 'min-var', minimises the variance of the bin sizes.

We can easily observe that finding a $\hat{c}$-optimal outcome in an SCG is equivalent to the minimax IMS problem: because the cost function is increasing, minimising the cost associated with the most-used resource is equivalent to minimise the size of the largest bin.

The equivalent IMS objective for the $\bar{c}$-optimal outcome is more nuanced and depends on the convexity of the cost function f. If f is linear, the cost associated with a resource of total weight v is v^2, and thus the $\bar{c}$ of an outcome is the sum of squares of the weights on each resource. Minimising this objective is equivalent to minimising the variance of the total weights on the resources, and so the problem is equivalent to the min-var IMS problem.

If f is 'highly' convex, e.g., $f(w) = e^w$, then the $\bar{c}$-optimal outcome tends to the solution of the IMS problem with lexicographic social welfare [76, 77]. Lexicographic social welfare is a refinement of the minimax objective, where among outcomes with the same minimax values, choose one with the minimum next-most-used resource, etc. This is due to the fact that cost associated with the most-used resource dominates the costs associated with all other resources.

In this thesis I shall primarily focus on $\hat{c}$-optimality due to its natural correspondence with the minimax objective of the IMS problem (i.e., complete all tasks in the shortest time) and its applicability to all increasing cost functions regardless of their convexity.

5.3 Player Partition and Communication

A partition π divides the set of players N into non-empty, mutually exclusive, and collectively exhaustive subsets, called 'coalitions'. To illustrate by way of an example, consider a game with players of weights $(1, 2, 3, 4)$. The notation $[1, 2, 3 \mid 4]$ represents a partition where the players of weights 1, 2 and 3 form a communicating coalition, while player of weight 4 does not communicate with any other player. The partition where all players form a single coalition (the *grand coalition*) is represented by $[1, 2, 3, 4]$, and the partition where each coalition consists of a single player (the *singleton partition*) is represented by $[1 \mid 2 \mid 3 \mid 4]$. A partition is assumed to be common knowledge to all players.

An SCG, G, augmented by a communication partition is specified by $G^\pi = (w, m, f, \pi)$ and played over two phases. In the communication phase, players within a coalition communicate with each other for an indefinite period of time. Once players are satisfied with their communication, the game proceeds to the action phase, where players choose their actions simultaneously, unbound by any agreement reached during the communication phase. Because communication allows for some players' actions to be correlated, I use a correlated strategy profile s to describe the players' collective strategy, where $s \in \Delta M^n$ is a probability distribution on the set of outcomes M^n, and $s(\mathbf{a})$ represents the probability of the outcome $\mathbf{a}$ occurring under this strategy profile.

I do not attempt to model how players communicate, nor prescribe a protocol that governs such communication. Instead I take an axiomatic approach which assumes that players within a coalition would agree to a pure action-profile (an agreement) $\mathbf{a}_C \in M^{|C|}$ if certain conditions are met, given some shared belief about the behaviour of players outside of their coalition, which I discuss below.

5.3.1 Inter-coalition beliefs

While players within the same coalition can communicate their intentions (and assess whether a proposed agreement is credible, efficient and fair), players across

different coalitions do not communicate with another. What beliefs should they hold about the behaviour of others?

With identical resources, any agreement reached by some coalition C will have a large number of symmetric agreements, achieved by permuting the labelling of resources. Due to this, I assume players outside of C respect the *principle of indifference*, where they attribute equal probability to all symmetric agreements reachable in C. The principle of indifference is the rule that in the absence of any distinguishing features between a set of feasible and mutually exclusive outcomes, one should attribute equal probability to each outcome.

In our three colleagues example where only Adam and Bob can communicate, the principle of indifference states that Adam and Bob should attribute probability $1/2$ to each outcome where Charlie chooses one of the two rooms, due to the fact that the two rooms are identical. Similarly, Charlie may reason that Adam and Bob would reach the agreement whereby they choose different rooms, but not know which person would choose which room. The principle of indifference states that Charlie should attribute probability $1/2$ to each of the two symmetric agreements. Notice that these probabilities reflect the epistemic uncertainty of each player about the behaviour of others; they do not mean that the players actually make stochastic choices.

Notice also that the two rooms (or resources) need not be actually identical; they need only to 'appear' identical to the players. That is, even if the rooms were different but players do not know how they differ, or which is the more desirable room, the principle of indifference states that they should attribute equal probability to each room.

In strategy-symmetric games such as the SCG, the principle of indifference generates a common belief for players in C about the actions of all players outside of C. I write $-C \coloneqq N \setminus C$ for the set of players outside of C, and $\mathbf{a}_{-C} \coloneqq \times_{i \in C} a_i \in M^{|-C|}$ as the joint action of player outside of C. A common belief μ over actions of players in $-C$ is a probability distribution over $M^{|-C|}$, written as $\mu \in \Delta(M^{|-C|})$.

5. Communication Partition

A belief that respects the principle of indifference is one where $\mu(\mathbf{a}_{-C}) = \mu(\mathbf{a}'_{-C})$ for any two symmetric action-profiles $\mathbf{a}_{-C}, \mathbf{a}'_{-C} \in M^{|-C|}$.

Under this natural assumption, the expected cost function g_μ faced by players within a coalition is both increasing and symmetric across resources, as I demonstrate below.

Let $\mathbf{a}_C$ be the joint-action chosen by players in C, and $v_x(\mathbf{a}_C)$ be the total weight of players choosing some resource x under $\mathbf{a}_C$, and $v_x(\mathbf{a}_{-C})$ the total weight of players choosing x outside of C. The expected cost faced by a player that chooses x can be expressed as

$$\begin{aligned}\mathbb{E}_\mu\left[f(v_x(\mathbf{a}))\right] &\coloneqq g_\mu(v_x(\mathbf{a}_C)) \\ &= \sum_{u=0}^{\infty} \mu(u) \cdot f(u + v_x(\mathbf{a}_C))\end{aligned}$$

where $\mu : \mathbb{N} \to [0, 1]$ is the probability distribution on $v_x(\mathbf{a}_{-C})$ based on the player's belief.

Proposition 28. *g_μ is strictly increasing if f is strictly increasing; g_μ is weakly convex if f is weakly convex.*

Proof. If f is strictly increasing, then each summand in g is strictly increasing, and therefore g is strictly increasing. For convexity, consider for any $v \in \mathbb{N}$, $g_\mu(v+2) + g_\mu(v) - 2g_\mu(v+1) = \sum_{u=0}^{\infty} \mu(u)[f(u+v+2) + f(u+v) - 2f(u+v+1)]$. By Jensen's inequality, if f is weakly convex then each summand is non-negative, and thus $g_\mu(v+2) + g_\mu(v) - 2g_\mu(v+1)$ is non-negative, which implies g_μ is weakly convex. □

Remark 29. A natural consequence of Proposition 28 is that players within a coalition need not consider how players outside of the coalition behave: they face an increasing cost function g_μ on $v_x(\mathbf{a}_C)$ for any belief μ that respects the principle of indifference. Effectively, players within each coalition play a subgame of the augmented SCG, $(\{w_i : i \in C\}, m, g, \pi)$, where π is the grand coalition. we can thus treat each coalition as an independent coordination problem with global communication between its players.

5.3.2 Intra-coalition agreements

I introduced a number of fairness conditions in Chapter 3. For the remainder of this thesis, I restrict my focus on fairness to the envy-freeness (EF) condition. I also consider credibility (Cr) and Pareto-optimality (PO) as necessary conditions for a coalition of players to reach an agreement. I refer to these conditions collectively as the *CrEF conditions*.

In the context of the communication partition, these conditions, first introduced in Chapter 3, can be extended as follows.

Definition 30 (The CrEF conditions)**.** Consider a subset of weighted players $C \subseteq N$ and a common belief $\mu \in \Delta(M^{|-C|})$ about the behaviour of players outside of C, and an agreement $\mathbf{a}_C \in M^{|C|}$.

