Fair Division of the Commons

Dominik Peters

Thesis

DPhil in Computer Science

University of Oxford
Department of Computer Science

Supervised by Prof Edith Elkind
Submitted: 10 May 2019

Viva: 2 September 2019

Abstract

A group of agents controls a common budget or owns some common resources. The agents need to decide how to divide this budget across various projects, or to distribute the resources among themselves. Each agent has their own preferences about the best use of the resources. We study ways in which the agents can make these decisions in a fair manner. By fairness, we will mean that for every group member, a proportional part of the common budget is spent in accordance with the member's interests. We will also be interested to take into account the interests of subgroups, and when appropriate aim to avoid envy between group members.

We consider several settings in this thesis, capturing different types of potential uses of the common budget. For example, we distinguish between projects that come with a fixed cost (and can be either fully funded or not at all), and projects that can flexibly scale with the amount of funding received. We also distinguish between uses that potentially benefit several or all group members (public goods) or uses that benefit only one agent (private goods).

For each scenario we consider, we formalise what we might mean by a "fair" outcome, and then design decision rules that guarantee fairness. Where possible, we additionally aim for rules that make Pareto-efficient use of the common budget. Unfortunately, many of our rules can be exploited by strategic agents who misreport their preferences. In these cases, we prove impossibility theorems which imply that no fair decision rule can be resistant to such strategic manipulation. These impossibility theorems are proved using a computer-aided technique based on SAT solvers, which allows us to obtain computer-generated but human-readable proofs. Further, we consider the computational complexity of the decision rules we consider. In most cases, they can be evaluated using efficient algorithms. In other cases, there are NP-completeness results, but we can show that efficient algorithms exist that work when preferences are well-behaved, in the sense of exhibiting underlying structure.

List of Chapters

Contents

Acknowledgements

Edith Elkind has been the perfect supervisor for me. She helped start my research by proposing fruitful open problems, and throughout my studies has provided great advice for the best ways forward and the best opportunities to pursue. Edith gives her students great freedom, and the success of this method for producing independent researchers is readily apparent in my distinguished academic brothers and sisters.

Thanks to the examiners of this thesis, Mike Wooldridge and Vince Conitzer, for their careful study and insightful comments and questions during the viva. I have also benefited from my transfer and confirmation vivas, and am grateful for the guidance I received from Mike, and from Elias Koutsoupias and Paul Goldberg.

I have been fortunate to work with incredible collaborators, and have especially enjoyed my research visits around the world. Felix Brandt has hosted me for five visits and had a strong influence on my research tastes. I'm honoured to be listed as an "external PhD student" on his CV. I also thank Ariel Procaccia, Bill Zwicker, Jörg Rothe, Markus Brill, Piotr Skowron, Martin Lackner, Rupert Freeman, and Jérôme Lang for being gracious and inspiring hosts. My former fellow DPhil student Ayumi Igarashi has been a great friend and collaborator. Thanks also to my collaborators Florian Brandl, Robert Bredereck, Piotr Faliszewski, Christian Geist, and Paul Harrenstein.

Office 017 in the basement of the Department of Computer Science has been the location of many joyous moments, and my thanks go out to everyone who passed through it: Markus Brill, Ayumi Igarashi, Muhammad Najib, Martin Lackner, Piotr Skowron, Robert Bredereck, Jiarui Gan, Haris Aziz, Bill Zwicker, Alex Voudouris, Warut Suksompong, and of course Paul Harrenstein (office 013) who put up with too much laughter coming through the walls of 017.

My eight years in Oxford have provided me with many friends who have been invaluable supporters, including Ali, Amadea, Andrea, Ari, Blago, Cheryl, Dominic, Eileen, Eliot, Elise, Francis, Golo, Greg, Hadrien, Jacob, Janet, Jens, Katrien, Kim, Lucius, Lukas, Marcus, Piotr, Rustin, Sofia, Tom, and Žiga.

I can hardly believe that I have the opportunity to work with so many amazingly talented people, and that I got to study at this incredible university. The reason I came here was the vision of my parents Andre and Beate, who pushed me to think bigger and to go abroad for my studies, and who supported me enthusiastically throughout.

Bibliographic Notes

Most of the contents of this thesis have been published in conference proceedings or journals. Much of the work was done in collaboration with others. Here, I list prior publications, and discuss any differences between the published version and the version included here. Where there are many coauthors, I indicate the extent of my contribution. All parts of the thesis are either written or significantly edited by me.

Part I

Chapter 1 is based on a joint paper with Felix Brandt and Christian Geist, which appeared in AAMAS 2016 and in the journal Mathematical Social Sciences [Brandt et al., 2016b, 2017]. It also appeared in the PhD thesis of Christian Geist. My contributions to the paper include programming, co-development of the method, production of the impossibility proof, and parts of the writing. The present version contains a new introduction, and a simplified proof of the main result. The published paper contains additional results about set-valued and probabilistic voting rules.

Chapter 2 is based on a paper published in the proceedings of TARK 2017 [Peters, 2017a]. The published paper contains additional results about set-valued voting rules and about a weakening of half-way monotonicity, as well as a sketch of the contents of Chapter 3.

Chapter 3 has not been published before, and Theorem 3.6 appears here for the first time.

Part II

Chapter 4 is based on a paper appearing in ACM EC 2019 [Freeman, Pennock, Peters, and Wortman Vaughan, 2019]. The published paper contains additional material about a game-theoretic interpretation of the Independent Markets mechanism. Section 4.7 appears here for the first time.

Chapter 5 is written for this thesis. The results have been obtained in collaboration with Felix Brandt and Christian Stricker, and are also included in a draft paper phrased in a somewhat different formal setting [Brandl, Brandt, Peters, Stricker, and Suksompong, 2019b].

Chapter 6 is based on a paper appearing in IJCAI 2019 [Airiau, Aziz, Caragiannis, Kruger, Lang, and Peters, 2019]. I joined my coauthors on this project relatively late, and my main contributions are in the exposition, the discussion in Section 6.4, and Theorem 6.7.

Part III

Chapter 7 is based on a paper appearing in AAMAS 2018 [Peters, 2018a].

Chapter 8 is based on a draft for a journal paper which in turn is based on two conference papers on similar topics [Yu, Chan, and Elkind, 2013, Peters and Elkind, 2016], the latter appearing in AAAI 2016. Most of the writing in the chapter as well as most results are my work, but I have not contributed to the proofs of Theorems 8.7 and 8.9 except for minor editing.

Chapter 9 is based on a draft of a journal paper combining two conference papers [Peters and Lackner, 2017, Peters, 2018b] appearing in AAAI 2017 and AAAI 2018 respectively.

Part IV

Chapter 10 is based on a paper appearing in ITCS 2019 [Bilò, Caragiannis, Flammini, Igarashi, Monaco, Peters, Vinci, and Zwicker, 2018]. My main contributions are in writing and exposition, as well as the proof of Theorem 10.18. Section 10.3 states a result from a paper appearing in IJCAI 2017 [Bouveret, Cechlárová, Elkind, Igarashi, and Peters, 2017]. The latter paper contains additional results about the computational complexity of deciding the existence of envy-free and of proportional allocations. This paper is also part of the DPhil thesis of Ayumi Igarashi [2018].

Chapter 11 is based on a paper appearing in AAAI 2019 [Igarashi and Peters, 2019], but omits some hardness proofs from the conference paper.

Chapter 12 is a brief note and has not previously been published.

Omitted Work

Some work performed during my DPhil studies is not included in this thesis, to obtain a shorter and more coherent document.

Hedonic Games and Coalition Formation

In a hedonic game, a set of agents needs to be partitioned into coalitions. Each agent i has preferences over which other agents are part of i's coalition. For example, there may be friendships or aversions between different agents, and agents may have different ideas about the ideal coalition size. The literature on hedonic games mostly aims for a stable partition in which no agent or group of agents wishes to change their coalition.

In my master's thesis [Peters and Elkind, 2015, IJCAI], we unify many computational complexity results for hedonic games in a single framework, and show that most of the existing results and several new ones can be proven using just three reductions. Later, I studied complexity issues in hedonic games with dichotomous preferences [Peters, 2016a, AAAI] and proved that reasoning about the core is Σ_2^p-complete for additive and for Boolean hedonic games [Peters, 2017b, ADT]. I also introduced the idea of graphical hedonic games where agents only care about the presence of their neighbours in a social network. Many stability problems become tractable if the social network has bounded degree and bounded treewidth [Peters, 2016b, AAAI]. I have contributed to a journal paper about fractional hedonic games [Aziz, Brandl, Brandt, Harrenstein, Olsen, and Peters, 2019b, ACM TEAC], solving some open problems from an earlier conference paper. We have also done some work on the related model of group activity selection problems [Igarashi, Peters, and Elkind, 2017, AAAI].

Structured Preferences

When we can identify underlying structure in agents' preferences, we can better model their desires, and it may help us in identifying better social decisions. Work along these lines is included in this thesis in Chapter 8 on preferences single-peaked on a tree and in Chapter 9 on preferences single-peaked on a circle.

I have also worked on the recognition problem of multidimensional Euclidean preferences, proving that it is ETR-complete [Peters, 2017c, AAAI]. We have also worked on single-crossing preferences and especially on identifying whether a preference profile is *almost* single-crossing [Jaeckle, Peters, and Elkind, 2018, AAAI, Lakhani, Peters, and Elkind, 2019, IJCAI].

Committee Elections

Part III of this thesis deals with the problem of multi-winner elections, where the aim is to find a good committee of k candidates, where k is fixed. Additional work on this topic considers the quality of some heuristics for finding good committees under different objective functions [Faliszewski, Lackner, Peters, and Talmon, 2018a, AAAI], and how to extend notions of proportionality for committee elections to rankings of candidates [Skowron, Lackner, Brill, Peters, and Elkind, 2017, IJCAI].

Voting

In Part I of this thesis, we obtain several impossibility theorems about standard single-winner voting. I have also contributed to a journal paper about single-winner voting [Bachmeier, Brandt, Geist, Harrenstein, Kardel, Peters, and Seedig, 2019, JCSS], proving that the famous Kemeny rule remains NP-hard to evaluate for exactly 7 voters. While Kemeny was known to remain hard for 4 voters, this is the first hardness result for a constant odd number of voters. We have also obtained an axiomatic characterisation of the Borda mean rule [Brandl and Peters, 2019, Social Choice and Welfare]. Further, we have studied the popular STV rule (Single Transferable Vote), considering issues of communication and incomplete preferences [Ayadi, Ben Amor, Lang, and Peters, 2019, AAMAS].

Overview Articles

We have produced a survey article and a book chapter on structured preferences [Elkind, Lackner, and Peters, 2016, 2017b]. We have also published a book chapter explaining the technique of using SAT solvers to prove impossibility theorems in social choice [Geist and Peters, 2017].

0. Introduction

In 2016, I participated in the "Oxford Prioritisation Project", which was a student research group set up by my friend Tom Sittler. We had been given GBP 10 000 by the Centre for Effective Altruism. The aim of the ten group members was to identify a charitable recipient for this money, and we hoped to maximise the expected impact of the donation. The search space of possible charities is vast, and many of them do things that seem positive and impactful. Thus, we needed to prioritise among them, and we built quantitative models to estimate the impact per dollar of several options. After a period of three months of research, we hoped to come to a unanimous decision for the destination of the money. We recognised that in the end there may still be disagreements, in which case we would have to take a vote. Voting ended up being unnecessary, and we all agreed to give the money to *80 000 Hours*, a group that advises young people about effective and altruistic career paths. But the episode made me think: if we wanted to vote over how to split the pool of GBP 10 000 among charities, what would be a good method to do so?

I worked on my DPhil as a member of Oxford's Balliol College. The graduate students of the college form the *Middle Common Room (MCR)*, which is simultaneously an organisation to represent student interests, and also a physical common room. This handsome room in Holywell Manor features leather sofas, portraits of long-dead influential members of Balliol, and a big table full of newspapers and magazines. The MCR takes out subscriptions of these newspapers and magazines, and has an annual budget of about GBP 2 000 for this purpose. Once a year, we send around a form allowing members to vote over which periodicals to subscribe to. The form allows each person to tick as many options as they want. Then the most popular choices are bought, up to the budget limit. Students with tastes close to the mainstream will have much to read, but this decision method may leave substantial minorities without a desired but niche reading option – the *Guardian* has eaten up most of the budget. What would be a fair and representative voting method?

The Department of Computer Science has only a small number of meeting rooms. Departmental classes, seminars, group discussions, and administrative meetings have to compete over slots in these rooms. Slots are given out on a first-come-first-served basis by emailing rooms@cs.ox.ac.uk with a request. Late-comers may miss out, or find that the room schedule is so fragmented that no single room is available for the required amount of time. More organised planners get most of their requests fulfilled, which others may find unfair. Is there a way to decide on a room schedule that is fair and efficient? Can it be done without blow-ups in computational complexity?

This thesis contains partial answers to the questions raised by these three stories. In each case, a group of people needs to decide how to use common resources and how to divide them among different options. Each case raises issues of fairness. And each case asks for the design of an appropriate procedure to aggregate the group's preferences.

The three stories illustrate different decision-making settings, and they suggest a taxonomy of problems where we need to divide a common budget. In the third story, when allocating time slots to users, a time slot is only useful to a single user. Time slots are *private goods*, as they are for the exclusive use of a single agent. In contrast, in the first two stories, the budget was divided among goods that were not exclusive: everyone can benefit from the newspaper subscriptions, and everyone is affected by the donation decisions. Thus, in these instances, we are deciding on the provision of *public goods*.

	public goods	private goods
divisible	Part II	–
indivisible	Part III	Part IV

Table 0.1.: A taxonomy of budget division problems.

We can further distinguish these cases along a second axis. When dividing an amount of money among charities, the budget is perfectly *divisible*, and we can send an arbitrary fraction of the pool to a single charity. (In practice, the divisibility of money is limited to cents, but the problem is more fruitfully modelled when ignoring this issue.) In contrast, when using a common budget to buy magazines, we have to make a binary decision for each option: we cannot subscribe to two-thirds of a magazine. In this case, the options are *indivisible*. Similarly, it is usually convenient to handle room bookings in discrete time slots of 30 or 60 minutes, making them also indivisible.

These two distinctions – public goods versus private goods, divisible options versus indivisible ones – give rise to a 2×2 table, shown in Table 0.1.[1] Our stories correspond to three of the four entries, and we will study these three models in Parts II, III, and IV of this thesis. The fourth entry concerns divisible private goods, a model commonly known as *cake-cutting*. We do not add to the cake-cutting literature in this thesis, but our discussion in Part IV takes inspiration from several famous results from cake-cutting.

We will study these different budget division problems using the method and the point of view of *social choice theory* and *computational social choice*. In social choice, we look at group decision making as a three-step process: First, the members of the group write down their preferences concerning the outcome (for example, by ticking all newspapers that they would like to read). Second, the reported preferences are analysed and aggregated using a previously-chosen algorithm, a *voting rule*. The voting rule will output a group decision (for example, a collection of newspapers to subscribe to). Third, the chosen outcome is implemented by the group.

The described process is only a model of actual group decision making, and necessarily ignores much real-life complexity. For example, social choice tends to ignore the process by which the group arrives at the set of options to vote over (for example, a list of potential subscriptions, or a list of potential charities). It also limits the types of preferences that group members are able to report (for example, it might only allow rankings or a set of ticks). It ignores group deliberation to reach consensus before a vote. However, the tools of social choice really shine at the second step of the procedure outlined above: the theorist can design voting rules that do an impressive job in weighing different preferences and in constructing an outcome that respects the interests of all group members.

For each of the three types of budget division problems that we analyse, we will aim for four different types of contributions: to formalise appropriate notions of fairness for the setting, to design good rules for obtaining a group decision, to explore the boundary of which desirable properties can be satisfied, and to study the computational complexity of the rules we propose.

- *Formalising fairness.* Fairness can mean very different things, and which kind of fairness is desirable to achieve (if any) depends on the context. To be able to properly discuss fairness, it helps to have formal definitions of what we mean. In budgeting problems of the kind we are looking at, it would seem to be unfair if we completely ignore the interests of some of the group members, in the sense of using none or very little of the budget in accordance with their interest. In the case of indivisible goods, it might not be possible to do enough

[1] I thank Brandon Fain for discussions on this point.

different things to partially satisfy everybody. However, we might want to require that once a large enough group of agents has similar interests, then part of the budget needs to be spent in accordance with them. We may also want to be fair to groups, so that a group of people with similar interests gets to 'control' a fraction of the budget in proportion to the group's size. For private goods, we can also consider the classical notion of avoiding envy.

- *Designing good rules.* Once we have identified a desirable form of fairness, and probably a collection of other essential properties such as Pareto efficiency, our next task is to design preference aggregation rules that satisfy them. In most cases, during the design process, we can draw inspiration from rules that have been successful in other parts of social choice, such as the idea of maximising the product (rather than the sum) of voter utilities. In other cases, we have to design rules from scratch, and in still others we will look for help from computer programs such as SAT solvers. Of course, the types of settings we consider have in many cases already been studied by others, in which case our task reduces to further analysing existing proposals.

- *Exploring the limits of aggregation rules.* In many cases, the results of the design process will fall short of our hopes, and the rules obtained will not satisfy all the properties we had identified as desirable. In this case, it is convenient to have an excuse, and a formal version of an excuse can be obtained in form of impossibility theorems. Such theorems have a long history in social choice; they show that no logically possible aggregation rule can satisfy all the desired properties, and thus they establish a formal trade-off between them. Impossibility theorems are used throughout this thesis to illuminate the settings we discuss. We obtain these using computer-aided proof techniques.

- *Ensuring computational efficiency.* A preference aggregation rule is less useful if it is computationally expensive to evaluate it. In many cases, we are able to complement our proposed rules with efficient algorithms. However, particularly when it comes to indivisible goods, the space of possible outcomes has a combinatorial character, and it is often provably hard (e.g., NP-complete) to find a socially optimal outcome. For these rules, we will look for algorithms that are efficient under additional assumptions about the input preferences. In particular, we will look at *structured* preferences and find that for many appealing types of structure, efficient algorithms can be obtained.

Part I opens our discussion with a methodological prelude. It introduces a method based on SAT solving for obtaining impossibility theorems in social choice. We illustrate the method using applications to single-winner voting, and so in contrast to the rest of the thesis, we do not discuss the division of a common resource. Still, the discussion in this first part lays the groundwork for what is to come: We will use the method to uncover trade-offs in each of the three models of fair division that we study in parts II, III, and IV.

An impossibility theorem gives a list of properties that no aggregation mechanism can simultaneously satisfy. It implies that, when designing mechanisms, we must choose some of the properties but must give up the rest. Commonly, these theorems identify a conflict between the quality of the social outcome selected by the mechanism and the resistance of this mechanism to strategic misrepresentation by the agents. One way to prove such an impossibility is to use logic tools such as SAT solvers to search over the space of all logically possible aggregation mechanisms; if the search fails to find an example of a mechanism satisfying all our desired properties, we have established an impossibility. Excitingly, a technique based on *minimal unsatisfiable sets* allows us to automatically extract a human-readable proof of the impossibility, providing rare examples of computer-generated proofs that are comprehensible to humans.

In Chapter 1, we study the no-show paradox. This is a surprising defect of many voting rules. A voting rule in this context asks every voter to submit a complete ranking of the candidates (imagine the election of the president of a country). Given these rankings, the voting rule must select a single winning candidate. Many possible voting rules have been proposed and have been used in elections around the globe. In the 1980s, social choice theorists noticed that under many of these rules, it can be beneficial for voters to abstain from an election. For example, it can happen that a voter who ranks candidate c in top position causes c to lose if the voter participates; if the voter abstains and does not submit the ranking, then c is elected by the rule. In particular, this occurs for voting rules from a class proposed by the 18th century French intellectual Condorcet. We focus on the case where exactly 4 candidates are running. Using a SAT solver, we construct a Condorcet rule which avoids the no-show paradox as long as at most 11 voters participate. On the other hand, we prove that no such rule exists for 12 or more voters. This improves upon a result of Moulin [1988b] who proved that no such rule exists for 25 or more voters.

In Chapter 2, we consider a related paradox that we call the preference reversal paradox. This occurs when it is sometimes advantageous for a voter to submit the complete opposite of the actual truthful preference ranking. Again, a surprising number of voting rules exhibit this paradox, including all Condorcet rules. We show that the paradox is unavoidable for Condorcet rules when there are at least 15 voters (if the number of voters is odd) or at least 24 voters (if it is even).

In Chapter 3, we build on our work on the preference reversal paradox to prove a variant of the Gibbard–Satterthwaite theorem. This is a foundational result of social choice theory which shows that every efficient voting rule can be manipulated, in the sense that voters can lie about their preferences and thereby obtain a better election outcome for themselves. The sole exception are the "dictatorial" rules, which identify a single voter and always select that voter's top choice – clearly, such rules are not manipulable, but they are rather undesirable. Our version of the Gibbard–Satterthwaite theorem tries to make it more vivid, by giving a more explicit account of the kind of manipulations that are unavoidable: every sensible voting rule can either be manipulated by completely reversing one's preferences, or by a manipulation that voting rules could easily avoid if they followed Condorcet's principle.

Part II studies the division of a divisible common budget or resource among several options which have a public goods character. As a high-stakes example, we can imagine a cabinet deciding on how to divide the government's budget among departments such as health, pensions, defence, education, transportation, and so on. The outcome of the decision can be visualised as a pie chart, showing the percentage of the budget spent on each area. The cabinet members have different preferences over how the pie chart should look, perhaps due to ideological differences or simply by favouring their own departments. Our aim is to design aggregation mechanisms that can turn these preferences into a final pie chart.

Similar decision problems appear frequently in lower-stakes scenarios, and the 'budget' to be divided need not be monetary. Let us briefly mention some applications. A team organising a conference may wish to decide how much time to assign to talks, poster sessions, invited talks, and coffee breaks, as a fraction of the total length of the conference. Coauthors need to decide how much space to devote to various topics in a textbook or article as a fraction of the fixed total length. A company which annually donates money to charity could let employees vote over which charities should receive a donation. Finally, in a parliamentary election, voters decide what percentage of parliament seats should go to each party, and in many countries there is some discontent with the current voting systems for this purpose.

Preferences over the division of the budget among projects could be rather complicated, and Part II is divided into three chapters which study different input formats for these preferences. In each case, we will try to identify a budget division that is good (in terms of Pareto efficiency) and which is fair. Fairness can have different meanings depending on the format of preferences, but in general we will aim for notions of proportionality. For example, if 40% of the voters are strong proponents of a particular project, then about 40% of the budget should be spent on it.

In Chapter 4, we ask each voter to specify how they would split the budget if they were the sole decision maker. Thus, each voter submits an ideal pie chart, their own budget proposal. A particularly natural way of aggregating the proposals is by taking the average: for each project, spend on it the average of the fractions reported in the input pie charts. This method has many desirable properties, but it has the drawback of being easily manipulated. For example, if we have to divide the budget among two projects and most voters propose a 50–50 split, then a voter favouring a 60–40 split would do well by instead proposing 100–0. We develop a class of mechanisms that are not manipulable, assuming that voter preferences are such that they prefer pie-charts that are close to their proposals according to the L1 metric, which sums up the differences on each project. This broad class of mechanisms includes a mechanism inspired by a market system which leads to outcomes that are proportional in a limited sense. It also includes a mechanism that maximises utilitarian social welfare, and we characterise this mechanism as the unique Pareto-efficient mechanism in our class. We then briefly consider a related setting where projects come with a minimal funding level below which they do not make sense. We show that this additional constraint makes it impossible for rules to be non-manipulable.

In Chapter 5, we ask voters for an approval set: they should indicate which of the possible projects they approve. They can approve as many projects as they like. We take a voter's happiness with a particular budget division to be the total amount of spending on the voter's approved projects. Bogomolnaia et al. [2005] studied this setting, and proposed a number of attractive aggregation rules. A particularly interesting rule selects the budget distribution that maximises the product of voter utilities, that is, the rule that maximises the Nash product. This rule is efficient and satisfies a strong core-like fairness property. The Nash rule is, however, manipulable. Bogomolnaia et al. [2005] conjectured that there is an impossibility theorem showing that no rule can simultaneously be efficient, non-manipulable, and fair in a minimal sense. We use SAT solvers to prove their conjecture.

In Chapter 6, we consider the most traditional input method in social choice theory: voters are asked to rank the different projects. We study a large class of rules that make sense in this setting; these are based on converting rankings into numerical scores, and then maximising a measure of social welfare. Egalitarian rules from our class satisfy individual fairness properties, and Nash-like rules satisfy group fairness notions including our new concept of SD-core.

Part III considers the division of a common budget among projects that must either be fully funded or not at all. The newspaper story at the beginning is an example of this type of problem. A higher-stakes occurrence is in many large cities around the world that use *participatory budgeting*. For example, in Paris, the city government sets aside about EUR 100 000 000 every year for this purpose. Residents of the city can submit a proposal for a project together with a required funding level. Example proposals include improving a neighbourhood park or renovating a local school. Then, there is a city-wide election where each voter can vote for up to 4 of the projects, and the most popular proposals get funded, up to the budget limit. Participatory budgeting started in Brazil and is getting more and more popular. City governments like it for the greater civic involvement.

The formal analysis of this setting in its full generality has only begun recently. Designing satisfying rules can be surprisingly tricky. In the poll over which newspapers Balliol MCR should buy, I volunteered to apply a voting rule based on the Nash product that seemed sensible to me. I took the utility of a member to be the number of approved periodicals that we would purchase, and then used an integer programming solver to find an affordable set that maximised the product of utilities. To my initial surprise, the result looked very different from what we had bought in previous years; in particular, no daily newspapers were purchased in optimum. The intuitive reason is that daily papers are much more expensive than weekly or monthly magazines, and the objective function made it uneconomical to purchase a daily, even though the number of ticks for the *Guardian* far exceeds that for any other choice. We fixed this problem by adding a constraint that at least two dailies must be included in the final result, but this ad hoc approach cannot be the last word. Without complicating the preference elicitation process, I do not see a good solution to this problem. The underlying issue is that an approval-based system cannot figure out how to trade off prices and approval scores.

To side-step this issue for now, the discussion in Part III will focus entirely on the *unit cost* case, where all projects have the same cost. Thus, the budget division problem consists of identifying exactly k projects to fund, for some fixed k. This problem has received significant attention by researchers in the last several years under the names of *multiwinner elections* and *committee selection*. We will mostly operate in the framing of electing a committee of people to represent a larger electorate, and study rules proposed for this purpose, such as the Chamberlin–Courant rule or Proportional Approval Voting.

In Chapter 7, like in Chapter 5, we consider a setting based on approvals, where voters can approve as many candidates as they like. The arguably most interesting rules for this setting have been proposed by late 19th century Danish and Swedish mathematicians Thiele and Phragmén. Their proposed rules have the aim of proportionally representing the electorate. However, both of these rules can be manipulated by voters who claim not to approve popular parties, a type of free-riding. We prove, again using SAT solvers, that every approval-based committee rule is manipulable in this way, as soon as it provides a minimum amount of voter representation.

In Chapters 8 and 9, we shift our focus to computational issues. The rules in the divisible case considered in Part II are, as we show, all easy to implement, in the sense that the output pie chart can be computed by fast algorithms. In contrast, many popular rules for the indivisible case have winner determination problems that are NP-complete. These rules are typically defined so that they maximise some objective function over the set of all possible committees. Since there are exponentially many possible committees, it can be hard to find the best committee using an efficient algorithm. We show that these intractability results can be avoided when the voter preferences are well-behaved. In particular, we study cases in which the candidate space has some underlying structure, so that the preferences are also structured. This structure can then be used in efficient winner determination algorithms for rules such as the Chamberlin–Courant rule.

In Chapter 8, we discuss preferences that are single-peaked on a tree. This is a generalisation of the classic concept of single-peaked preferences which are suitable, for example, when voting over the value of a numerical quantity. Preferences are single-peaked when the alternatives can be arranged on an axis from left to right, such that each voter's most preferred alternative forms a peak, and preferences are decreasing as we move to alternatives further away on the axis. When preferences are single-peaked, most voting impossibilities and paradoxes go away, and so do computational intractability results – for example, Chamberlin–Courant can be computed in polynomial time for single-peaked preferences. Unfortunately, only an exponentially small fraction of preference profiles are single-peaked, and hence these positive results often do not apply in practice. We consider a more permissive generalisation of single-peakedness, where the alternative space is allowed to have any tree structure. We show that an egalitarian version of the Chamberlin–Courant rule remains tractable on this more general domain, but that the standard version becomes NP-complete. However, we show that if the underlying tree is well-behaved, tractability results can again be obtained. To use the algorithms we propose in this chapter, it is necessary to have a good understanding of how to find a tree on which a given preference profile is single-peaked, and we give a detailed graph-theoretic treatment of this issue.