The agreement is *credible* (Cr) if for all $i \in C$ and $x \in M$,

$$\mathbb{E}_\mu[f(v_{a_i}(a_C))] \leq \mathbb{E}_\mu[f(v_x(\mathbf{a}_C) + w_i)].$$

The agreement is *Pareto-optimal* (PO) if $\nexists a'_C \in M^{|C|}$ such that for all $i \in C$,

$$\mathbb{E}_\mu[f(v_{a_i}(\mathbf{a}'_C))] \leq \mathbb{E}_\mu[f(v_{a_i}(a_C))]$$

and for some $j \in C$,

$$\mathbb{E}_\mu[f(v_{a_j}(\mathbf{a}'_C))] < \mathbb{E}_\mu[f(v_{a_j}(\mathbf{a}_C))].$$

The agreement is *envy-free* (EF) if for all $i, j \in C$,

$$\mathbb{E}_\mu[f(v_{a_i}(\mathbf{a}_C) - w_i)] = \mathbb{E}_\mu[f(v_{a_j}(\mathbf{a}_C) - w_j)].$$

The agreement is *covering* (Co) if

$$|C| \geq m \implies v_x(\mathbf{a}_C) > 0 \quad \forall x \in M,$$

$$|C| < m \implies |\{x \in M | v_x(a_C) > 0\}| = |C|.$$

5. Communication Partition

Credibility ensures that players will abide by the agreement (which are non-binding) due to the lack of a profitable unilateral deviation. This is derived from the notion of 'self-committing' messages in the cheap-talk literature ([78]). It requires the agreement to be a Nash equilibrium among players in the coalition given their common belief μ about the behaviour of others outside the coalition.

Pareto-optimality is the assumption that when players can freely coordinate, they will coordinate on an outcome on the Pareto-frontier. As we will see in Proposition 31, this is equivalent to the 'covering' condition for an SCG. The covering condition simply states that the maximum number of resources are used. In the case where $n < m$, this requires n resources to be used; in the case where $n \geq m$, it requires all m resources to be used.

Envy-freeness is a condition which ensures that no deadlock arises in the communication phase; a deadlock can arise if some players are envious of the assignment of another player and would prefer to swap the assigned resource with that player. As we have seen in Chapter 3, this is a very strong fairness condition, and for some coalition of players no envy-free agreement (and therefore no CrEF agreement) exists.

Proposition 31 sets out the relations between these conditions in SCGs.

Proposition 31. *An agreement a_C is covering if and only if it is Pareto-optimal; a covering agreement that is envy-free is credible; a credible agreement is a covering agreement.*

Proof. ($Co \implies PO$): Any changes to a covering agreement necessarily involves moving some players away from some resource x, and adding them to another resource y, thereby increasing the cost to players that already use resource y. So there can be no Pareto-improvement on a covering agreement.

($PO \implies Co$): If an agreement is not covering, then some player i can move from a shared resource x to an unused resource y. This lowers the costs of player i and all the other players that choose x, while the costs of all remaining players are unchanged. A non-covering agreement is therefore not Pareto-optimal.

($Co \wedge EF \implies Cr$): Consider a covering and non-credible agreement, so that there is a beneficial deviation by some player i from resource x to resource y, i.e. $v_x(a_C) > v_y(a_C) + w_i$. Because the agreement is covering, there is some player j that chooses y in a_C, so i must envy j and the agreement is not envy-free.

($Cr \implies Co$): A non-covering agreement is clearly not credible, as any players sharing a resource would deviate to an unused one. □

A corollary of Proposition 31 is that to find a CrEF agreement, we simply need to find a covering and envy-free agreement. We use this fact later to simplify the search for CrEF agreements.

It is also worth noting that while these conditions are defined in expected terms with respect to a common prior μ about the behaviour of players outside of C, we can in fact drop the expectation component $\mathbb{E}_\mu$ and replace f with the increasing function g_μ due to Proposition 28 and Remark 29.

The example of the three colleagues sharing two rooms, where they can all communicate together before choosing a room, can be modelled by an augmented SCG (w, m, f, π) with $w = (1, 1, 1)$, $m = 2$, f some increasing function, and π being the grand coalition. Consider the two types of agreement that may be reached.

The first is where all three players choose the same room. This is clearly not credible, as each player could benefit from a unilateral deviation to the empty room, nor is it Pareto-optimal, as any one player moving to the empty room would reduce the costs of all players.

The second is where two players share a room and the third does not. This is certainly credible (no player can benefit from unilaterally switching their room), and Pareto-optimal, but it is not envy-free, since either of the players sharing a room envies the player who does not share. Thus no CrEF agreement exists under the 'grand coalition' partition.

While not all coalitions $C \subseteq N$ can reach a CrEF agreement, we can always construct a partition π where all coalitions $C \in \pi$ can reach a CrEF agreement. Consider two trivial cases. The first is where all coalitions are singleton, and each player makes their own decision without communication. By the principle of

indifference, each player has no information about the choices of others, and since all resources are identical, each player is indifferent between them. This results in each player choosing one of the resources. From the perspective of an outsider, each player is playing a uniformly mixed strategy, resulting in a strategy profile that is the mixed-strategy Nash equilibrium.

Another less trivial case is one where each coalition is of size $\leq m$. In each coalition, any agreement where no two players share the same resource is a CrEF agreement. From the perspective of an outsider, we would expect a correlated strategy profile to be played where no two players within the same coalition chooses the same resource.

5.3.3 Partition equilibrium

When every coalition within a partition π reaches a CrEF agreement, they result in a correlated strategy profile $s \in \Delta M^n$ where $\sum_{a \in M^n} s(a) = 1$, which is a probability distribution on all outcomes of the game, M^n. This probability distribution reflects the epistemic uncertainty of an outsider: while each coalition reaches a deterministic agreement, the probability distribution represents the epistemic uncertainty over which of the symmetric agreements is reached within each coalition. Let $\mathbf{a}_C \in M^{|C|}$ be an agreement among players in coalition C, and let $s(\mathbf{a}_C)$ denotes the marginal probability that $\mathbf{a}_C$ is played by players in C under the correlated strategy profile s. Two agreements $\mathbf{a}_C$ and $\mathbf{a}'_C$ are symmetric if there exists some relabelling of resources $\psi : M \to M$ such that $\psi(\mathbf{a}_C) = \mathbf{a}'_C$.

Definition 32 (Partition equilibrium)**.** A correlated strategy profile s of an augmented game G^π is a partition equilibrium if for all $C \in \pi$, $s(\mathbf{a}_C) > 0$ if and only if $\mathbf{a}_C$ is a CrEF agreement, and for any two symmetric CrEF agreements $\mathbf{a}_C$ and $\mathbf{a}'_C$, $s(\mathbf{a}_C) = s(\mathbf{a}'_C)$.

If s is a partition equilibrium of G^π, we also say that the partition π induces the partition equilibrium s, or that s is a π-induced partition equilibrium of the base game G.

5. Communication Partition

For a correlated strategy profile s to be a partition equilibrium, the definition requires that each coalition reaches a CrEF agreement, each CrEF agreement has a positive marginal probability of occurring (to an outside observer), and that the same marginal probability must be attributed to all CrEF agreements that are symmetric to each other.

Notice that the definition does not specify how we should assess the relative probabilities of two non-symmetric CrEF agreements within a coalition. As I later show in Theorem 45, it turns out that all CrEF agreements are symmetric within a coalition (if they exist), so they have the same marginal probability of occurring in a partition equilibrium.

Remark 33. Notice that the credibility requirement implies that in SCGs, every partition equilibrium is a Nash equilibrium. From this perspective, the set of partition equilibria inducible by communication partitions can be seen as a refinement of the Nash equilibria of the game.

Let us return to the example of the three colleagues sharing two rooms, and consider the partition equilibrium induced by Adam and Bob discussing which rooms to choose, i.e., $\pi = [\text{Adam}, \text{Bob} \mid \text{Charlie}]$. First consider the CrEF agreements that can be reached by Adam and Bob, which are the two agreements where they choose different rooms (either Adam the first room and Bob the second room, or Bob the first room and Adam the second room). While Adam and Bob know which of the two CrEF agreements they've reached, from the perspective of an outsider, or Charlie, the two agreements are symmetric, and should therefore be given the same marginal probability of 1/2.