In Chapter 9, we introduce and study a different generalisation of single-peaked preferences: those that are single-peaked on a circle. This preference model can make sense, for example, when voting over the time of a recurrent meeting (with times arranged on a 24-hour clock), or for facility location when facilities can only be built on the boundary of a city or plot of land. We show how to efficiently recognise whether a given profile is single-peaked on a circle, and we study the axiomatic properties of this domain restriction. Also, we show that many hard committee selection rules, including Chamberlin–Courant and Proportional Approval Voting, are polynomial-time computable when preferences are single-peaked on a circle. We prove this by developing special integer linear programming formulations for the relevant winner determination problems, and show that these formulations are totally unimodular whenever the input preferences are single-peaked on a circle.

Part IV studies the allocation of private goods. A collection of indivisible items needs to be distributed among agents who have different valuations for these items. Agents can receive several items. If an item is allocated to an agent, no other agent can use that item. The private-goods nature of the setting allows us to study a fairness notion that is quite different from the proportionality-type notions that we studied in Parts II and III: we can aim for envy-freeness, which requires that no agent thinks that another agent received a strictly more valuable bundle.

The word "commons" in the thesis title is meant to both refer to something commonly owned (such as the budgets in Parts II and III), as well as to land (like in the grazing land of the Tragedy of the Commons). When dividing land among several parties, an important consideration is to not make the individual pieces disconnected. Disconnected pieces of land are much less useful. The classic literature on the division of a perfectly divisible resource ("cake-cutting") has taken this consideration seriously, and many popular protocols aim to minimise the number of "cuts" needed to achieve a fair outcome. However, the literature on indivisible items has traditionally taken the set of items to have no internal structure, so that the notion of a "connected" bundle of items does not make sense. However, in practice, the set of items may well have more structure. The case of booking time slots in a meeting room illustrates this: a 90-minute meeting will be more productive if held during a contiguous chunk of time, rather than split into three 30-minute parts. Another example might occur when an organisation moves to a new building, and has to assign offices to various teams: team communication will be aided if team members are assigned adjacent offices.

We formalise this idea by taking the set of items to be the vertices of a graph. While deciding on how to allocate the items, we will only allow ourselves to give out bundles which induce a connected subgraph. The basic and most interesting example occurs when the underlying graph is a path, and this can in particular model time slots. Our basic question is whether positive results from the literature on indivisible items can still be obtained when imposing connectivity constraints.

In Chapter 10, we look at two notions of fairness that have recently been introduced by Budish [2011] and that have become very influential: the maximin share guarantee (MMS) and envy-freeness up to one good (EF1). Both properties can easily be adapted to apply to the case with connectivity constraints. While there exists examples where MMS cannot be obtained in the world without connectivity constraints, we show that when the underlying graph does not have cycles (and so is a forest), a connected MMS allocation always exists and can be found efficiently. For EF1, several protocols have been introduced that achieve fairness according to this standard, but none of them can be adapted to honour connectivity constraints. Thus, we look at protocols with few cuts developed for cake-cutting and try to discretise them. Using this approach, we are able to prove that an EF1 allocation is guaranteed to exist when the underlying graph is a path, for either two or for three agents. For four or more agents, we use Sperner's lemma to show that there always exists an allocation which is envy-free up to two items (EF2).

In Chapter 11, we study Pareto-optimality and focus on additive valuations. Without connectivity constraints, it is easy to obtain a Pareto-optimal allocation: just assign each item to the agent who values it highest. This maximises utilitarian social welfare and is thus Pareto-optimal. However, this approach does not respect connectivity constraints, and the computational problem of finding a Pareto-optimum becomes interesting. We show that the problem can be solved efficiently when the item graph is very simple (a path or a star), but that it becomes NP-hard when the graph is a tree. We also show by an example that there are instances where no EF1 allocation is Pareto-optimal, when items are arranged on a path. This is in contrast to the situation with connectivity constraints, where it is known that EF1 and Pareto-optimality can be jointly achieved by maximising Nash welfare.

In Chapter 12, we briefly discuss the problem of finding a strategyproof mechanism that identifies EF1 allocations on a path, and find that there do not exist any. We connect this result to work on cake-cutting.

Part I.

Methodological Prelude: Impossibility Theorems and SAT Solving

1. The No-Show Paradox

An important class of voting rules is the class of *Condorcet extensions*. If there is an alternative which beats every other alternative by a majority in a pairwise comparison, then a Condorcet extension must choose that alternative. A seminal result by Moulin [1988b] shows that every Condorcet extension suffers from the *no-show paradox*: in some situations, voters can obtain a better election result if they abstain from the election. Moulin's proof works if there are at least 4 alternatives and 25 voters. We leverage SAT solving to obtain an elegant human-readable proof of Moulin's result that requires only 12 voters. Moreover, the SAT solver is able to construct a Condorcet-consistent voting rule that satisfies participation as well as a number of other desirable properties for up to 11 voters, proving the optimality of the above bound.

1.1. Introduction

The result that founded social choice theory is Arrow's Impossibility Theorem, and impossibility theorems have played a central role in the field ever since. They can push the field forward by motivating a search for a good way out, and they can lead to the design of new methods and mechanisms. An impossibility theorem proves that no mechanism can simultaneously satisfy a number of desirable properties. Thus, it is a formal way to prove that there is a real trade-off between these properties: we can only satisfy some, but not all. Most commonly, theorists have found trade-offs between the quality of the social choice, and the mechanism's vulnerability to strategic behaviour. Knowing trade-offs of this kind is important to guide the mechanism design process. Many otherwise attractive preference aggregation rules are vulnerable to strategic behaviour; an impossibility theorem can provide a justification for using the rule regardless. The converse is also often seen, where a rule designed to be resistant to strategic behaviour fails other desirable axioms, and again an impossibility result can suggest that this is not the mechanism designer's fault.

Annoyingly, impossibility theorems in social choice can be difficult to prove. The famous dictatorship-style results of Arrow and of Gibbard—Satterthwaite have mathematically elegant proofs based on ultrafilters, and there are dozens of alternative proofs published, employing many delightful techniques. But this is the exception. In most other cases, truly elegant proofs are not available. Instead, the proofs have a combinatorial feel, and depend on the construction of several related preference profiles, together with a path through these profiles which corresponds to a contradiction proof. These proofs are typically not insightful, and due to the type of mathematical environment considered, we usually can't hope for something more aesthetically pleasing.

A good example is a theorem about the no-show paradox proved by Hervé Moulin, which is the subject of this chapter. Fishburn and Brams [1983], while studying the popular voting rule STV (single transferable vote), noticed an odd behaviour of that rule: There are situations where a voter might bring about a worse outcome by participating in the election. They found a specific profile (see Figure 1.1) in which STV declares candidate b to be the election winner, yet if a new voter reporting preferences $a \succ b \succ c$ is added to the profile, STV now declares the alternative c as the election winner. Hence, it is better for the voter to abstain, since the voter

417	82	143	357	285	324
a	a	b	b	c	c
b	c	a	c	a	b
c	b	c	a	b	a

(a) The plurality scores for a, b, c are 499, 500, and 609, respectively. No-one has a majority, so a is eliminated. A majority of voters prefers b to c, so b wins under STV.

419	82	143	357	285	324
a	a	b	b	c	c
b	c	a	c	a	b
c	b	c	a	b	a

(b) Two additional abc voters join. Now STV eliminates b as the plurality loser, and a majority prefers c to a, so c wins under STV, which is worse for the new voters.

Figure 1.1.: Example of Fishburn and Brams [1983] showing a no-show paradox of STV. Column headers denote how many voters submit the shown ranking. (Much smaller examples exist.)

prefers b to c, giving rise to a "no-show paradox". Voting rules from the class of *scoring rules* (including plurality and Borda's rule) never exhibit this paradox. However, further investigation revealed that STV is not the only offender, and indeed almost all standard rules (except scoring rules) suffer from the paradox. In particular, all known rules satisfying Condorcet's consistency axiom[1] seemed to show the paradox. Moulin's theorem formalises this observation, and states that every voting rule either exhibits a no-show paradox, or it fails Condorcet-consistency.

Moulin proves his result by presenting a specific preference profile containing the votes of 25 voters over 4 alternatives. He then considers various combinations of voters abstaining from the election, and shows by a case analysis that each either results in a no-show paradox, or a failure of Condorcet-consistency. The construction of this proof is an awesome achievement, and the student attempting to design alternative proofs by hand might find that the effort ends in prolonged frustration. The student might end up thinking that this task would be better handled automatically by computers, and that computers could more efficiently search for those magic profiles that suffice to prove an impossibility. The student is in luck, because recent advances in AI and constraint solving mean that the task can indeed be automated.

Moulin's result proves that every Condorcet-consistent voting rule admits at least one situation where the no-show paradox occurs, but all these situations involve many voters. Indeed, Moulin's result does not rule out the existence of a Condorcet-consistent rule which avoids the paradox in all situations with fewer than 25 voters. Such a rule would be practically useful, since many decisions are reached in small committees. We say that a rule satisfies *participation* if it avoids the paradox. Our research question is this: What is the largest n for which there exists a Condorcet-consistent rule satisfying participation? Conversely, what is the smallest n for which Moulin's theorem can be proved?

By hand, it is possible to make Moulin's proof slightly more efficient, and prove the theorem for $n = 21 < 25$ voters, as noted in a Master's thesis at TU Munich [Kardel, 2014]. Further improvements are difficult; and ideally, we would want a matching lower bound, and it is completely unclear how to obtain non-trivial lower bounds. Using a computer-aided approach, we achieve these aims: we show that Condorcet-consistency and participation are incompatible for 4 alternatives and $n = 12$ voters, but that the same result does not hold with $n = 11$ voters.[2]

Out method is based on SAT solving. We fix a number n of voters and 4 alternatives, and then write down a large formula of propositional logic (in conjunctive normal form) whose satisfying assignments encode a voting rule satisfying Condorcet-consistency and participation. These

[1]An alternative x is a Condorcet winner if for every other alternative y, a majority of voters prefers x to y. Condorcet-consistency requires that the voting rule outputs the Condorcet winner, whenever one exists.

[2]For the case of $m = 3$ alternatives, Moulin [1988b] showed that the maximin rule with lexicographic tie-breaking is a Condorcet extension satisfying participation.

formulas use an explicit encoding: we introduce a variable for each possible input (a preference profile) and each possible output (an alternative) of the voting rule. Clauses are used to constrain the voting rule so that is satisfies our axioms. We can then pass the formula to a SAT solver. Even though our formulas are extremely large (with millions of variables and clauses), and even though the satisfiability problem is NP-complete, the resulting formula turns out to be solvable in a relatively short time. This is both due to recent progress in solving algorithms, and due to the structure of our formula, which seems to be easily solved.

Now, the SAT solver may either find that our formula is satisfiable or not. If the formula is satisfiable, a satisfying assignment encodes a voting rule satisfying our axioms. From the solver output, we can immediately extract an explicit example of such a rule, but only in form of a look-up table (which says, for each preference profile separately, what the outcome should be). In the case of unsatisfiability, we have obtained an impossibility result: no voting rule satisfies both of our requirements. Now, SAT solvers do not usually give a 'witness' of unsatisfiability. Some solvers can produce a trace that can be used to check the unsatisfiability by computer tools, but these traces tend to be very large. While these files can make for fun articles in the popular press (recent examples include a 13 GB proof of the Erdős Discrepancy Conjecture [Konev and Lisitsa, 2014], and a 200 TB solution to the Boolean Pythagorean Triples Problem [Heule et al., 2016]), they cannot be understood by humans. Excitingly, in our case, it is possible to obtain a *human-readable* proof of the impossibility. Our method is based on extracting a *minimal unsatisfiable set* (MUS), which is a minimal (with respect to set inclusion) selection of clauses which are already unsatisfiable. For some voting problems such as ours, where clauses tend to be 'local', these MUSes can be extremely small, and only contain a few dozen clauses. By interpreting these clauses as proof steps, it is then possible to construct (by hand) a contradiction proof in natural language. This proof will capture an impossibility for fixed n and m, though it is often possible to extend the proof to larger parameters using induction.

The approach described here is based on previous work by Tang and Lin [2009], Geist and Endriss [2011], Brandt and Geist [2016], and Brandl et al. [2015a]. A straightforward application of this methodology is not enough for the problem we study in this chapter, because the formulas produced grow linearly with the number of possible preference profiles, and there are too many such profiles for the parameter values we need to consider. To deal with this, we used several new techniques. In particular, we extracted knowledge from computer-generated proofs of weaker statements and then used this information to guide the search for proofs of more general statements. In the following sections, we will describe this methodology and present the proof.

1.2. Related Work

The no-show paradox was first observed by Fishburn and Brams [1983] for the STV voting rule. Ray [1986], Lepelley and Merlin [2000], and Brandt et al. [2019] investigate how frequently this phenomenon occurs on random profiles. The main impossibility theorem addressed in this chapter is due to Moulin [1988b] and requires at least 25 voters. This bound was improved to 21 voters by Kardel [2014]. Simplified presentations of Moulin's proof are given by Schulze [2003] and Smith [2007]. Holzman [1988] and Sanver and Zwicker [2009] strengthen Moulin's theorem by weakening Condorcet-consistency and participation, respectively. Duddy [2014a] shows the incompatibility of Condorcet-consistency and weaker notions of participation when allowing weak preferences. Pérez [2001] considers these notions in the context of set-valued voting rules and shows that all common Condorcet extensions except the maximin rule and Young's rule violate these properties. Jimeno et al. [2009] prove variants of Moulin's theorem for set-valued voting rules based on the optimistic and the pessimistic preference extension. Determining optimal bounds on the number of voters for these paradoxes has been recognised as an open problem. For example, Pérez notes that "a practical question, which has not been dealt with here, refers to the number of candidates and voters that are necessary to invoke the studied paradoxes" [Pérez, 2001, p. 614] and Duddy [2014a] concludes that "we do not know what upper bound is imposed on the number of potential voters by the conjunction of Condorcet consistency and [...] the participation principle in the case of linear orderings. And these upper bounds may fall as the number of candidates rises. These are important open problems since voting is often conducted by small groups of individuals." The influence of the number of voters and alternatives has recently also been studied in other contexts of social choice theory [see, e.g., Campbell and Kelly, 2010, Campbell et al., 2012].

When assuming that voters have incomplete preferences over sets or lotteries, participation and Condorcet-consistency can be satisfied simultaneously and positive results for common Condorcet-consistent voting rules (such as the top cycle) have been obtained by Brandt [2015] and Brandl et al. [2015a,b]. A particularly positive result was recently obtained for maximal lotteries, a probabilistic Condorcet extension due to Fishburn [Brandl et al., 2019a].