Similarly, Charlie does not communicate with any other player before making his choice, so Adam, Bob, and any other outsider should attribute probability 1/2 to Charlie choosing either room. We are thus left with a partition equilibrium where four outcomes have equal and positive probability of occurring: Adam and Charlie share the first room; Bob and Charlie share the first room; Adam and Charlie share the second room; Bob and Charlie share the second room.

5.4 Worst-case Analysis

To assess the efficiency of a partition equilibrium, one can consider its expected cost or its worst-case cost. I focus on the latter for two reasons: the worst-case is simple to calculate, and is particularly important in practical applications where excessive cost can have catastrophic consequences, such as causing disruptive shutdowns or permanent damage to the system.

The *worst-case outcome* $\hat{a}(s)$ is the highest-cost outcome that arises with positive probability under the correlated strategy profile s, i.e.,

$$\hat{a}(s) \coloneqq \operatorname*{argmax}_{a:s(a)>0} \hat{c}(a) = \operatorname*{argmax}_{a:s(a)>0} \left[\max_{x \in M} f(v_x(a))\right]$$

Since f can be any arbitrarily increasing function, it is more useful to consider the makespan of the worst-case outcome, $\max_{x \in M} v_x(\hat{a}(s))$. This *worst-case makespan* is comparable to the makespan of the IMS problem. For a given partition, I use the ratio of the *worst-case makespan* of the induced partition equilibrium to the optimal makespan as a measure of its efficiency.

More generally, I use the ratio of the worst-case makespan of the optimal partition to the optimal makespan to describe the efficiency of the partition mechanism. This ratio is known as the approximation of the mechanism [79] and represents how close the mechanism comes to inducing the optimal outcome. [1]

In the previous three colleagues example with the partition [Adam, Bob | Charlie], the worst-case makespan is 2, since at most two colleagues end up choosing the same room. Contrast this with an alternative partition [Adam | Bob | Charlie], where no player communicates with each other. In this case, the partition equilibrium is one where each player chooses each room with probability 1/2 independently of each other, which corresponds to the MSNE of the base game. The worst-case makespan of this partition equilibrium is 3, since there is a positive probability that the three colleagues all choose the same room.

[1] A similar concept is the *price of anarchy* [56], which is the cost ratio of the worst-case equilibrium to the socially optimal outcome. The approximation is a more appropriate concept here as we are comparing the worst-case outcome of a unique equilibrium to the socially optimal outcome.

5. Communication Partition

In this particular game, the first partition induces a partition equilibrium where the worst-case outcome has an approximation of $2/2 = 1$, i.e., it always induces the social-optimal outcome. The second partition induces a partition equilibrium where the worst-case outcome has an approximation of $3/2$.

6
Optimal Partition

In Chapter 5 I introduced the notion of a communication partition, and set out how a partition augmented game is played among a set of players. I defined the partition equilibrium, and discussed its existence in partition augmented games. I also discussed the worst-case outcome of a partition equilibrium, and defined an optimal partition as a partition that induces a partition equilibrium with the best worst-case outcome.

In this chapter, I characterise the optimal partition under this worst-case measure. That is, given a base game G, what is the partition that induces the best worst-case outcome? We first look at the special case where all players are of equal weight ('unweighted players'), then study the general case where players have unequal weights ('weighted players'), and provide an approximation for the optimal partition.

6.1 Unweighted Players

When players are unweighted, we can represent the load on a resource by the number of players choosing such a resource, i.e., $v_x(\mathbf{a}) = |\{i \in N : a_i = x|$. [1]

[1]Such games are not only strategy-symmetric, but also player-symmetric, since players are both anonymous and have identical utility functions.

6.1.1 Optimal outcomes

Before considering what the optimal partition looks like, let us first consider what the optimal outcome looks like. I first consider an agreement where players are distributed among resources as evenly as possible, and show that it is socially optimal with respect to $\bar{c}$ and $\hat{c}$.

Definition 34 (Even distribution)**.** A joint action $\mathbf{a}_C \in M^{|C|}$ of a set C of unweighted players is *evenly distributed* if for all $x, y \in M$, $|v_x(\mathbf{a}_C) - v_y(\mathbf{a}_C)| \leq 1$.

The requirement that the differences in the number of players choosing different resource is at most 1 means that players are spread as evenly as possible among all resources.

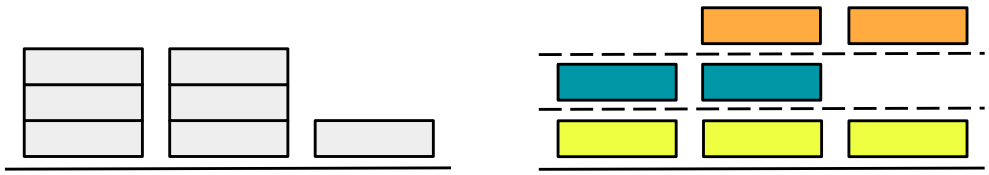
Lemma 35 (Optimal outcomes)**.** *An outcome $\mathbf{a} \in M^n$ is $\bar{c}$-optimal if and only if $\mathbf{a}$ is evenly distributed. An outcome $a \in M^n$ is $\hat{c}$-optimal if $\mathbf{a}$ is evenly distributed.*

Proof. ($\Longrightarrow$, $\bar{c}$-optimality) Consider an outcome a that is not evenly distributed, i.e., there exists some $x, y \in M$ such that $n_x(\mathbf{a}) - n_y(\mathbf{a}) \geq 2$ (for brevity I drop the reference to $\mathbf{a}$ in the remainder of this proof). For $\bar{c}$, the costs associated with these resources are $n_x f(n_x) + n_y f(n_y)$. If one of the players choosing x changes their choice to y, then the costs associated with these resources become $(n_x - 1)f(n_x - 1) + (n_y + 1)f(n_y + 1)$. The effect of this change can be expressed as

$$\begin{aligned}&(n_x - 1)[f(n_x) - f(n_x - 1)] - (n_y + 1)[f(n_y + 1) - f(n_y)] \\ &+ f(n_x) - f(n_y)\end{aligned}$$

By assumption, $n_x - 1 \geq n_y + 1$, and by the convexity of f, $f(n_x) - f(n_x - 1) \geq f(n_y + 1) - f(n_y)$, and thus the sum of the first two terms is non-negative. Since f is increasing, $f(n_x) > f(n_y)$, and the sum of the final two terms is positive. The change therefore decreases $\bar{c}$, so $\mathbf{a}$ cannot be $\bar{c}$-optimal.

($\Longleftarrow$, $\bar{c}$-optimality & $\hat{c}$-optimality) Notice that all outcomes where players are evenly distributed have the same social costs $\bar{c} = lf(y + 1) + (n - l)f(y)$ and $\hat{c} = f(y + 1)$, where $y = \lfloor \frac{n}{m} \rfloor$, and $l = n \bmod m$. Since these are the optimal social costs, all evenly distributed outcomes are $\bar{c}$-optimal and $\hat{c}$-optimal. □



(a) An uneven yet $\hat{c}$-optimal outcome

(b) An unbalanced partition that always induces $\hat{c}$-optimal outcomes

Figure 6.1: Examples of 7-player 3-resource outcomes. Players are represented as blocks and resources as columns. A partition is represented by dotted lines and coloured coalitions.

Note that a $\hat{c}$-optimal outcome need not be evenly distributed. See for example Figure 6.1(a).

6.1.2 Optimal partition

I now show that under the principle of indifference, efficient outcomes can be induced by partitioning players in a balanced way.

Definition 36 (Balanced partition)**.** A partition π on N is *balanced* if at most one coalition has size $|C| < m$, with the remaining coalitions having sizes such that $m \mid |C|$.

In other words, a balanced partition is the result of partitioning players into coalitions of sizes that are divisible by m, with at most one coalition of 'remainders' that is smaller than m. Naturally, if the total number of players is divisible by the number of resources, i.e., $m \mid n$, then a balanced partition π is one where every coalition is divisible by m.

We can extend the definition of optimality from outcomes to correlated strategy profiles:

Definition 37. A correlated strategy profile $s \in \Delta(M^n)$ is $\hat{c}$ (or $\bar{c}$)-optimal if its worst-case outcomes is $\hat{c}$ (or $\bar{c}$)-optimal.

In other words, a correlated strategy profile is $\hat{c}$ (or $\bar{c}$)-optimal if all the outcomes that occur with positive probability are $\hat{c}$ (or $\bar{c}$)-optimal.

We can now establish, for each coalition within a balanced partition (i.e., $m \mid |C|$, $|C| < m$, or neither), the conditions for an agreement to be credible, Pareto-optimal, and envy-free.

Lemma 38. *An agreement $\mathbf{a}_C$ reached by a coalition of players $C \in \pi$ where $m \mid |C|$ satisfies the CrEF conditions if and only if each resource is chosen by exactly $\frac{|C|}{m}$ players.*

Proof. ($\implies$) By Proposition 28, players in C need only consider the actions of other players within the coalition. Suppose an agreement $\mathbf{a}_C$ is such that some resource x is chosen by more players than some other resource y, i.e., $n_x(\mathbf{a}_C) > n_y(\mathbf{a}_C)$. Then $f(n_x(\mathbf{a}_C)) > f(n_y(\mathbf{a}_C)))$, and a player choosing x would envy a player choosing y, so $\mathbf{a}_C$ is not envy-free.

($\impliedby$) Suppose an agreement $\mathbf{a}_C$ is such that each resource is chosen by the same number of players. It is clear that such an agreement is envy-free (all players incur the same cost) and covering, and therefore by Proposition 31 it is also credible and Pareto-optimal. □

Lemma 39. *An agreement $\mathbf{a}_C$ reached by a coalition of players $C \in \pi$ where $|C| < m$ satisfies the CrEF conditions if and only if each resource is chosen by at most one player.*

Proof. ($\implies$) By Proposition 28, players in C need only consider the actions of other players within the coalition. Suppose an agreement $\mathbf{a}_C$ is such that some resource x is chosen by more than 1 player, and therefore there is some resource y that is not chosen by any player. Then the agreement is not covering, and by Proposition 31 not Pareto-optimal.

($\impliedby$) Suppose an agreement $\mathbf{a}_C$ is such that each resource is chosen by at most 1 player. It is clear that such an agreement is envy-free (all players incur the same cost $f(1)$) and covering, and therefore by Proposition 31 it is also credible and Pareto-optimal. □

Lemma 40. *A coalition of players $C \in \pi$ where $|C| > m$ and $m \nmid |C|$ cannot reach a CrEF agreement.*

Proof. For an agreement to be Pareto-optimal, it must be covering, and since $|C| > m$, all resources are used. Because $m \nmid |C|$, some resources must be shared by more players than others, and thus the envy-free condition cannot be satisfied. □

From Lemma 38 and Lemma 39, we can see that a balanced partition π induces a very particular feasible strategy profile, where each coalition with $m \mid |C|$ reach an agreement where all the resources are chosen by the same number of players, and the coalition with $|C| < m$ (if there is one such coalition) reach a covering agreement. I now show that this induced partition equilibrium is always efficient. Moreover, due to Lemma 40, a partition equilibrium is $\bar{c}$-efficient only if the partition which induces it is balanced.

Theorem 41 (Optimal partition)**.** *A π-induced partition equilibrium is $\bar{c}$-optimal if and only if π is balanced. A π-induced partition equilibrium is $\hat{c}$-optimal if it is balanced.*

Proof. ($\implies$, $\bar{c}$-optimality) By Lemma 38, Lemma 39 and Lemma 40, only partitions where for each $C \in \pi$, $m \mid |C|$ or $|C| < m$ can induce a partition equilibrium, so we need only consider such partitions. For such a partition to be unbalanced, there must exist distinct $C, D \in \pi$ such that $|C| < m$ and $|D| < m$, and π induces agreements in C and D that are evenly distributed. If $|C| + |D| \leq m$, then there is a positive probability (since players cannot coordinate across coalitions) that some resource x is chosen by two players, one from each coalition. If $|C| + |D| > m$, then there is a positive probability that some resource x is chosen by none of the players. In either case, there is a positive probability that the resulting outcome is not evenly distributed, and therefore the induced partition equilibrium is not $\bar{c}$-optimal.

($\impliedby$, $\bar{c}$-optimality and $\hat{c}$-optimality) If $m \mid n$, then a balanced partition consists of all coalitions of sizes divisible by m, so by Lemma 38 each coalition reaches an evenly distributed agreement, and all possible outcomes are evenly distributed. Therefore the induced partition equilibrium is $\bar{c}$-optimal and $\hat{c}$-optimal. If $m \nmid n$, then by Lemma 39 the coalition C where $|C| < m$ reaches an agreement where at

most 1 player chooses each resource. Combined with agreements reached by all other coalitions, all possible outcomes are again evenly distributed (since the difference in the number of players choosing each resource is at most 1), and therefore the induced partition equilibrium is $\bar{c}$-optimal and $\hat{c}$-optimal. □

Notice that a partition need not be balanced to induce a $\hat{c}$-optimal generalised strategy profile. For an example see Figure 6.1b.

6.1.3 Differential cost functions

A natural extension to consider is whether our results generalise to broader classes of SCG. In this section I briefly discuss the extent to which this analysis can be generalised to resources with differential cost functions.

When resources have different costs f_x for each $x \in M$, and that these costs are known to the players, we face two further issues concerning whether there exists a communication partition that induces efficient outcomes.

First, envy-freeness becomes an even stronger condition to satisfy. Consider a simple example of a game of two players $\{1, 2\}$ and 2 resources $\{x, y\}$, where $f_x(n) = 1.1 \cdot f_y(n)$. That is, resource x is a little more costly than y given the same number of users n. Let us first consider the case where the partition is the grand coalition. Clearly, the $\bar{c}$-optimal and $\hat{c}$-optimal outcome is for the players to choose different resources, and such an agreement would also be credible and Pareto-optimal. However, such an agreement could not be envy-free, since the player that is allocated x would envy the player that is allocated the cheaper resource y, and thus players might reach a deadlock. Alternatively, both players choosing the same resource (x or y) is clearly envy-free, but not credible nor Pareto-optimal.

Second, the strategy symmetry of the game is broken, and we can no longer rely on the principle of indifference to provide a common, symmetric prior. In the above example, if the partition is a singleton partition (i.e., players cannot communicate), then it is not clear how each player should model the behaviour of the other. Additional assumptions about how players behave (or more precisely, how players believe others behave) are required to analyse this problem.

Nevertheless, both issues fall away if we assume that all players are aware of the distribution of the cost functions among resources, but not the cost associated with each resource. As I showed in Proposition 28, a convex combination of increasing and convex functions is also increasing and convex, and the expected cost function is again identical across resources. Thus players will behave as if resources are identical. Our analysis on the efficient partition in SCG is therefore applicable to games with differential cost functions where only the distribution is known.

6.2 Weighted Players

Let us now consider the general case where players are weighted. I discuss the conditions under which a partition equilibrium exists, the uniqueness of partition equilibrium for a given partition, how to find the optimal partition (i.e., the partition that induces the partition equilibrium with the best worst-case makespan), and the ratio of the worst-case makespan of the optimal partition to the optimal makespan.

6.2.1 Existence and uniqueness of partition equilibrium

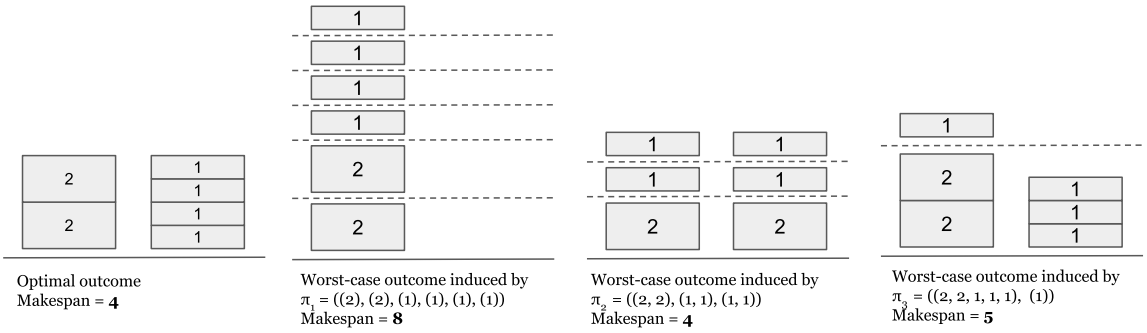


Figure 6.2: Examples of the worst-case outcome and the worst-case makespan of partition equilibria induced by different partitions, for the WSCG with $w = (2, 2, 1, 1, 1, 1)$ and $m = 2$. π_1 consists of singleton coalitions only; π_2 consists of type 1 coalitions only; π_3 consists of a type 1 coalition and a type 2 coalition.