The computer-aided techniques in this chapter are inspired by Tang and Lin [2009], who reduced well-known impossibility results from social choice theory (such as Arrow's theorem) to finite instances, which can then be checked automatically. This methodology has been extended and applied to new problems by Geist and Endriss [2011], Brandt and Geist [2016], and Brandl et al. [2015a]. Geist and Peters [2017] give a survey of the technique and the results obtained with it. More generally, SAT solvers have also proven to be quite effective for other problems in economics. A prominent example is the ongoing work by Fréchette et al. [2016] in which SAT solvers are used for the development and execution of the FCC's upcoming reverse spectrum auction.

Our approach can be seen as an instance of *automated mechanism design* [Conitzer and Sandholm, 2002], which uses algorithms to construct a mechanism satisfying properties such as truthfulness and budget balance for a specific problem instance. A problem instance would be specified by a collection of possible agent types as well as a prior distribution over them, and a typical aim would be to find a mechanism that maximises a notion of expected social welfare or expected revenue subject to truthfulness. Such a problem can often be solved using an integer linear programming formulation with a similar structure as the SAT formulations that we use. In our applications, there is no prior, and we are interested only in the existence question rather than an optimisation of some objective function. Also, while our aim is to find an impossibility theorem, the standard outlook of automated mechanism design is more positive: the hope is that known impossibility theorems are unlikely to apply to the specific instance the mechanism designer is interested in [Conitzer and Sandholm, 2003]. While there may not be known general recipes for obtaining a good mechanism for the instance, an automated approach can find an optimal mechanism tailored to the problem at hand. However, the automated mechanism design literature contains some examples of impossibilities obtained from integer linear programming, for example in the context of mechanisms for deciding charitable donations [Conitzer and Sandholm, 2011, Footnote 13]. The initial paper on automated mechanism design [Conitzer and Sandholm, 2002] studies the computational complexity of the problem of finding an optimal mechanism for a given instance, and finds hardness results when aiming for deterministic mechanisms. A similar question would be interesting for problems of the kind we study: for example, given an explicit list of possible preference profiles, what is the complexity of deciding whether there is a strategyproof and Pareto-efficient voting rule defined only on those profiles?

1.3. Preliminaries

Let A be a set of m alternatives. Let N be a set of n voters, not all of which need to participate in the election. By $\mathcal{N} := 2^N \setminus \{\emptyset\}$ we denote the set of *electorates*, i.e., non-empty subsets of N. We will mostly consider the case of 4 alternatives, and take $A = \{a, b, c, d\}$. We use the labels

x, y for generic elements of A.

A *linear order* $\succcurlyeq$ is a complete, antisymmetric, transitive binary relation over A. We write $\succ$ for the strict (irreflexive) part of $\succcurlyeq$. The set of all linear orders over A is denoted by $A!$. For brevity, we denote by *abcd* the preference relation $a \succcurlyeq_i b \succcurlyeq_i c \succcurlyeq_i d$, eliding the identity of voter i, and similarly for other preferences. For a linear order $\succcurlyeq$, we write $\text{top}(\succcurlyeq)$ for the most-preferred element of A, so that $\text{top}(\succcurlyeq) \succcurlyeq x$ for all $x \in A$.

A *preference profile* P is a function from an electorate $N' \in \mathcal{N}$ to the set of linear orders $A!$, assigning to every voter $i \in N$ a preference relation. Thus, the set of all profiles is $\bigcup_{N' \in \mathcal{N}} A!^{N'}$. We will always write $\succcurlyeq_i$ for the linear order $P(i)$, and it will be clear from the context which profile P is meant.

To define the participation axiom, we need notation for adding or removing a voter from a profile. For a preference profile $P \in A!^{N'}$ with $(i, \succcurlyeq_i) \in P$, and $j \in N \setminus N'$, $\succcurlyeq_j \in A!$, we write

$$P - i := P \setminus \{(i, \succcurlyeq_i)\}, \qquad\qquad P + (j, \succcurlyeq_j) := P \cup \{(j, \succcurlyeq_j)\}.$$

If the identity of the voter is clear or irrelevant, we refer to $P - i$ by $P - \succcurlyeq_i$, and write $P + \succcurlyeq_j$ instead of $P + (j, \succcurlyeq_j)$. If k voters with the same preferences $\succcurlyeq_i$ are to be added or removed, we write $P + k \cdot \succcurlyeq_i$ and $P - k \cdot \succcurlyeq_i$, respectively.

We wish to study voting rules, which, given a preference profile, decide on a winning alternative. We consider two formal definitions of voting rules, depending on whether all voters are guaranteed to participate, or not.

Definition 1.1.

- A *variable-electorate voting rule* is a function $f : \bigcup_{N' \in \mathcal{N}} A!^{N'} \to A$.
- A *fixed-electorate voting rule* is a function $f : A!^N \to A$.

A variable-electorate voting rule induces a fixed-electorate voting rule in the obvious way. Note that our definitions require f to be *resolute*, meaning that f returns exactly one winner for each profile P. In this chapter, we study variable-electorate voting rules, while the next chapter considers a similar question that is relevant for fixed electorates.

Definition 1.2. A variable-electorate voting rule f satisfies *participation* if all voters always weakly prefer voting to not voting, i.e., if $f(P) \succcurlyeq_i f(P - i)$ for all $P \in A!^{N'}$ and $i \in N'$ with $N' \in \mathcal{N}$.

Equivalently, participation requires that for all preference profiles P not including voter j, we have $f(P + \succcurlyeq_j) \succcurlyeq_j f(P)$. By induction, participation also requires that $f(P + k \cdot \succcurlyeq_j) \succcurlyeq_j f(P)$ for any $k \geqslant 1$: thus, a group of agents with identical preferences weakly prefers joining.

Given a profile $P \in A!^{N'}$, we say that $a \in A$ is the *Condorcet winner* in P if $|\{i \in N' : a \succ_i b\}| > |\{i \in N' : b \succ_i a\}|$ for all $b \in A \setminus \{a\}$. Thus, a Condorcet winner wins against every other alternative in a pairwise majority comparison. If a Condorcet winner exists in a profile, then it is clearly unique.

Definition 1.3. A *Condorcet extension* is a voting rule that selects the Condorcet winner whenever it exists. Thus, f is a Condorcet extension if for every preference profile P that admits a Condorcet winner x, we have $f(P) = x$. We say that f is *Condorcet-consistent*.

The *majority margins* of P is the map $g_P : A \times A \to \mathbb{Z}$ with $g_P(x, y) = |\{i \in N' \mid x \succcurlyeq_i y\}| - |\{i \in N' \mid y \succcurlyeq_i x\}|$. If $g_P(x, y) > 0$ for some alternatives $x, y \in A$, then a majority of voters prefers x to y. The majority margins can be viewed as the adjacency matrix of a *weighted tournament* T_P. We say that a preference profile P *induces* the weighted tournament T_P. An example of a weighted tournament is shown in Figure 1.2. An alternative x is the *Condorcet winner* in P if and only if $g_P(x, y) > 0$ for all $y \in A \setminus \{x\}$.

2	3	3	2
a	b	c	d
b	d	a	c
d	c	b	a
c	a	d	b

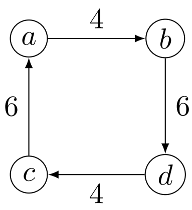


Figure 1.2.: An example of a preference profile and its majority margins shown as a weighted tournament. The number at the top of each column indicates how many voters submit the relevant ranking; thus there are 10 voters in total in this example. The weighted tournament is the digraph with vertex set A, and with an arc $x \to y$ if and only if $g_P(x, y) > 0$; we label the arc with $g_P(x, y)$. In the example, 7 voters prefer a to b and 3 voters prefer b to a, and so the arc from a to b is labelled by $4 = 7 - 3$. Also, 5 voters prefer a to d and 5 voters prefer d to a, and so there is a majority tie; thus we do not draw an arc between a and d.

1.4. Method: SAT Solving for Computer-Aided Proofs

The results of this chapter were obtained with the aid of a computer. In this section, we describe the method that we employed. The main tool in our approach is an encoding of our problem into propositional logic. We then use SAT solvers to decide whether there exists a Condorcet extension satisfying participation. If the answer is yes, the solver returns an explicit voting rule with the desired properties. If the answer is no, we use a process called *MUS extraction* to find a short certificate of this fact which can be translated into a human-readable proof. By successively proving stronger theorems and using the insights obtained through MUS extraction, we arrived at results as presented in their full generality in this chapter.

1.4.1. SAT Encoding

Our aim is to construct a family $(\varphi_n)_{n\geqslant 1}$ of propositional formulas such that φ_n is satisfiable if and only if there exists a voting rule f that satisfies Condorcet-consistency and participation if $m = 4$ and $|N| = n$. We take $A = \{a, b, c, d\}$ throughout the discussion in this section. A natural encoding proceeds like this: Generate all profiles over 4 alternatives with at most n voters. For each such profile P, introduce 4 propositional variables $x_{P,a}$, $x_{P,b}$, $x_{P,c}$, $x_{P,d}$, where the intended meaning of $x_{P,a}$ is

$$x_{P,a} \text{ is set true} \iff f(P) = a.$$

We then add clauses requiring that for each profile P, $f(P)$ takes exactly one value, and we add clauses requiring f to be Condorcet-consistent and satisfy participation.

Sadly, the encoding sketched above is not tractable for the values of n that we are interested in: the number of variables and clauses used grows as $\Theta(24^n)$, because there are $4! = 24$ possible preference relations over 4 alternatives and thus 24^n profiles with n voters. For $n = 7$, this leads to more than 400 billion variables, and for $n = 15$ we exceed 10^{22} variables.

To escape this combinatorial explosion, we will temporarily restrict our attention to *pairwise* voting rules. These are voting rules whose outcome only depends on the majority margins of the input profile. Recall that we write T_P for the weighted tournament induced by P. Formally, a voting rule f is pairwise if $f(P) = f(P')$ whenever $T_P = T_{P'}$. Examples of pairwise voting rules are Kemeny's rule, tournament solutions like Copeland, and Borda. To specify a pairwise rule, we only need to decide the output for each possible weighted tournament. Abusing notation, for a pairwise rule, for each weighted tournament T, we write $f(T)$ for the outcome of f at each profile P with $T = T_P$.

The number of weighted tournaments induced by profiles with n voters grows much slower than the number of profiles. Our computer enumeration[3] suggests a growth of order about 1.5^n. This much more manageable (yet still exponential) growth allows us to consider problem instances up to $n \approx 16$.

Other than referring to (weighted) tournaments instead of profiles, our encoding into propositional formulas now proceeds exactly like before. For each tournament T, we introduce the variables $x_{T,a}$, $x_{T,b}$, $x_{T,c}$, $x_{T,d}$, one for each alternative. We require that at each T, exactly one alternative is chosen, and so we define the formula

$$\varphi_{\text{functionality}}(T) \equiv \big(x_{T,a} \vee x_{T,b} \vee x_{T,c} \vee x_{T,d}\big) \wedge \bigwedge_{x \neq y} (\neg x_{T,x} \vee \neg x_{T,y})$$

The first part of the formula requires that at least one alternative is selected, and the second part requires that at most one alternative is selected. With our intended interpretation of the variables $x_{T,x}$, all satisfying assignments of $\bigwedge_T \varphi_{\text{functionality}}(T)$ are functions from tournaments into $\{a, b, c, d\}$.

Since we are interested in voting rules that satisfy participation, we also need to encode this property. To this end, let $T = T_P$ be the tournament induced by P and let $\succcurlyeq$ be a preference relation. Define $T + \succcurlyeq \, := T_{P+\succcurlyeq}$. (The tournament $T + \succcurlyeq$ is independent of the choice of P.) We define

$$\varphi_{\text{participation}}(T, \succcurlyeq) \equiv \bigwedge_{x} \Big(x_{T,x} \to \bigvee_{y \succcurlyeq x} x_{T+\succcurlyeq,y}\Big).$$

Requiring f to be Condorcet-consistent is straightforward: if tournament T admits b as the Condorcet winner, we add

$$\varphi_{\text{condorcet}}(T) \equiv \neg x_{T,a} \wedge x_{T,b} \wedge \neg x_{T,c} \wedge \neg x_{T,d},$$

and we add similar formulas for each tournament that admits a Condorcet winner. Then the models of the conjunction of all the $\varphi_{\text{functionality}}$, $\varphi_{\text{participation}}$, and $\varphi_{\text{condorcet}}$ formulas are precisely the pairwise voting rules satisfying Condorcet-consistency and participation.

By adapting the $\varphi_{\text{condorcet}}$ formulas, we can impose more stringent conditions on f. For example, we can use this to exclude Pareto-dominated alternatives, and to require f to always pick from the top cycle.

1.4.2. SAT Solving and MUS Extraction

The formula we have obtained above are all given in *conjunctive normal form* (CNF), and thus can be evaluated without further transformations by any off-the-shelf SAT solver. In order to physically produce a CNF formula as described, we employ a straightforward Python script that performs a breadth-first search[4] to discover all weighted tournaments with up to n voters (see Algorithm 1 for a schematic overview of the program). The script outputs a CNF formula in the standard DIMACS format, and also outputs a file that, for each variable $x_{T,x}$, records the tournament T and alternative x it denotes. This is necessary because the DIMACS format uses uninformative variable descriptors (consecutive integers) and memorising variable meanings allows us to interpret the output of the SAT solver.

[3]For $n = 1, \ldots, 14$, there are 24, 219, 1 136, 4 175, 12 216, 30 429, 67 264, 135 621, 254 200, 449 031, 755 184, 1 218 659, 1 898 456, 2 868 825 weighted tournaments induced by profiles with n voters, respectively.

[4]The large memory requirements of a BFS were the largest bottleneck in our approach. Sadly, a DFS does not work in our context.

Algorithm 1 Generate formula for up to n voters

$T_0 \leftarrow$ {weighted tournament on $\{a, b, c, d\}$ with all edges having weight 0}.
for $k = 1, \ldots, n$ **do**
  $T_k \leftarrow \emptyset$
  for $T \in T_{k-1}$ **do**
    for $\succcurlyeq \in A!$ **do**
      Calculate $T' := T + \succcurlyeq$
      if T' has not been seen previously, i.e., $T' \notin T_0 \cup \cdots \cup T_k$ **then**
        Add T' to T_k
        Write $\varphi_{\text{functionality}}(T'), \varphi_{\text{condorcet}}(T')$
      Write $\varphi_{\text{participation}}(T, \succcurlyeq)$

As an example, the output formula for $n = 15$ in DIMACS format has a size of about 7 GB and uses 50 million variables and 2 billion clauses, taking 6.5 hours to write. Plingeling [Biere, 2013], a popular SAT solver, solves this formula in 50 minutes of wall clock time, half of which is spent parsing the formula.