Let us first consider how CrEF agreements can be reached in a coalition, for coalitions of sizes $|C| \leq m$ and $|C| > m$.

Lemma 42. *If a coalition C of size $|C| \leq m$ reaches a CrEF agreement $\mathbf{a}_C \in M^{|C|}$, then all players choose different resources, i.e., $a_i \neq a_j$ for all $i, j \in C$, $i \neq j$.*

Proof. This is a direct application of Proposition 31. A CrEF agreement must be covering, which for a coalition of size $|C| \leq m$ means that players choose different resources. Such an agreement is clearly envy-free. □

For a coalition of size $|C| > m$ to reach CrEF agreements, there are additional conditions on the membership of C that need to be met.

Lemma 43. *If a coalition C of size $|C| > m$ reaches a CrEF agreement $\mathbf{a}_C \in M^{|C|}$, then there exists some integer b such that $w_i \mid b$ for all $i \in C$, where each resource is shared by $\frac{b}{w_i} + 1$ players of the same weight w_i.*

Proof. By Proposition 31, we only need to consider covering agreements that are envy-free. Let us consider the envy-freeness condition when all resources are used, which is essentially that $v_{a_i} - w_i = v_{a_j} - w_j$ for all $i, j \in C$ (we can ignore the expectation over μ due to Proposition 28).

If $a_i = a_j$, i.e., two players choose the same resource, then $v_{a_i} = v_{a_j}$, and we have $w_i = w_j$. So all players that are assigned the same resource must have the same weight. If $a_i \neq a_j$, then the condition implies that there exists some value $b \coloneqq v_{a_i} - w_i$ that is constant across all $i \in C$ (or more precisely, across all resources $x \in M$). □

For simplicity we refer to any coalition of size $|C| \leq m$ as a 'type 1' coalition, and any coalition that can reach a CrEF agreement of size $|C| > m$ as a 'type 2' coalition.

Example 44 (Type 2 coalition)**.** Consider the following type 2 coalition, where C consists of players of weights $w = (3, 3, 1, 1, 1, 1)$, and $m = 2$. The agreement $\{\{3, 3\}, \{1, 1, 1, 1\}\}$ is a CrEF agreement, where the value of $b = 3$. We can easily verify that no player envies another: no player of weight 3 would prefer to swap their assigned resource with a player of weight 1 and vice versa.

Theorem 45 (Uniqueness of Partition Equilibrium)**.** *A partition π induces a partition equilibrium if and only if it consists of type 1 and type 2 coalitions only. If an induced partition equilibrium exists, then it is unique.*

Proof. For existence, we know from Lemma 42 and Lemma 43 that type 1 and type 2 coalitions are the only coalitions for which CrEF agreements can be reached. Therefore only partitions consisting of type 1 and type 2 coalitions can induce a partition equilibrium.

For uniqueness, notice that for any type 1 coalition, all CrEF agreements are strategy-symmetric (i.e., can transform from one to another by a permutation of resources), therefore any partition equilibrium would place the same marginal probability on each CrEF agreement. Similarly, Lemma 43 shows that in a type 2 coalition there is only one way of achieving a CrEF agreement (up to strategy-symmetry), therefore any partition equilibrium would place the same marginal probability on each such agreement. □

As I previously noted, there always exists a partition that induces a partition equilibrium: we can simply construct one consisting of type 1 coalitions. However, such a partition equilibrium may not be optimal (among all partition equilibria that can be induced by any partition), nor efficient (compared to the optimal outcome).

Examples of the worst-case outcomes of partition equilibria induced by different partitions are shown in Figure 6.2.

6.2.2 Optimal partition

Since CrEF agreements can only be reached within type 1 and type 2 coalitions, we need only consider partitions consisting solely of type 1 and type 2 coalitions when searching for an optimal partition.

First, I establish an optimal way of partitioning players into type 1 coalitions only, which I term the *optimal type 1 partition*.

Consider the worst-case outcome induced by a partition consisting solely of type 1 coalitions. Because each coalition is of type 1, the CrEF agreement reached

within each coalition is such that no two players are assigned the same resource. And thus, the worst-case outcome is one where all of the heaviest players within each coalition choose the same resource. Thus, the optimal type 1 partition is one that minimises the sum of weights of the heaviest player in each coalition. One way to achieve this is to first sort players into a non-increasing weight order, then sequentially partition every m players into a coalition, as I demonstrate below.

Lemma 46 (Optimal type 1 partition)**.** *The partition* $[w_1, \ldots, w_m \mid w_{m+1}, \ldots, w_{2m} \mid \lfloor w_{(\lfloor \frac{(n-1)}{m} \rfloor m)+1}, \ldots, w_n]$*, where players are sorted in a non-increasing weight order, is an optimal partition consisting solely of type 1 coalitions.*

Proof. The worst-case makespan of a partition equilibrium consisting only of type 1 coalitions is the makespan of an outcome where the leading player (i.e., player with highest weight) of all coalitions chooses the same resource. So minimising the worst-case makespan is equivalent to minimising the sum of the weights of the leading player of each coalition. The proposed partition therefore has a worst-case makespan of $\sum_{k=0}^{\lfloor (n-1)/m \rfloor} w_{1+km}$.

Consider an arbitrary partition π and the leading players in each coalition in π, where players are sorted in a non-increasing weight order. The first leading player must have weight equal to w_1, since the heaviest player is always a leading player. The second leading player must have weight $\geq w_{m+1}$, since there can be at most m players in the coalition consisting of the first leading player. Similarly, the third leading player must have weight $\geq w_{2m+1}$, and so on. Thus the worst-case makespan of π must be $\geq \sum_{k=0}^{\lfloor (n-1)/m \rfloor} w_{1+km}$. Since the proposed partition has a worst-case makespan $\sum_{k=0}^{\lfloor (n-1)/m \rfloor} w_{1+km}$, it is optimal. □

The following two examples illustrate that neither type 1 nor type 2 coalition is consistently more efficient than the other. This suggests that to find the optimal partition, we must consider all the ways players can be partitioned into type 1 and type 2 coalitions.

Example 47 (Type 2 coalition more efficient than type 1 partition)**.** Consider an SCG with $w = (2, 2, 2, 2, 1, 1, 1)$ and $m = 3$. The grand coalition, which is type 2, induces the outcome $\{\{2, 2\}, \{2, 2\}, \{1, 1, 1\}\}$ with $\hat{c} = 4$. If we instead partition players into the optimal type 1 partition $[2, 2, 2 \mid 2, 1, 1 \mid 1]$, we get the worst-case outcome $\{\{2, 2, 1\}, \{2, 1\}, \{2, 1\}\}$ with $\hat{c} = 5$.

Example 48 (Type 1 partition more efficient than Type 2 coalition)**.** Consider an SCG with $w = (4, 4, 1, 1, 1, 1, 1)$ and $m = 2$. The type 1 partition $[4, 4 \mid 1, 1 \mid 1, 1 \mid 1]$ induces the worst-case outcome $\{\{4, 1, 1, 1\}, \{4, 1, 1\}\}$ with $\hat{c} = 7$. In contrast, the partition consisting of a single type 2 coalition induces the outcome $\{\{4, 4\}, \{1, 1, 1, 1, 1\}\}$, with $\hat{c} = 8$.

The optimal partition can therefore be found by considering all partitions that consist solely of type 1 and type 2 coalitions. I am not aware of any search algorithms that can find the optimal partition without incurring a similar combinatoric blow-up.

6.2.3 Approximation

As discussed in Section 5.4, the ratio of worst-case makespan of the optimal partition to the makespan of the optimal outcome serves as a measure of the efficiency of the partition mechanism. I now show that the worst-case makespan of the optimal partition is a 2-approximation of the makespan of the optimal outcome, in contrast to the case of no communication (i.e., a singleton partition), where the worst-case makespan is an m-approximation of the makespan of the optimal outcome.

Given an SCG G, I use $\hat{\pi}(G)$ to denote its optimal partition, $\hat{\pi}_1(G)$ its optimal type 1 partition (the optimal partition that consists only of type 1 coalitions), $\hat{v}_\pi$ the worst-case makespan of a partition equilibrium induced by the partition π, and v^* the makespan of the optimal allocation.

Theorem 49. *For any SCG, $\hat{v}_{\hat{\pi}} \leq 2v^*$. That is, the optimal partition induces a partition equilibrium whose worst-case makespan is a 2-approximation of the optimal makespan.*

Proof. The proof consists of two parts. The first part bounds the makespan of the optimal partition equilibrium by that of the makespan of the optimal type 1 partition, and show that the optimal type 1 partition induces a worst-case outcome that is equivalent to the round-robin allocation of resources in the IMS problem. We can then use the known result from [52] that the makespan of the round-robin allocation is a 2-approximation of the makespan of the optimal allocation.

First notice that $\hat{v}_{\hat{\pi}_1} \geq \hat{v}_{\hat{\pi}}$, since the optimal partition must be at least as good as the optimal type 1 partition. We've seen in Example 48 that the equality holds for some game G, so the bound is tight.

Next notice that the worst-case makespan of the partition equilibrium induced by $\hat{\pi}_1$ is identical to the makespan of the round-robin allocation, where player i is allocated to resource $(i-1 \mod m)+1$ (players are indexed in non-increasing weight order). In both cases, the makespan is $\sum_{k=0}^{\lfloor (n-1)/m \rfloor} w_{1+km}$, which is v_1, the weights assigned to resource 1.

We now follow the same argument that proves the makespan of the round-robin scheduler is a 2-approximation of the optimal makespan. Clearly, the optimal makespan is lower-bounded by the average weight across resources $\frac{\sum_i w_i}{m}$ and the weight of the heaviest player w_1.

Let $\tilde{a}$ be the round-robin allocation. Absent w_1, the weights on resource 1 must be lower than the weight on resource m (with the lowest allocated weights), which is in turn upper-bounded by the average weight allocated to a resource. I.e., $v_1(\tilde{a}) - w_1 < v_m(\tilde{a}) \leq \frac{\sum_i w_i}{m}$. So $v_1(\tilde{a})$ is upper-bounded by $\frac{\sum_i w_i}{m} + w_1$.

Putting the inequalities together, we have $\hat{v}_{\hat{\pi}} \leq \hat{v}_{\hat{\pi}_1} = v_1(\tilde{a}) \leq \frac{\sum_i w_i}{m} + w_1$ and $v^* \geq \max(\frac{\sum_i w_i}{m}, w_1)$, which implies $\hat{v}_{\hat{\pi}} \leq 2v^*$.

□

Contrast this result to that of the game without a partition mechanism, which can be modelled as a game with a singleton partition. The induced partition equilibrium is the mixed-strategy Nash equilibrium where all players choose each resource with equal probability. Clearly, the worst-case outcome is one where all

players choose the same resource, with the worst-case makespan $\sum_i w_i$, which is an m-approximation of the optimal makespan v^*.

7 Conclusion

7.1 Summary

In this thesis, I have explored how communication between players can aid coordination in a congestion setting typical of scheduling and load-balancing problems. I have taken an axiomatic approach and studied the outcomes that can be achieved under various conditions of credibility, efficiency, and fairness.

In Part I of this thesis, I studied the computational complexity of finding a partition equilibrium (if it exists) for a combination of the axiomatic conditions. I found that only contiguous assignments satisfy some key fairness properties. Since the search space for contiguous assignments is significantly smaller than all possible assignments, finding a satisfying assignment turned out to be tractable for a wide range of fairness properties.

In Part II, I explored the impact of partitioning players into coalitions and studied the induced partition equilibrium under the conditions of credibility, efficiency, and envy-freeness. For unweighted players, an optimal partition is a balanced one. For weighted players, the optimal partition can consist of both type 1 and type 2 coalitions. Finally, the worst-case outcome of the optimal type 1 partition is a 2-approximation of the optimal outcome.

In contrast with the concept of Nash equilibrium, this work explicitly models the coordination between players and considers the correlated strategies that result. This can be seen as a way of selecting a Nash equilibrium through the choice of a communication partition, where the induced partition equilibrium is a particular Nash equilibrium of the base game. Conversely, this model can also be used to infer the latent communication partition that exists (e.g., in the case of collusive behaviour in the real world), based on the outcomes observed.

7.2 Future Directions

The structure of the communication partition is very generic and can be applied to any strategic-form games. In this thesis, I focused on a particular subset of games with specific measures of efficiency. The study of partition equilibrium can be extended in many directions.

On the measure of social efficiency, I mainly considered the social welfare function $\hat{c}$, with some cursory discussion on $\bar{c}$. In practical problems, the optimal partition with respect to $\bar{c}$ will depend on the exact shape of the cost function f. Studying the optimal partition for some standard cost functions (e.g. linear, logarithmic, quadratic, exponential) may yield useful, practical insight.

In this thesis I considered the worst-case makespan of the partition equilibrium. This is particularly meaningful when the goal is to avoid catastrophic outcomes in systems where exceeding a certain makespan threshold is highly risky or costly. An alternative objective that can be explored is to minimise the expected makespan of the partition equilibrium.

The cost to a player is modelled as proportional to the total weight assigned to the resource which the player has chosen. This is an appropriate assumption in systems where the cost of congestion on a resource is borne by all players using the resource. An alternative assumption, which is also studied in the scheduling literature, is to take the cost of a player as proportional to the completion time of their specific task, and thus a player whose task is scheduled earlier incurs a smaller cost. This alternative model brings additional complexity to the problem,

since players are coordinating not only on the resource assignment, but also the order in which tasks are processed on a resource.

I briefly discussed asymmetric strategies in Section 6.1.3, and noted that we can no longer rely on the indifference principle to model a player's epistemic uncertainty over the behaviour of those outside of their coalition. Further consideration on how to model the general case of asymmetric strategies can broaden the applicability of this model to a wide range of multi-agent systems.

Finally, further research on the computational complexity of finding the optimal partition, and the approximation of efficient algorithms, will strengthen the applicability of this model to real-world problems.

References

[1] Wei-Chen Lee, Alessandro Abate and Michael J. Wooldridge. 'Equilibrium Selection via Cheap-talk Partition: Extended Abstract'. In: *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems* (2025).

[2] Wei-Chen Lee, Alessandro Abate and Michael J. Wooldridge. 'Equilibrium Selection via Cheap-talk Partition: Extended Abstract'. In: *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems* (2025).

[3] Wei-Chen Lee, Alessandro Abate and Michael J. Wooldridge. 'Partition Equilibria in Weighted Singleton Congestion Games'. In: *Proceedings of the European Conference on Artificial Intelligence* (2025).

[4] Martin J Osborne and Ariel Rubinstein. *A course in game theory*. MIT press, 1994.

[5] Drew Fudenberg and Jean Tirole. *Game theory*. MIT press, 1991.

[6] J v. Neumann. 'Zur theorie der gesellschaftsspiele'. In: *Mathematische annalen* 100.1 (1928), pp. 295–320.

[7] Joseph Y. Halpern. 'Beyond Nash Equilibrium: Solution Concepts for the 21st Century'. In: *Lectures in Game Theory for Computer Scientists*. Ed. by Krzysztof R. Apt and ErichEditors Grädel. Cambridge University Press, 2011, pp. 264–290.

[8] John F Nash Jr. 'Equilibrium points in n-person games'. In: *Proceedings of the national academy of sciences* 36.1 (1950), pp. 48–49.

[9] Constantinos Daskalakis, Paul W. Goldberg and Christos H. Papadimitriou. 'The complexity of computing a Nash equilibrium'. In: *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing - STOC '06*. ACM Press, 2006. DOI: 10.1145/1132516.1132527. URL: https://doi.org/10.1145/1132516.1132527.