In case a given instance is satisfiable, the solver returns a satisfying assignment, giving us an existence proof and a concrete example for a voting rule satisfying participation (and any further requirements imposed). In case a given instance in unsatisfiable, we would like to have a short certificate of this fact as well. One possibility for this is having the SAT solver output a resolution proof (usually using the standard DRUP or DRAT formats). This yields a machine-checkable proof, but has two major drawbacks: the generated proofs can be uncomfortably large, and they do not give human-accessible insights about *why* the formula is unsatisfiable.

We handle this problem by computing a *minimal unsatisfiable subset (MUS)* of the unsatisfiable CNF formula. An MUS is a subset of the clauses of the original formula which itself is unsatisfiable, and is minimal with respect to set inclusion: removing any clause from it yields a satisfiable formula. We used the tools MUSer2 [Belov and Marques-Silva, 2012] and MARCO [Liffiton et al., 2016] to extract MUSes.

Note that for purposes of extracting human-readable proofs, it is desirable for the MUS to be as small as possible, and also to refer to as few different tournaments as possible. The first issue can be addressed by running the MUS extractor repeatedly, instructing it to order clauses randomly (note that clause sets of different cardinalities can be minimally unsatisfiable with respect to set inclusion); similarly, we can use tools like MARCO to enumerate all MUSes and look for small ones. The second issue can be addressed by computing a *group MUS*: here, we partition the clauses of the CNF formula into *groups*; then, we look for a minimal set of groups whose union is unsatisfiable. In our case, the clauses referring to a given tournament T form a group. In practice, finding a group MUS first and then finding a standard (clause-level) MUS within the group MUS yielded sets of size much smaller than MUSes returned without the intermediate group-step (often by a factor of 10).

To translate an MUS into a human-readable proof, we created another program that visualises the MUS in a convenient form. Roughly, the visualisation program proceeds as follows: an MUS contains two types of clauses. Some (like $\varphi_{\text{functionality}}, \varphi_{\text{condorcet}}$) refer to a single tournament; we draw a vertex for each such tournament mentioned in the MUS. On the other hand $\varphi_{\text{participation}}(T, \succcurlyeq)$ clauses *connect* the outcomes at two tournaments together; we draw an arc between them for each $\varphi_{\text{participation}}$ clause occurring in the MUS, and label the arc with $\succcurlyeq$. We also annotate vertices with their Condorcet winners if $\varphi_{\text{condorcet}}$ clauses are mentioned. Indeed, this program outputs essentially the 'proof diagram' shown in Figure 1.3. We think that interpreting such a diagram is quite natural. More importantly, the automatically produced graphs allowed us to quickly judge the quality of an extracted MUS. As described, each MUS induces a directed graph whose vertices are profiles. One way to quickly judge how 'complicated' the corresponding impossibility proof is, is by counting the number of high-degree nodes in that graph. Roughly, at each node that is connected to more than two other nodes we will have to perform a case distinction. Thus, a proof with fewer high-degree vertices will be easier to understand.

1.4.3. Incremental Proof Discovery

The SAT encoding described in Section 1.4.1 only concerns pairwise voting rules, yet our negative result does not require or use this assumption. We were able to remove this assumption, by going through multiple rounds of generating and evaluating SAT formulas, extracting MUSes, and using the insights generated by this as 'educated guesses' to solve more general problems.

Following the process as described so far led to a proof that for 4 alternatives and 12 voters, there is no pairwise Condorcet extension that satisfies participation. That proof used the assumption of pairwiseness, i.e., it assumed that the voting rule returns the same alternative on profiles inducing the same weighted tournament. However, intriguingly, the preference profiles mentioned in the proof did not contain all $4! = 24$ possible preference relations over $\{a, b, c, d\}$. In fact, the proof only used 10 of the possible orders. Further, each profile appearing in the proof included $P_{\text{base}} = \{abdc, bdca, cabd, dcab\}$ as a subprofile. As we argued at the start of Section 1.4.1, it is intractable to search over the entire space of preference profiles. On the other hand, it is much easier to merely search over all extensions of P_{base} that contain at most $n = 12$ voters and only contain copies of the 10 orders previously identified. The SAT formula produced by doing exactly this turned out to be unsatisfiable, and a small MUS extracted from it gave rise to Theorem 1.5 below.

1.5. Main Result

Before we come to the proof of our main result, we prove a lemma which can be seen as an 'induction step'. The proof we obtain from the SAT solver works for a fixed number of alternatives ($m = 4$), and the following lemma allows us to extend the reach of the proof to any number of alternatives greater than that.

Lemma 1.4. *Suppose that f is a (variable-electorate) Condorcet extension satisfying participation, and let P be a preference profile. Let $B \subsetneq A$ be a set of alternatives such that each voter ranks every $x \in B$ below every $y \in A \setminus B$. Then $f(P) \notin B$.*

Proof. We say that the members of B are *bottom alternatives.* We prove the lemma by induction on the number of voters $|N|$ in P. If P consists of a single voter i, then, since f is a Condorcet extension, $f(P)$ must return i's top choice,[5] which is not a bottom alternative. If P consists of at least 2 voters, and $i \in N$, then by participation $f(P) \succcurlyeq_i f(P - i)$. If $f(P)$ were a bottom alternative, then so would be $f(P - i)$, contradicting the inductive hypothesis. □

We are now in a position to state and prove our main claim that Condorcet extensions cannot avoid the no-show paradox for 12 or more voters (when there are at least 4 alternatives), and that this result is optimal.

The following computer-aided proof can be understood by examining the corresponding 'proof diagram', shown in Figure 1.3. An arrow such as $P \xrightarrow{+abcd} P'$ indicates that profile P' is obtained from P by adding a voter *abcd*, and is read as "if one of the bold green alternatives (here *ab*) is selected at P, then one of them is selected at P'" (by participation). The 'leaves'

[5] We are not using the full force of Condorcet-consistency here, and only require a notion of unanimity or faithfulness.

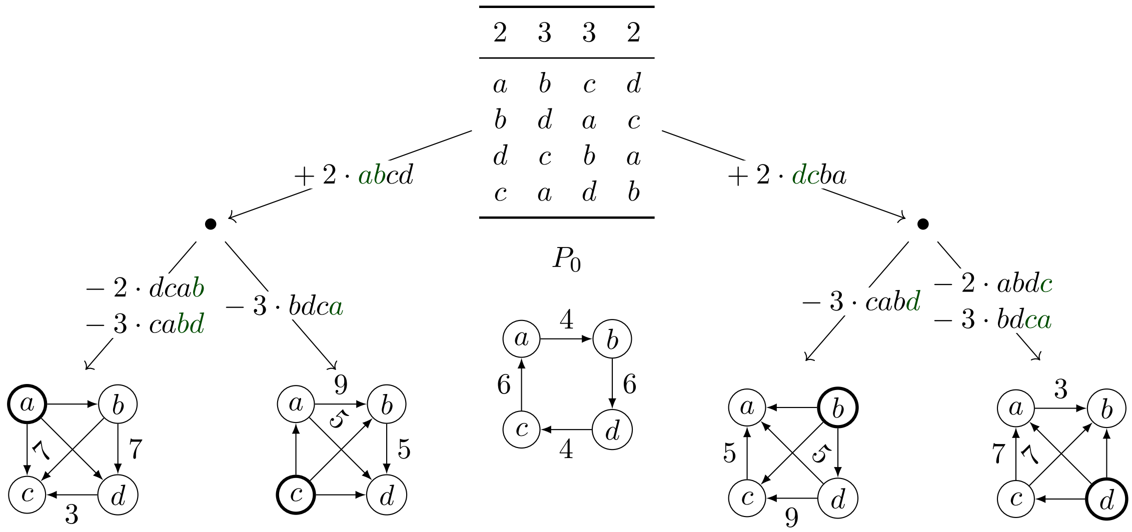


Figure 1.3.: Computer-aided proof of Theorem 1.5 in graphical form, showing that there is no Condorcet extension that satisfies participation for $m \geqslant 4$ and $n \geqslant 12$.

in the diagrams are profiles admitting a Condorcet winner, and we then print the weighted tournament associated with this profile (unlabelled arcs have weight 1, arcs not printed have weight 0). The Condorcet winner is shown in a bold circle. In each case, the Condorcet winner contradicts what is required by the participation axiom.

Theorem 1.5. *There is no variable-electorate Condorcet extension that satisfies participation for $m \geqslant 4$ and $n \geqslant 12$.*

Proof. We first consider the case $m = 4$. The proof follows the structure depicted in Figure 1.3. Let $P_0 = 2 \cdot abdc + 3 \cdot bdca + 3 \cdot cabd + 2 \cdot dcab$ be the preference profile with 10 voters shown in the figure, and assume for a contradiction that f is a Condorcet extension that satisfies participation.

Since P_0 remains fixed after relabelling alternatives according to $abcd \mapsto dcba$ and reordering voters, we may assume without loss of generality that $f(P_0) \in \{a, b\}$. (An explicit proof in case $f(P) \in \{c, d\}$ is indicated in Figure 1.3.)

Let $P_1 = P_0 + 2 \cdot abcd$ be the profile obtained from P_0 after two new voters with preferences $abcd$ join. Since f satisfies participation, and $f(P_0) \in \{a, b\}$, we must have that $f(P_1) \in \{a, b\}$, since otherwise the new voters would be worse off by joining the electorate.

We now perform a case distinction on the value of $f(P_1)$.

Suppose $f(P_1) = a$. Let $P_2 = P_1 - 3 \cdot bdca$. Since f satisfies participation, $f(P_2) = a$, since otherwise the two leaving voters would be better off abstaining. Calculating the majority margins of profile P_2 (see the second tournament from the left in Figure 1.3), we find that c is a Condorcet winner in P_2. Thus, the fact that $f(P_2) = a$ contradicts that f is a Condorcet extension.

Suppose instead that $f(P_1) = b$. Let $P_3 = P_1 - 2 \cdot dcab$; by participation, we have that $f(P_3) = b$ since otherwise the leaving voters are better off. Next let $P_4 = P_3 - 3 \cdot cabd$. Again by participation, since $f(P_3) = b$, we must have $f(P_4) \in \{b, d\}$, or otherwise the leaving voters would be better off. However, in profile P_4, alternative a is a Condorcet winner (see the left-most tournament in Figure 1.3), and so we have a contradiction to f being a Condorcet extension.

Since either case leads to a contradiction, there can be no voting rule f with the desired properties.

If $m > 4$, we introduce new alternatives $x_1, x_2, \ldots, x_{m-4}$ and place them to the bottom of the voters in P_0 and in all other votes. By Lemma 1.4, f chooses from $\{a, b, c, d\}$ at each step, allowing the proof to go through as for the case $m = 4$. ☐

```
a,#1,(1,1,1,1,1,1)
a,#1,(1,1,1,1,1,-1)
a,#1,(1,1,1,-1,1,1)
a,#1,(1,1,1,-1,-1,1)
a,#1,(1,1,1,1,-1,-1)
a,#1,(1,1,1,-1,-1,-1)
b,#1,(-1,1,1,1,1,1)
b,#1,(-1,1,1,1,1,-1)
b,#1,(-1,-1,1,1,1,1)
b,#1,(-1,-1,-1,1,1,1)
b,#1,(-1,1,-1,1,1,-1)
b,#1,(-1,-1,-1,1,1,-1)
c,#1,(1,-1,1,-1,1,1)
c,#1,(1,-1,1,-1,-1,1)
```

```
a,#11,(9,11,3,9,1,-9)
a,#11,(11,9,3,7,1,-9)
c,#11,(5,-9,-1,-11,-1,7)
c,#11,(5,-9,-1,-11,-1,5)
c,#11,(3,-11,-1,-9,1,7)
c,#11,(3,-11,-3,-9,1,7)
c,#11,(3,-11,-3,-11,-1,7)
b,#11,(-1,3,-5,-3,5,-3)
b,#11,(-3,3,-7,-3,5,-3)
b,#11,(-3,1,-7,-3,5,-3)
c,#11,(-3,1,-5,-5,5,-1)
a,#11,(3,7,11,-3,9,11)
a,#11,(3,7,11,-3,9,9)
a,#11,(3,7,11,-5,9,11)
```

Figure 1.4.: Excerpt of look-up table giving a pairwise Condorcet extension satisfying participation for $n \leqslant 11$ voters (from Theorem 1.6). Each row lists a weighted tournament as $(g_P(a,b), g_P(a,c), g_P(a,d), g_P(b,c), g_P(b,d), g_P(c,d))$ with a chosen alternative, and with the number of voters inducing the tournament.

A remarkable and unexpected aspect of the computer-aided proof above is its use of symmetry. The preference profile P_0 considered at the start of the proof has self-symmetries which allow us to exclude certain outcomes without loss of generality. Such symmetries do not appear in the original hand-made proof by Moulin [1988b]. In hindsight, it makes intuitive sense that the most efficient proof (in terms of the number of voters required) is symmetric in this way, and we will see similar symmetries in the computer-aided proofs found throughout this thesis.

The following result says that our bound on the number of voters is tight, because our propositional formula for $n = 11$ voters is satisfiable. We do not have a good understanding of the rule found by the solver (though see the discussion below for some statistics), and the rule could be ill-behaved. However, a useful feature of our computer-aided approach is that we can easily add additional requirements for the desired voting rule. Our formula for $n = 11$ voters remains satisfiable even after we add the requirement that no Pareto-dominated alternatives are selected, that no Condorcet loser is selected, and that the selected alternative is contained in the *top cycle* (also known as the *Smith set*). For definitions of these common social choice properties, see Fishburn [1977].

Theorem 1.6. *There is a Condorcet extension f that satisfies participation for $m = 4$ and $n \leqslant 11$. Moreover, f is pairwise, Pareto-optimal, and a refinement of the top cycle.*

The Condorcet extension f is given as a look-up table, which is derived from the output of a SAT solver. The look-up table lists all 1 204 215 weighted tournaments inducible by up to 11 voters and assigns each an output alternative (see Figure 1.4 for an excerpt). The relevant text file has a size of 28 MB (gzipped 4.5 MB). We deposited this file in the public repository *Harvard Dataverse*, together with a Python script verifying that it describes a voting rule that satisfies participation [Brandt et al., 2016c].

Comparing this voting rule with known voting rules, it turns out that it selects one of the maximin winners in 99.8% and one of the Kemeny winners in 98% of all weighted tournaments. Note that there can be multiple maximin and Kemeny winners in a given profile; the rule agrees with the maximin rule with lexicographic tie-breaking on 95% of weighted tournaments. The similarity with the maximin rule is interesting because maximin satisfies participation for $m = 3$ alternatives [Moulin, 1988b]. A well-documented flaw of the maximin rule is that it fails to be a

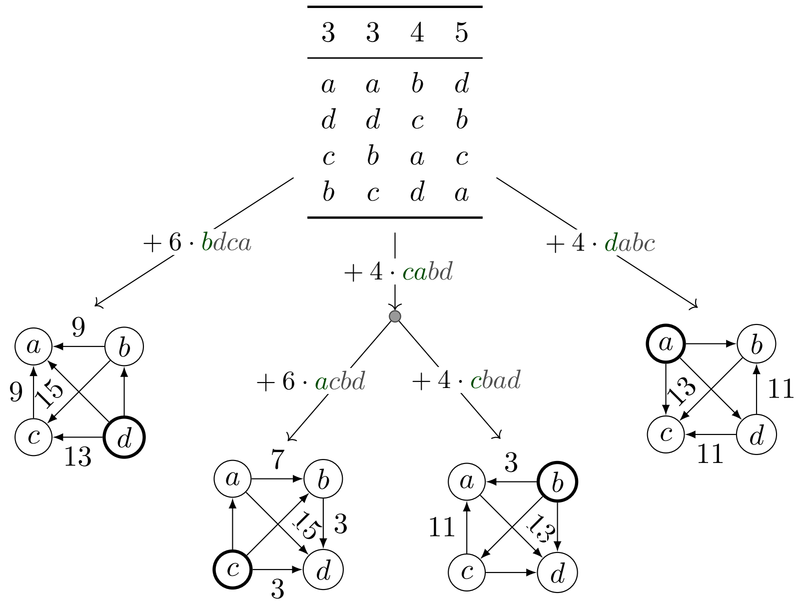


Figure 1.5.: Proof diagram for Moulin's original proof.

refinement of the top cycle (and may even return Condorcet losers), while our rule avoids these pathologies: Our computer-generated rule always picks from the top cycle while it remains very close to the maximin rule. Also, 80% of the weighted tournament inducible by profiles with $n \leqslant 11$ voters admit a Condorcet winner, which uniquely determines the output of the rule; this can be used to reduce the size of the look-up table.

1.6. Conclusions

We have given a tight result about the number of voters required to prove Moulin's theorem. Inspecting the proof and its use of a self-symmetric profile, one might suspect that the crux of the impossibility is our assumption that voting rules are resolute. If one were to allow ties between several alternatives, the impossibility might go away. Now, it is not completely clear how to define participation for irresolute voting rules, but in the published version of this chapter [Brandt et al., 2017], we consider two options (optimistic and pessimistic participation) and show that they too lead to impossibility, albeit requiring a greater number of voters.

The graphical representation of our proof can also be used to compactly represent Moulin's original proof [Moulin, 1988b]. To do this, several applications of Moulin's Claim 3 have to be decoded into the explicit votes that are added to the profiles under consideration. This was already done in the expositions of Schulze [2003] and Smith [2007] and the resulting proof diagram is shown in Figure 1.5.

Note that our negative result (Theorem 1.5) applies to every $m \geqslant 4$, but our positive result (Theorem 1.6) only works for $m = 4$. We were unable to check using our approach whether no-show paradoxes occur with even fewer voters when the number of alternatives grows, because the branching factors are too large when there are 5 alternatives (and hence $5! = 120$ possible preference relations). We leave this question for the (possibly far) future. Further, it would be interesting to gain a deeper understanding of the computer-generated Condorcet extension that satisfies participation for up to 11 voters. So far, we only know that it (slightly) differs from all Condorcet extensions that are usually considered in the literature. As a first step, it would be desirable to obtain a representation of this rule that is more concise than a look-up table.

2. The Preference Reversal Paradox

In this chapter, we consider a paradox closely related to the no-show paradox of the last chapter. We prove that every Condorcet-consistent voting rule can be manipulated by a voter who completely reverses their preference ranking, assuming that there are at least 4 alternatives. For the case of precisely 4 alternatives, we exactly characterise the number of voters for which this impossibility result can be proved. This result is one ingredient of the "disjunctive Gibbard–Satterthwaite theorem" that we prove in the following chapter.

2.1. Introduction

The Gibbard–Satterthwaite Theorem establishes that every non-trivial voting rule can be manipulated by voters through misrepresenting their preferences. In this chapter, we will see that Condorcet extensions (voting rules that select the Condorcet winner if one exists) suffer from a particularly offensive failure of strategyproofness: all of them can be manipulated by a voter who completely reverses their preference ranking. For example, such a voting rule might designate c to be the winning alternative if voter i truthfully reports the ordering $a \succ_i b \succ_i c \succ_i d$, but choose b as the winner if voter i instead reports the ordering $d \succ_i c \succ_i b \succ_i a$. Since i truthfully prefers b to c, this is a successful manipulation, which one might consider surprising given that i misreported every possible pairwise comparison. We will say that voting rules that are manipulable in this way suffer from the *preference reversal paradox*. While all Condorcet extensions exhibit this paradox, scoring rules (such a plurality and Borda's rule) are immune.

Preference reversal paradoxes were first introduced by Sanver and Zwicker [2009] in their study of monotonicity properties; they say that voting rules which avoid this paradox satisfy *half-way monotonicity*.[1] As Sanver and Zwicker [2009] show, half-way monotonicity is a weaker property than *participation*, an axiom stating that a voter cannot obtain a strictly better result by abstaining from an election; equivalently, participation says that voting truthfully guarantees a (weakly) better result than not voting at all. As we saw in the last chapter, participation is incompatible with Condorcet-consistency, so that Condorcet extensions must suffer from the no-show paradox. This result is often interpreted as showing that all Condorcet extensions are *manipulable* (through abstention). Notice, however, that this notion of manipulation (referring to electorates of different sizes) is quite different from the fixed-electorate manipulations that are the subject of the Gibbard–Satterthwaite Theorem, where a voter changes their preference ordering in some way [see also Núñez and Sanver, 2017]. We will see that half-way monotonicity, which is both weaker than participation and weaker than strategyproofness in the Gibbard–Satterthwaite sense, is already incompatible with Condorcet-consistency.

This result first appeared in Sanver and Zwicker [2009] who gave a proof that, for 4 or more alternatives and for sufficiently many voters, Condorcet extensions must fail half-way monotonicity. However, their proof contains an arithmetical mistake[2] that is non-trivial to fix. The proof technique also is only able to establish an impossibility for electorates containing a

[1] They chose this name because half-way monotonicity is a weaker version of their notion of *one-way* monotonicity.

[2] In the last paragraph of the proof of their Theorem 5.2, they calculate that $n^*(Q) = 30 + 8$, when in fact $n^*(Q) = 30 + 4 \cdot m! \gg 38$ which makes their "Condition M" inapplicable to profile Q. This problem was noticed by Wei Yu and Tokuei Higashino (Zwicker, private communication).

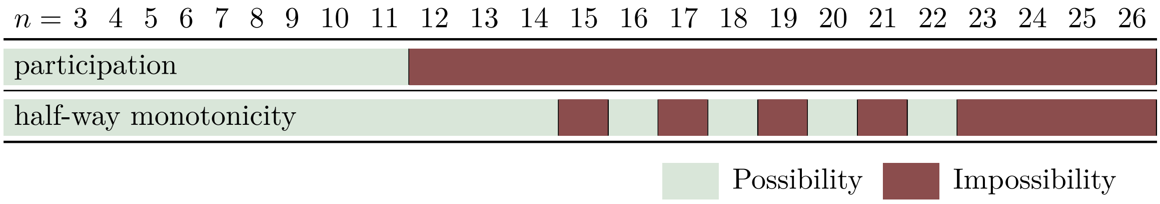


Table 2.1.: Numbers n of voters for which Condorcet extensions can satisfy participation or half-way monotonicity, when there are exactly $m = 4$ alternatives.

sufficiently large *even* number of voters. Further, their proof requires at least 702 voters to go through, this bound growing exponentially as the number of alternatives increases,[3] which leaves open the question of whether the preference reversal paradox is a problem in practical voting situations with moderate numbers of voters.

We aim to give a new and direct proof of the impossibility. We ask a similar question as in the last chapter, namely to find the exact number of voters required for the impossibility to hold. Since half-way monotonicity is weaker than participation, we would expect more than 12 voters to be necessary, and that guess is accurate. However, it turns out that we need to treat the cases of electorates with odd and even numbers of voters separately: using the technique explained in the previous chapter, we give proofs that require 15 voters for the odd case and 24 voters for the even case. These constant bounds hold for any number $m \geqslant 4$ of alternatives. We are able to show that these results are tight: for the case of precisely 4 alternatives,[4] there exist Condorcet extensions satisfying half-way monotonicity for up to 13 voters and 22 voters, respectively. This yields an alternating pattern of possibility and impossibility, shown in Table 2.1.

In the previous chapter we saw that, for 4 alternatives, Moulin's impossibility requires 12 voters to go through, while there exists a Condorcet extension satisfying participation for up to 11 voters. This gives us a rough but intriguing way to compare the relative strengths of participation and half-way monotonicity (see Table 2.1); we can see that half-way monotonicity is weaker than participation, but not by much.

2.2. Half-way Monotonicity and Participation

We start by defining half-way monotonicity. If $\succ \in A!$ is a linear order, then the *reverse* linear order $\succ^{\text{rev}}$ is defined by $a \succ^{\text{rev}} b \iff b \succ a$ for all $a, b \in A$. Given a profile $P \in A!^N$, we write $(P_{-i}, \succ_i') := P|_{N\setminus\{i\}} \cup \{(i, \succ_i')\}$ for the profile obtained from P by replacing i's vote by $\succ_i'$.

Definition 2.1. A fixed-electorate voting rule f satisfies *half-way monotonicity* if

$$f(P_{-i}, \succ_i) \succcurlyeq_i f(P_{-i}, \succ_i^{\text{rev}}) \quad \text{for all profiles } P \in A!^N \text{ and all voters } i \in N.$$

Thus, voters weakly prefer voting truthfully to voting the reverse of their preferences. If a rule violates half-way monotonicity, we say that it suffers from the *preference reversal paradox.*

Let us compare half-way monotonicity to the axiom of the previous chapter, participation. It turns out that participation is a stronger requirement than half-way monotonicity, in the sense that if a variable-electorate rule f satisfies participation, then the induced fixed-electorate rule on N satisfies half-way monotonicity. This was shown by Sanver and Zwicker [2009, Theorem

[3]The large number arises because the proof uses several copies of the full profile containing a copy of each of the $m!$ preference orders. Fixing the arithmetical error described above tends to necessitate using many more voters than this (Zwicker, private communication).

[4]For 3 alternatives, it is known that the maximin rule with some fixed tie-breaking is a Condorcet extension satisfying half-way monotonicity.

4.1] using a proof that established several interrelated implications among their monotonicity axioms. Here, we give a direct proof of this implication. The idea behind the proof is simple: a reversal of a vote is equivalent to a voter leaving the election and a new voter with reversed preferences joining it. Applying participation to both electorate changes, we find that the vote reversal cannot make the voter better off.

Lemma 2.2 (Sanver and Zwicker, 2009). *If a variable-electorate voting rule f satisfies participation, then f satisfies half-way monotonicity.*

Proof. Let $P \in A!^N$ be a profile and let $i \in N$ be a voter with preferences $\succcurlyeq_i$ in P. Consider the profile $P - i$ with i removed. By participation, we have $f(P) \succcurlyeq_i f(P - i)$. Also by participation, we have $f(P_{-i}, \succ_i^{\text{rev}}) \succcurlyeq_i^{\text{rev}} f(P - i)$. Putting these together, and using the definition of the reverse of an order, we have

$$f(P) \succcurlyeq_i f(P - i) \succcurlyeq_i f(P_{-i}, \succ_i^{\text{rev}}).$$

Thus, using transitivity, we have verified half-way monotonicity. □

Interestingly, to deduce half-way monotonicity for electorates of n voters, we only require participation to hold between electorates of size $n - 1$ and n. Núñez and Sanver [2017] also prove the implication of Lemma 2.2 by proposing an intermediate "Condition λ" that is implied by participation and that implies half-way monotonicity.

2.3. Results

We obtain our results in this setting using basically the same method discussed in the previous chapter, starting by considering pairwise rules only. Using SAT solvers and a similar encoding, we can establish that for specific n and m, a (pairwise) Condorcet extension satisfying half-way monotonicity exists. An important difference to the situation of the previous chapter concerns the generalisation of such a result to other values of n and m. Due to the variable-electorate nature of the participation axiom, it is clear that if there is no appropriate rule for 12 voters, then there also cannot be one for 13, 14, 15, ... voters; conversely, a good rule for 11 voters immediately induces a rule for fewer voters. However, the analogue for half-way monotonicity is not as clear, because we now operate in a fixed-electorate setting.

The following lemma functions as an induction step, and allows us to conclude that positive results also hold for smaller n and negative results hold for larger n, *as long as parity is preserved.* This caveat is important, and we will see that half-way monotonicity is less restrictive on Condorcet extensions defined for even electorates.

Lemma 2.3 (Induction Step). *Fix a number m of alternatives, and let $n \geqslant 1$. If there exists a Condorcet extension defined on electorates with $n+2$ voters which satisfies half-way monotonicity, then there also exists a Condorcet extension for n voters satisfying half-way monotonicity.*

Proof. Fix some linear order $\succ_*$ over A. Suppose $|N| = n$, and suppose f_{n+2} is a Condorcet extension satisfying half-way monotonicity, defined for the electorate $N \cup \{i, j\}$. Then define the voting rule f_n on the electorate N by

$$f_n(P) := f_{n+2}(P + (i, \succ_*) + (j, \succ_*^{\text{rev}})) \text{ for all profiles } P \in A!^N.$$

Then the voting rule f_n is Condorcet-consistent: if a profile $P \in A!^N$ admits a Condorcet winner, then this alternative remains the Condorcet winner after adding two completely opposed orders to P, since this operation does not change the majority margins. Further, f_n satisfies half-way monotonicity: let $P \in A!^N$ be a profile, in which $k \in N$ has preferences $\succ_k$. Then, because f_{n+2} satisfies half-way monotonicity.

$$\begin{aligned} f_n(P_{-k}, \succ_k^{\text{rev}}) = \;& f_{n+2}(P_{-k} + (k, \succ_k^{\text{rev}}) + (i, \succ_*) + (j, \succ_*^{\text{rev}})) \\ \not\succ_k \;& f_{n+2}(P_{-k} + (k, \succ_k) + (i, \succ_*) + (j, \succ_*^{\text{rev}})) = f_n(P_{-k}, \succ_k), \end{aligned}$$

and thus also f_n satisfies half-way monotonicity. □

Contrapositively, this lemma implies that an incompatibility between Condorcet-consistency and half-way monotonicity for n voters also applies to $n + 2k$ voters, for each $k \geqslant 0$. Thus, in our impossibilities below, we only need to handle the base case for $n = 15$ and $n = 24$, respectively.