[10] Reinhard Selten. 'Spieltheoretische behandlung eines oligopolmodells mit nachfrageträgheit: Teil i: Bestimmung des dynamischen preisgleichgewichts'. In: *Zeitschrift für die gesamte Staatswissenschaft/Journal of Institutional and Theoretical Economics* H. 2 (1965), pp. 301–324.

[11] Reinhard Selten. 'Reexamination of the Perfectness Concept for Equilibrium Points in Extensive Games'. In: *International Journal of Game Theory* 4 (1975).

[12] Drew Fudenberg and Jean Tirole. 'Perfect Bayesian equilibrium and sequential equilibrium'. In: *Journal of Economic Theory* 53.2 (1991), pp. 236–260. DOI: https://doi.org/10.1016/0022-0531(91)90155-W. URL: https://www.sciencedirect.com/science/article/pii/002205319190155W.

[13] David M Kreps and Robert Wilson. 'Sequential equilibria'. In: *Econometrica: Journal of the Econometric Society* (1982), pp. 863–894.

[14] John C Harsanyi and Reinhard Selten. 'A general theory of equilibrium selection in games'. In: *MIT Press Books* 1 (1988).

[15] John C Harsanyi. 'A new theory of equilibrium selection for games with complete information'. In: *Games and Economic Behavior* 8.1 (1995), pp. 91–122.

[16] Robert Aumann. 'Nash equilibria are not self-enforcing'. In: *Economic decision making: Games, econometrics and optimisation* (1990), pp. 201–206.

[17] Robert J. Aumann. 'Subjectivity and correlation in randomized strategies'. In: *Journal of Mathematical Economics* 1.1 (Mar. 1974), pp. 67–96. DOI: 10.1016/0304-4068(74)90037-8. URL: https://doi.org/10.1016/0304-4068(74)90037-8.

[18] Robert J. Aumann. 'Correlated Equilibrium as an Expression of Bayesian Rationality'. In: *Econometrica* 55.1 (1987), pp. 1–18. URL: http://www.jstor.org/stable/1911154 (retrieved 7/8/2023).

[19] G Khaciyan Leonid. 'A polynomial algorithm for linear programming'. In: *Doklady Akademii Nauk SSSR* 244.1093-1096 (1979), pp. 3–7.

[20] John Nash. 'Non-Cooperative Games'. In: *Annals of Mathematics* 54.2 (1951), pp. 286–295. URL: http://www.jstor.org/stable/1969529 (retrieved 2/9/2024).

[21] Joseph Farrell. 'Communication, coordination and Nash equilibrium'. In: *Economics Letters* 27.3 (Jan. 1988), pp. 209–214. DOI: 10.1016/0165-1765(88)90172-3. URL: https://doi.org/10.1016/0165-1765(88)90172-3.

[22] Joseph Farrell and Matthew Rabin. *Cheap Talk*. Tech. rep. 1996, pp. 103–118.

[23] Russell Cooper et al. 'Communication in coordination games'. In: *The Quarterly Journal of Economics* 107.2 (1992), pp. 739–771.

[24] Gary Charness. 'Self-serving cheap talk: A test of Aumann's conjecture'. In: *Games and Economic Behavior* 33.2 (2000), pp. 177–194.

[25] Kenneth Clark, Stephen Kay and Martin Sefton. 'When are Nash equilibria self-enforcing? An experimental analysis'. In: *International Journal of Game Theory* 29.4 (2001), pp. 495–515.

[26] Vincent P. Crawford and Joel Sobel. 'Strategic Information Transmission'. In: *Econometrica* 50.6 (Nov. 1982), p. 1431. DOI: 10.2307/1913390. URL: https://doi.org/10.2307/1913390.

[27] Imre Bárány. 'Fair Distribution Protocols or How the Players Replace Fortune'. In: *Mathematics of Operations Research* 17.2 (1992), pp. 327–340. URL: http://www.jstor.org/stable/3689961 (retrieved 15/5/2024).

[28] Yuval Heller, Eilon Solan and Tristan Tomala. 'Communication, correlation and cheap-talk in games with public information'. In: *Games and Economic Behavior* 74.1 (2012), pp. 222–234. DOI: https://doi.org/10.1016/j.geb.2011.05.002. URL: https://www.sciencedirect.com/science/article/pii/S089982561100087X.

[29] Françoise Forges. 'Universal Mechanisms'. In: *Econometrica* 58.6 (1990), pp. 1341–1364. URL: http://www.jstor.org/stable/2938319 (retrieved 15/5/2024).

[30] Elchanan Ben-Porath. 'Cheap talk in games with incomplete information'. In: *Journal of Economic Theory* 108.1 (2003), pp. 45–71. DOI: https://doi.org/10.1016/S0022-0531(02)00011-X. URL: https://www.sciencedirect.com/science/article/pii/S002205310200011X.

[31] Dino Gerardi. 'Unmediated communication in games with complete and incomplete information'. In: *Journal of Economic Theory* 114.1 (2004), pp. 104–131. DOI: https://doi.org/10.1016/S0022-0531(03)00124-8. URL: https://www.sciencedirect.com/science/article/pii/S0022053103001248.

[32] Ittai Abraham et al. 'Distributed computing meets game theory: robust mechanisms for rational secret sharing and multiparty computation'. In: *Proceedings of the Twenty-Fifth Annual ACM Symposium on Principles of Distributed Computing*. PODC '06. Denver, Colorado, USA: Association for Computing Machinery, 2006, pp. 53–62. DOI: 10.1145/1146381.1146393. URL: https://doi.org/10.1145/1146381.1146393.

[33] Yuval Heller. 'Minority-proof cheap-talk protocol'. In: *Games and Economic Behavior* 69.2 (2010), pp. 394–400. DOI: https://doi.org/10.1016/j.geb.2009.11.004. URL: https://www.sciencedirect.com/science/article/pii/S0899825609002139.

[34] Ittai Abraham et al. *Implementing Mediators with Asynchronous Cheap Talk*. 2018. arXiv: 1806.01214 [cs.DC].

[35] Julian Jamison. 'Valuable Cheap Talk and Equilibrium Selection'. In: *Games* 11.3 (2020). DOI: 10.3390/g11030034. URL: https://www.mdpi.com/2073-4336/11/3/34.

[36] Roger B. Myerson. 'Graphs and Cooperation in Games'. In: *Mathematics of Operations Research* 2.3 (1977), pp. 225–229. URL: http://www.jstor.org/stable/3689511 (retrieved 16/8/2024).

[37] Dimitris Fotakis, Spyros Kontogiannis and Paul Spirakis. 'Atomic Congestion Games Among Coalitions'. In: *Automata, Languages and Programming*. Springer Berlin Heidelberg, 2006, pp. 572–583. DOI: 10.1007/11786986_50. URL: http://dx.doi.org/10.1007/11786986_50.

[38] John C. Harsanyi. 'Games with Incomplete Information Played by "Bayesian" Players, I-III'. In: *Management Science* 14.7 (1967-68), pp. 159–182, 320–334, 486–502. URL: http://www.jstor.org/stable/2628894.

[39] Jean-François Mertens and Shmuel Zamir. 'Formulation of Bayesian analysis for games with incomplete information'. In: *International journal of game theory* 14.1 (1985), pp. 1–29.

[40] Robert Aumann and Adam Brandenburger. 'Epistemic Conditions for Nash Equilibrium'. In: *Econometrica* 63.5 (1995), pp. 1161–1180. URL: http://www.jstor.org/stable/2171725 (retrieved 24/10/2023).

[41] John Von Neumann and Oskar Morgenstern. 'Theory of games and economic behavior'. In: (1947).

[42] Igal Milchtaich. 'Congestion Games with Player-Specific Payoff Functions'. In: *Games and Economic Behavior* 13.1 (1996), pp. 111–124. DOI: https://doi.org/10.1006/game.1996.0027. URL: https://www.sciencedirect.com/science/article/pii/S0899825696900275.

[43] Matthias Blonski. 'Anonymous games with binary actions'. In: *Games and Economic Behavior* 28.2 (1999), pp. 171–180.

[44] Matthias Blonski. 'Characterization of pure strategy equilibria in finite anonymous games'. In: *Journal of Mathematical Economics* 34.2 (2000), pp. 225–233.