Before we present the proofs, let us have a look at our positive result.

Proposition 2.4 (Possibilities). *For $m = 4$ alternatives, and for either $n = 13$ or $n = 22$ voters, there exists a Condorcet extension satisfying half-way monotonicity.*

Again, these voting rules are only available as look-up tables. Both of the voting rules mentioned are pairwise, so only depend on the weighted tournament induced by the input profile.

Our negative results are proved with arguments of a similar style as before, see the proof diagrams in Figures 2.1 and 2.2. In the figures, an arc from P to P' labelled "*reverse 2 dcba*" is interpreted as "P' is obtained from P by reversing the preferences of 2 voters with preferences $dcba$ in P. Now, if the voting rule chooses a or b at P, then the rule must also choose a or b at P' by half-way monotonicity". The profiles at the leaves all admit a Condorcet winner, which leads to a contradiction. The general proof strategy of our impossibility proofs is as follows: we identify an initial profile P_0, and iterate through each possible value of $f(P_0) \in A$. Assuming that $f(P_0) = x$, say, will then, by half-way monotonicity, imply restrictions on the possible values that f can take at profiles obtained from P_0 by reversing some of the votes. In particular, it will imply that f must not pick the Condorcet winner at some of these profiles, contradicting f being a Condorcet extension.

In the previous chapter, working with participation, we proved an induction step that allowed us to lift an impossibility for four alternatives to apply to any $m \geqslant 4$. Such an induction step is harder to prove here, and so we handle the general case explicitly in the following proofs.

As we noted, we will treat the cases of odd and even electorates separately, since the induction step of Lemma 2.3 only works in steps of two. Let us start with the odd case.

Theorem 2.5 (Odd Electorates). *For $m \geqslant 4$ alternatives and odd $n \geqslant 15$, there does not exist a Condorcet extension satisfying half-way monotonicity.*

Proof By Lemma 2.3, we only need to handle the case with $n = 15$. Write $A = \{a, b, c, d\} \cup X$, where $X = \{x_1, \dots, x_{m-4}\}$. Suppose there exists a half-way monotonic Condorcet extension f for 15 voters. Consider the 15-voter profile P_0 depicted on the right. As usual, the column numbers indicate how many voters submit a given ordering. The X at the bottom of each vote should be replaced by an arbitrary ordering of the alternatives in X. Our proof is by case analysis on the value of $f(P_0)$, arriving at a contradiction in each case.

1	3	3	4	2	2
a	a	b	c	d	d
b	b	d	a	c	c
c	d	c	b	a	b
d	c	a	d	b	a
X	X	X	X	X	X

Suppose first that $f(P_0) \in \{a, b\} \cup X$. Let P_1 be the profile after one $dcba$ voter reverses their preferences in P_0. By half-way monotonicity, we have $f(P_1) \in \{a, b\} \cup X$. Suppose that $f(P_1) \in \{a\} \cup X$. Let P_2 be the profile after two $bdca$ voters reverse their preferences in P_1. By half-way monotonicity, we have $f(P_2) \in \{a\} \cup X$; however c is the Condorcet winner in P_2, contradicting Condorcet-consistency of f. Thus $f(P_1) = b$. Let P_3 be the profile obtained from P_1 after one $dcab$ voter and two $cabd$ voters reverse their preferences. By half-way monotonicity, we have $f(P_3) \in \{b, d\}$. However, a is the Condorcet winner in P_3, a contradiction.

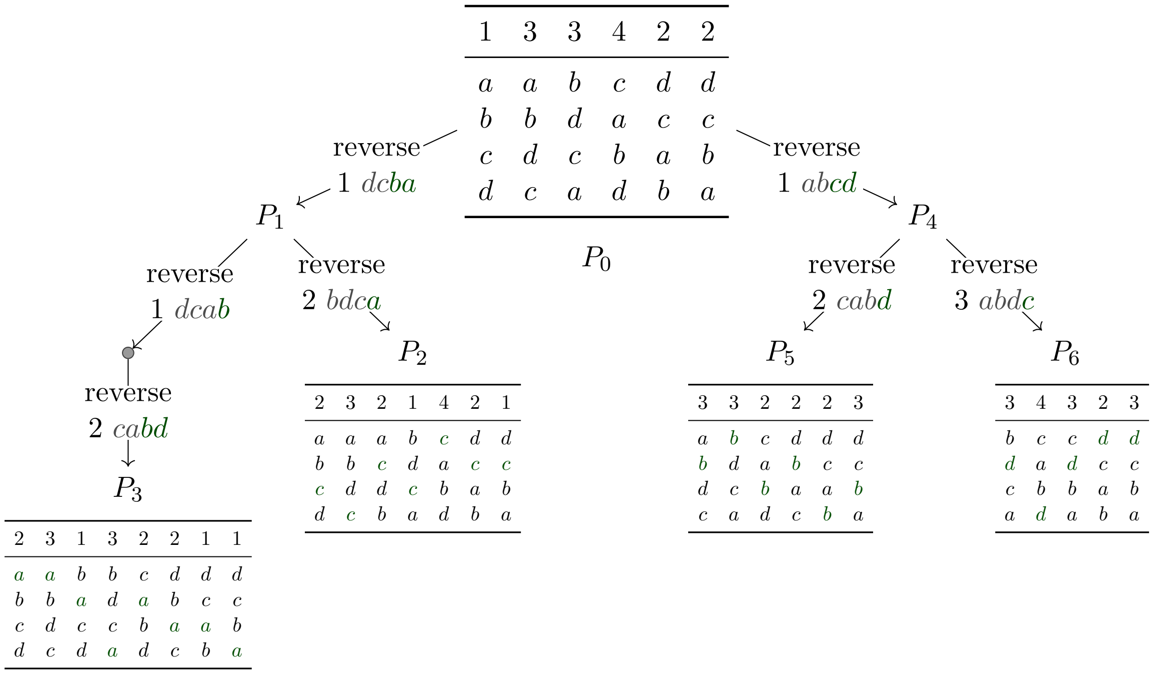


Figure 2.1.: Proof diagram of the proof of Theorem 2.5.

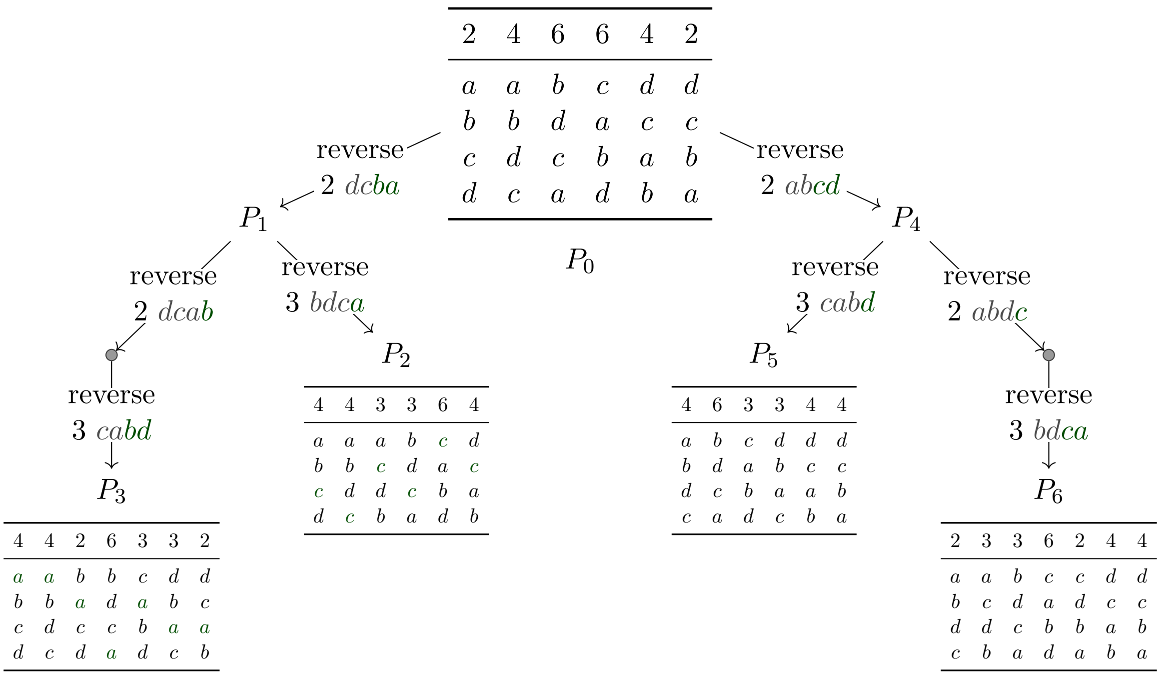


Figure 2.2.: Proof diagram of the proof of Theorem 2.6.

Thus $f(P_0) \in \{c, d\}$. Let P_4 be the profile obtained from P_0 by reversing an *abcd* voter. By half-way monotonicity, $f(P_4) \in \{c, d\}$. Suppose $f(P_4) = d$. Let P_5 be the profile obtained from P_4 by reversing two *cabd* voters; then $f(P_5) = d$. But b is the Condorcet winner at P_5, a contradiction. Hence $f(P_4) = c$. Let P_6 be the profile obtained from P_4 by reversing three *abdc* voters; then $f(P_6) = c$. But d is the Condorcet winner at P_6, a contradiction. □

While constructing these proofs, our SAT-based search was aided by only considering profiles made up of the about 6–10 preference orders that appear in the proofs for the no-show paradox.

The bound on n for even electorates is significantly higher than for odd ones. Intuitively, the reason is that Condorcet-consistency is less demanding in even electorates, because Condorcet winners need to beat every other alternative by a majority margin of at least 2.

Theorem 2.6 (Even Electorates)**.** *For $m \geqslant 4$ alternatives and even $n \geqslant 24$, there does not exist a Condorcet extension satisfying half-way monotonicity.*

Proof By Lemma 2.3, we only need to handle the case with $n = 24$. Write $A = \{a, b, c, d\} \cup X$, where $X = \{x_1, \dots, x_{m-4}\}$. Suppose there exists a half-way monotonic Condorcet extension f for 24 voters. Consider the 24-voter profile P_0 depicted on the right. The column numbers indicate how many voters submit a given ordering; for example, there are exactly 4 voters in P_0 with the ordering $a \succ b \succ d \succ c \succ X$. The X at the bottom should be replaced by an arbitrary ordering of the alternatives in X. Our proof is by case analysis on the value of $f(P_0)$, arriving at a contradiction in each case.

2	4	6	6	4	2
a	a	b	c	d	d
b	b	d	a	c	c
c	d	c	b	a	b
d	c	a	d	b	a
X	X	X	X	X	X

Suppose first that $f(P_0) \in \{a, b\} \cup X$. Let P_1 be the profile after two *dcba* voters reverses their preferences in P_0. By half-way monotonicity, we have $f(P_1) \in \{a, b\} \cup X$. Suppose that $f(P_1) \in \{a\} \cup X$. Let P_2 be the profile after three *bdca* voters reverse their preferences in P_1. By half-way monotonicity, we have $f(P_2) \in \{a\} \cup X$; however c is the Condorcet winner in P_2, contradicting Condorcet-consistency of f. Thus $f(P_1) = b$. Let P_3 be the profile obtained from P_1 after two *dcab* voter and three *cabd* voters reverse their preferences. By half-way monotonicity, we have $f(P_3) \in \{b, d\}$. However, a is the Condorcet winner in P_3, a contradiction.

Thus $f(P_0) \in \{c, d\}$. Let P_4 be the profile obtained from P_0 by reversing two *abcd* voters. By half-way monotonicity, $f(P_4) \in \{c, d\}$. Suppose $f(P_4) = d$. Let P_5 be the profile obtained from P_4 by reversing three *cabd* voters; then $f(P_5) = d$. But b is the Condorcet winner at P_5, a contradiction. Hence $f(P_4) = c$. Let P_6 be the profile obtained from P_4 by reversing two *abdc* and three *bdca* voters; then $f(P_6) = c$. But d is the Condorcet winner at P_6, a contradiction. □

One may wonder whether it is a coincidence that our cut-off for half-way monotonicity in even electorates ($n = 24$) is double the cut-off for participation ($n = 12$). The answer is no, as suggested by the proof of Theorem 4.1(3) of Sanver and Zwicker [2009], which (roughly) shows that half-way monotonicity for $2n$ voters implies participation for n voters, at least in the presence of homogeneity and reversal cancellation. In fact, we have obtained the proof of Theorem 2.6 by taking the proof of Theorem 1.5, and doubling all the profiles involved in the proof.

2.4. Conclusions

We have seen that the impossibility of the previous chapter holds even for the weaker axiom of half-way monotonicity. In the conference version [Peters, 2017a], we study analogues of our results for set-valued voting rules [see also Sanver and Zwicker, 2012], and also consider a weakened version of half-way monotonicity.

Thanks to Felix Brandt, Christian Geist, and Bill Zwicker for discussions that led to this work, and to the anonymous reviewers of TARK-17 for excellent suggestions.

3. A Disjunctive Gibbard–Satterthwaite Theorem

We combine the result of the previous chapter and a strengthening of a result due to Campbell and Kelly to prove a "disjunctive Gibbard–Satterthwaite theorem". This theorem establishes the familiar fact that all sensible voting rules are manipulable, but makes this conclusion more precise by illuminating some types of manipulations that are unavoidable. Polemically, it shows that every voting rule is either "needlessly" or "egregiously" manipulable. A result of this form was first proposed by William S. Zwicker.