[45] Felix Brandt, Felix Fischer and Markus Holzer. 'Symmetries and the complexity of pure Nash equilibrium'. In: *Journal of computer and system sciences* 75.3 (2009), pp. 163–177.

[46] John C Harsanyi. 'Games with randomly disturbed payoffs: A new rationale for mixed-strategy equilibrium points'. In: *International journal of game theory* 2.1 (1973), pp. 1–23.

[47] Jacob Bernoulli. *The art of conjecturing, together with letter to a friend on sets in court tennis*. JHU Press, 2006.

[48] Pierre Simon marquis de Laplace. *Essai philosophique sur les probabilités*. Bachelier, 1825.

[49] John Maynard Keynes. *A treatise on probability*. Courier Corporation, 2013.

[50] Edwin T Jaynes. 'Information theory and statistical mechanics'. In: *Physical review* 106.4 (1957), p. 620.

[51] Michael R. Garey and David S. Johnson. 'Computers and intractability: a guide to the theory of np-completeness'. In: *WH Free. Co., San Fr* 38 (1979), p. 40.

[52] R. L. Graham. 'Bounds on Multiprocessing Timing Anomalies'. In: *SIAM Journal on Applied Mathematics* 17.2 (1969), pp. 416–429. URL: http://www.jstor.org/stable/2099572 (retrieved 28/3/2025).

[53] Richard Korf. 'Objective Functions for Multi-Way Number Partitioning'. In: *Proceedings of the International Symposium on Combinatorial Search* 1.1 (Aug. 2010), pp. 71–72. DOI: 10.1609/socs.v1i1.18172. URL: http://dx.doi.org/10.1609/socs.v1i1.18172.

[54] Robert W. Rosenthal. 'A class of games possessing pure-strategy Nash equilibria'. In: *International Journal of Game Theory* 2.1 (Dec. 1973), pp. 65–67. DOI: 10.1007/bf01737559. URL: http://dx.doi.org/10.1007/BF01737559.

[55] Dov Monderer and Lloyd S. Shapley. 'Potential Games'. In: *Games and Economic Behavior* 14.1 (May 1996), pp. 124–143. DOI: 10.1006/game.1996.0044. URL: http://dx.doi.org/10.1006/game.1996.0044.

[56] Elias Koutsoupias and Christos Papadimitriou. 'Worst-case equilibria'. In: *Computer Science Review* 3.2 (May 2009), pp. 65–69. DOI: 10.1016/j.cosrev.2009.04.003. URL: http://dx.doi.org/10.1016/j.cosrev.2009.04.003.

[57] George Christodoulou and Elias Koutsoupias. 'The price of anarchy of finite congestion games'. In: *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing*. Baltimore, MD, USA: Association for Computing Machinery, 2005, pp. 67–73. DOI: 10.1145/1060590.1060600. URL: https://doi.org/10.1145/1060590.1060600.

[58] George Christodoulou and Elias Koutsoupias. 'On the Price of Anarchy and Stability of Correlated Equilibria of Linear Congestion Games, ,' in: *Algorithms – ESA 2005*. Springer Berlin Heidelberg, 2005, pp. 59–70. DOI: 10.1007/11561071_8. URL: http://dx.doi.org/10.1007/11561071_8.

[59] Baruch Awerbuch, Yossi Azar and Amir Epstein. 'The Price of Routing Unsplittable Flow'. In: *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*. STOC05. ACM, May 2005. DOI: 10.1145/1060590.1060599. URL: http://dx.doi.org/10.1145/1060590.1060599.

[60] Sebastian Aland et al. 'Exact Price of Anarchy for Polynomial Congestion Games'. In: *SIAM Journal on Computing* 40.5 (Jan. 2011), pp. 1211–1233. DOI: 10.1137/090748986. URL: http://dx.doi.org/10.1137/090748986.

[61] Kshipra Bhawalkar, Martin Gairing and Tim Roughgarden. 'Weighted Congestion Games: The Price of Anarchy, Universal Worst-Case Examples, and Tightness'. In: *ACM Transactions on Economics and Computation* 2.4 (Oct. 2014), pp. 1–23. DOI: 10.1145/2629666. URL: http://dx.doi.org/10.1145/2629666.

[62] Dimitris Fotakis et al. 'The Structure and Complexity of Nash Equilibria for a Selfish Routing Game'. In: *Automata, Languages and Programming*. Ed. by Peter Widmayer et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 123–134.

[63] Panagiota N Panagopoulou and Paul G Spirakis. 'Algorithms for pure Nash equilibria in weighted congestion games'. In: *Journal of Experimental Algorithmics (JEA)* 11 (2007), pp. 2–7.

[64] Tobias Harks and Max Klimm. 'On the existence of pure Nash equilibria in weighted congestion games'. In: *Mathematics of Operations Research* 37.3 (2012), pp. 419–436.

[65] Tim Roughgarden and Éva Tardos. 'How bad is selfish routing?' In: *Journal of the ACM (JACM)* 49.2 (2002), pp. 236–259.

[66] Tim Roughgarden. 'The price of anarchy is independent of the network topology'. In: *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*. 2002, pp. 428–437.

[67] José R Correa, Andreas S Schulz and Nicolás E Stier-Moses. 'Selfish routing in capacitated networks'. In: *Mathematics of Operations Research* 29.4 (2004), pp. 961–976.

[68] José R Correa, Andreas S Schulz and Nicolás E Stier-Moses. 'A geometric approach to the price of anarchy in nonatomic congestion games'. In: *Games and Economic Behavior* 64.2 (2008), pp. 457–469.

[69] Tim Roughgarden. 'Intrinsic robustness of the price of anarchy'. In: *Journal of the ACM (JACM)* 62.5 (2015), pp. 1–42.

[70] Berthold Vöcking. 'Selfish Load Balancing'. In: *Algorithmic Game Theory*. Ed. by Noam Nisan et al. Algorithmic Game Theory. Cambridge University Press, 2007, pp. 517–542.

[71] Richard M. Karp. 'Reducibility among Combinatorial Problems'. In: *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations, held March 20–22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, and sponsored by the Office of Naval Research, Mathematics Program, IBM World Trade Corporation, and the IBM Research Mathematical Sciences Department*. Ed. by Raymond E. Miller, James W. Thatcher and Jean D. Bohlinger. Boston, MA: Springer US, 1972, pp. 85–103. DOI: 10.1007/978-1-4684-2001-2_9. URL: https://doi.org/10.1007/978-1-4684-2001-2_9.

[72] John C Harsanyi. 'Cardinal welfare, individualistic ethics, and interpersonal comparisons of utility'. In: *Journal of political economy* 63.4 (1955), pp. 309–321.

[73] John Rawls. *A Theory of Justice*. Revised. Cambridge, MA: Belknap Press of Harvard University Press, 1999.

[74] R Walter. 'Comparing the minimum completion times of two longest-first scheduling-heuristics'. In: *Cent Eur J Oper Res 21, 125–139* (2013). DOI: https://doi.org/10.1007/s10100-011-0217-4.

[75] Michael R. Garey and David S. Johnson. 'Computers and Intractability. A Guide to the Theory of Np-Completeness'. In: *Journal of Symbolic Logic* 48.2 (1983), pp. 498–500. DOI: 10.2307/2273574.

[76] Amartya Sen. *Collective choice and social welfare: Expanded edition*. Penguin UK, 2017.

[77] Peter J Hammond. 'Equity, Arrow's Conditions, and Rawls' Difference Principle'. In: *Econometrica* 44.4 (July 1976), pp. 793–804. DOI: None. URL: https://ideas.repec.org/a/ecm/emetrp/v44y1976i4p793-804.html.

[78] Joseph Farrell and Matthew Rabin. 'Cheap Talk'. In: *Journal of Economic Perspectives* 10.3 (Aug. 1996), pp. 103–118. DOI: 10.1257/jep.10.3.103. URL: https://doi.org/10.1257/jep.10.3.103.

[79] Jason D. Hartline. 'Approximation in Mechanism Design'. In: *The American Economic Review* 102.3 (2012), pp. 330–336. URL: http://www.jstor.org/stable/23245552 (retrieved 28/3/2025).