3.1. Introduction

In the previous two chapters, we saw two theorems that can be seen as criticisms of Condorcet's principle to elect Condorcet winners when they exist. We saw that any rule following this principle will suffer from the no-show and the preference reversal paradox. However, there are also important advantages to electing Condorcet winners. Notably, Condorcet extensions are resistant to certain types of strategic manipulations. Formally, we say that a fixed-electorate voting rule f is *manipulable* if for some profile P, there is a voter i who can manipulate by reporting $\succ_i'$ instead of their truthful preferences $\succ_i$, such that $f(P_{-i}, \succ_i') \succ_i f(P_{-i}, \succ_i)$. Thus, voter i strictly prefers misrepresenting their preferences. The famous Gibbard–Satterthwaite theorem states that all sensible voting rules are manipulable:

Theorem 3.1 (Gibbard, 1973, Satterthwaite, 1975)**.** *Suppose that $n \geqslant 2$ and $m \geqslant 3$. Let f be a fixed-electorate voting rule $f : A!^N \to A$ which is onto, and which is not manipulable. Then there is some $i \in N$ such that $f(P) = \text{top}(\succ_i)$ for all $P \in A!^N$. Thus, f is a dictatorship.*

On the other hand, the following well-known observation says that Condorcet extensions cannot be manipulated when we only look at profiles admitting a Condorcet winner:

Proposition 3.2. *Let P be a profile, and suppose $P' = (P_{-i}, \succ_i')$ is a profile obtained from P after voter i misreports their preferences. If both P and P' admit a Condorcet winner, and f is a Condorcet extension, then $f(P') \not\succ_i f(P)$, so the manipulation was not successful.*

Proof. For a contradiction, suppose that $f(P') = b \succ_i a = f(P)$. Since a is the Condorcet winner at P, there is a strict majority $N' \subseteq N$ of voters who prefer a to b in P. Since $b \succ_i a$, we have $i \notin N'$. Hence, all voters in N' also prefer a to b in P', forming a strict majority. This contradicts that b is the Condorcet winner at P'. □

Surprisingly, a kind of converse to Proposition 3.2 holds: if f is *not* a Condorcet extension, then it is possible for a voter to manipulate between two profiles which both admit a Condorcet winner. For example, the Borda rule is not Condorcet-consistent, and admits the following manipulation:

v_1	v_2	v_3
a	a	b
b	b	a
c	c	c
d	d	d

P

$\longrightarrow$

v_1	v_2	v_3
a	a	b
b	b	c
c	c	d
d	d	a

P'

In both P and P', alternative a is the Condorcet winner because a majority of voters ranks a in top position. Borda selects a at profile P, but selects alternative b at profile P'. This gives voter v_3 a successful manipulation. I shall say that a voting rule which can be manipulated among profiles with Condorcet winners is *needlessly manipulable*, because the manipulation could be avoided if the voting rule would pick the Condorcet winners at those profiles.

The converse alluded to above was proved by Campbell and Kelly. To state their theorem, let us introduce some definitions. We consider a fixed-electorate setting, where N is the set of voters. Let $\mathcal{D} \subseteq A!^N$ be a *domain*, i.e., a subcollection of profiles. A voting rule *on the domain* $\mathcal{D}$ is a map $f : \mathcal{D} \to A$. We say that

- f is *onto* if for all $a \in A$, there is $P \in \mathcal{D}$ with $f(P) = a$;
- f is *Pareto* if whenever $P \in \mathcal{D}$ is a profile such that $a \succ_i b$ for all $i \in P$, we have that $f(P) \neq b$;
- f is *non-dictatorial* if there is no $i \in N$ such that $f(P) = \text{top}(\succ_i)$ for all profiles $P \in \mathcal{D}$;
- f is *anonymous* if $f(\sigma P) = f(P)$ for all permutations σ of N such that $\sigma P \in \mathcal{D}$;
- f is *manipulable* if there exists a profile P, a voter i, and a linear order $\succ_i'$ such that both $(P_{-i}, \succ_i') \in \mathcal{D}$ and $(P_{-i}, \succ_i) \in \mathcal{D}$, and $f(P_{-i}, \succ_i') \succ_i f(P_{-i}, \succ_i)$. Thus, voter i strictly prefers misrepresenting their preferences.

Note that Pareto implies onto, and that anonymity is stronger than non-dictatorship.

Let $\mathcal{D}_{\text{Condorcet}}$ denote the domain of all profiles $P \in A!^N$ that admit a Condorcet winner, and let $f_{\text{Condorcet}} : \mathcal{D}_{\text{Condorcet}} \to A$ be the *Condorcet rule* which assigns to each profile its Condorcet winner. We can now state the theorem.

Theorem 3.3 (Campbell and Kelly, 2003, 2016)**.** *Suppose N contains an odd number of voters and $|A| \geqslant 3$. Let $f : \mathcal{D}_{Condorcet} \to A$ be a non-dictatorial and onto voting rule. Then f is not manipulable if and only if $f = f_{\text{Condorcet}}$.*

In other words, among voting rules defined on the domain $\mathcal{D}_{\text{Condorcet}}$, the Condorcet rule is characterised as the unique rule that is non-dictatorial, onto, and strategyproof. We will be interested in the implications of the Campbell–Kelly Theorem for voting rules $f : A!^N \to A$ defined for *all* profiles, not just for $\mathcal{D}_{\text{Condorcet}}$. It is tempting to make an argument like the following:

> *(incorrect)* Suppose $f : A!^N \to A$ is a voting rule that is non-dictatorial and onto, but that is not a Condorcet extension. Then, $f|_{\mathcal{D}_{\text{Condorcet}}}$ is a non-dictatorial and onto voting rule defined on $\mathcal{D}_{\text{Condorcet}}$. Since it differs from the Condorcet rule, by Theorem 3.3, it is manipulable on $\mathcal{D}_{\text{Condorcet}}$. Thus, f is needlessly manipulable.

The argument fails because $f|_{\mathcal{D}_{\text{Condorcet}}}$ need not be non-dictatorial or onto. These properties are not preserved under restricting the domain of f.[1] However, the argument as above goes through if we replace non-dictatorship by the stronger condition of anonymity, and replace ontoness by Pareto.[2]

Corollary 3.4. *Suppose N contains an odd number of voters and $|A| \geqslant 3$. Let $f : A!^N \to A$ be a voting rule on the full domain. Suppose that f is anonymous and Pareto. If f is* not *Condorcet-consistent, then f is manipulable on $\mathcal{D}_{Condorcet}$.*

Proof. If f is anonymous, then $f|_{\mathcal{D}_{\text{Condorcet}}}$ is also anonymous and (because $\mathcal{D}_{\text{Condorcet}}$ is closed under permuting votes) thus non-dictatorial. Similarly, if f is Pareto, then $f|_{\mathcal{D}_{\text{Condorcet}}}$ is onto: let $x \in A$, and let P be a profile in which $\text{top}(\succ_i) = x$ for all voters $i \in N$. Then $P \in \mathcal{D}_{\text{Condorcet}}$ since x is a Condorcet winner. Since x Pareto-dominates every other alternative and f is Pareto, $f(P) = x$, and hence $f|_{\mathcal{D}_{\text{Condorcet}}}(P) = x$, which shows that $f|_{\mathcal{D}_{\text{Condorcet}}}$ is onto. Thus, by Theorem 3.3, $f|_{\mathcal{D}_{\text{Condorcet}}}$ is manipulable. □

Hence, for odd n, every anonymous and Pareto rule is needlessly manipulable.

3.2. The Campbell–Kelly Theorem for Even Numbers of Voters

An obvious gap of these results is that they require the number of voters to be odd. What about even numbers of voters? It turns out that Theorem 3.3 does not hold for even n: Merrill [2011] constructed examples of rules other than the Condorcet rule that satisfy the conditions of Theorem 3.3. We quickly discuss these examples here for completeness.

The reason that the theorem fails if n is even, is that if a (say) is the Condorcet winner at a profile $P \in \mathcal{D}_{\text{Condorcet}}$, then no voter can unilaterally change the Condorcet winner: every profile $P' \in \mathcal{D}_{\text{Condorcet}}$ that differs from P only in the preferences of a single voter will also have a as its Condorcet winner. This is because, for even n, a Condorcet winner beats every other candidate by a margin of at least 2. Hence, for example, the following rule is strategyproof but non-dictatorial if n is even:

$$f_{\dagger}(P) = \begin{cases} \text{top}(\succ_1) & \text{if } a \text{ is the Condorcet winner of } P, \\ \text{top}(\succ_2) & \text{otherwise.} \end{cases}$$

Here, $\text{top}(\succ_i)$ refers to the most-preferred alternative of voter i. Note that this rule is not anonymous: it depends on the identities of the voters. Merrill [2011] gives another example of a 'bad' rule that is strategyproof: take the alternatives to be the members of $\mathbb{Z}_m$ equipped with modular arithmetic, and consider the following rule:

$$f_{\dagger\dagger}(P) = f_{\text{Condorcet}}(P) + 1.$$

This rule is again strategyproof on $\mathcal{D}_{\text{Condorcet}}$, and it is anonymous, but it fails to be neutral or Pareto. (A voting rule is *neutral* if whenever we permute the alternative names in a profile (and the resulting profile still is in the domain), then the output of f is permuted by the same permutation.)

Campbell and Kelly recently proved the following version of their theorem for even n.

[1] For example, consider the voting rule that returns voter 1's top choice if there is a Condorcet winner, and voter 2's top choice otherwise. Or, consider the voting rule that outputs a fixed $x \in A$ if there is a Condorcet winner, and voter 2's top choice otherwise.

[2] The same argument works if we replace Pareto by the weaker notion of *unanimity*, which requires an alternative x to be selected if every voter ranks x in top position.

Theorem 3.5 (Campbell and Kelly, 2015)**.** *Let $n \geqslant 4$ be even, and let $m \geqslant 4$. Let $f : \mathcal{D}_{\text{Condorcet}} \to A$ be an* anonymous and neutral *voting rule. Then f is strategyproof if and only if $f = f_{\text{Condorcet}}$.*

Again it is difficult to understand the implications of this result for rules f defined on the full domain $A!^N$. A resolute voting rule defined on the full domain can usually not be both anonymous and neutral [Moulin, 1983], since the latter conditions force there to be tied outcomes.

In this section, we prove another analogue of the Campbell–Kelly theorem for even n, using anonymity and Pareto (instead of neutrality) as our background axioms. As usual, this theorem is obtained with the help of SAT solvers, though in this case the "human-produced" part is more significant. We present the proof, and afterwards indicate how SAT solvers were used.

The proof will consider many possible manipulations, where in some profile P, voter i changes from $\succ_i$ to $\succ_i'$, resulting in profile P'. Strategyproofness can then be used to conclude that $f(P) \succcurlyeq_i f(P')$, so that the manipulating voter is weakly worse off. In the proof, we will also repeatedly consider the possibility of voter i *reversing the change*, by which we mean the reverse manipulation from P' to P, where strategyproofness allows us to conclude that $f(P') \succcurlyeq_i' f(P)$.

Theorem 3.6. *Let $n \geqslant 4$ be even or odd, and let $m \geqslant 3$. Let $f : \mathcal{D}_{\text{Condorcet}} \to A$ be an* anonymous and Pareto *voting rule. Then f is strategyproof if and only if $f = f_{\text{Condorcet}}$.*

Proof. Write $A = \{a, b_1, \dots, b_{m-1}\}$. We consider the case for even n in detail, since it cannot be deduced from existing results, and indicate what needs to be changed for odd n at the end. So write $n = 2k$ for some $k \geqslant 2$, and set $k^+ := k + 1$ and $k^- := k - 1$. Thus, $n = k^+ + k^-$, and k^+ is the smallest number of voters required to obtain a strict majority.

Suppose for a contradiction that f is anonymous, Pareto, and strategyproof, but that $f \neq f_{\text{Condorcet}}$. Then, by relabeling alternatives, we may assume that there exists a profile P with $f_{\text{Condorcet}}(P) = a$ but $f(P) = b_1$.

Because $f_{\text{Condorcet}}(P) = a$, there are at least k^+ voters who prefer a to b_1. Partition $N = N_1 \cup N_2 \cup N_3$ into three sets, where $N_1 \cup N_2$ is the set of voters who prefer a to b_1 in P, N_3 is the set of voters who prefer b_1 to a in P, and $|N_1| = k^+$. The set N_2 may be empty in case that $|N_1 \cup N_2| = k^+$. Note that $|N_2 \cup N_3| = n - k^+ = k^-$.

Let P_* be the profile in which

$$\begin{aligned} &a \succ_i b_1 \succ_i b_2 \succ_i \cdots \succ_i b_{m-1} && \text{for } i \in N_1, \\ &b_1 \succ_i b_2 \succ_i \cdots \succ_i b_{m-1} \succ_i a && \text{for } i \in N_2 \cup N_3. \end{aligned}$$

We claim that $f(P_*) = b_1$.

We have assumed that $f(P) = b_1$. Suppose a voter $i \in N_1$ changes their vote in P to $ab_1 \dots b_{m-1}$. Let us call the resulting profile P'. If $f(P') = a$, this would be a successful manipulation (since by definition of N_1, i prefers a to b_1), contradicting strategyproofness. If $f(P') = b_j \neq b_1$, then voter i reversing the change would be a successful manipulation (since $f(P) = b_1$ and b_1 is preferred to b_j in $ab_1 \dots b_{m-1}$), a contradiction. Hence $f(P') = b_1$. Repeating the same argument while letting each voter in N_1 replace their vote by $ab_1 \dots b_{m-1}$, we find that $f(P_{***}) = b_1$, where P_{***} is the profile in which each voter from N_1 reports $ab_1 \dots b_{m-1}$ and all other voters report the same preferences as they report in P.

Next, we start from P_{***} and let each voter from N_2 replace their preferences by $b_1 \dots b_{m-1}a$. At each step, the winner selected by f must be b_1, because otherwise reversing the change would be a successful manipulation (because b_1 is the top choice in $b_1 \dots b_{m-1}a$). Thus, we find that $f(P_{**}) = b_1$, where in P_{**} each voter in N_1 reports $ab_1 \dots b_{m-1}$, each voter in N_2 reports $b_1 \dots b_{m-1}a$, and each voter in N_3 reports the same as in P.

Finally, we start from P_{**} and let each voter from N_3 replace their preferences by $b_1 \dots b_{m-1}a$. At each step, the winner selected by f must be b_1 (because otherwise reversing the change would be a successful manipulation since b_1 is the top choice in $b_1 \dots b_{m-1}a$). Thus, we find that $f(P_*) = b_1$, proving the claim.

We will now consider several other profiles shown in Figure 3.1. For notational convenience, we write $c := b_{m-1}$, and write "$\cdots$" for the ranking $b_2 \succ \cdots \succ b_{m-2}$. The profiles only specify how many voters submit each ranking; since we have assumed that f is anonymous, this is sufficient to determine f's output at each profile.

k^+	k^-
a	b_1
b_1	$\vdots$
$\vdots$	c
c	a

(a) P_*

k^+	k^-
a	b_1
b_1	$\vdots$
$\vdots$	a
c	c

(b) P_2

k^+	k^-
a	b_1
c	$\vdots$
b_1	a
$\vdots$	c

(c) P_3

k^+	k^-
a	c
c	b_1
b_1	$\vdots$
$\vdots$	a

(d) P_4

2	k^-	k^-
a	b_1	c
c	$\vdots$	a
b_1	a	b_1
$\vdots$	c	$\vdots$

(e) P_5

2	k^-	k^-
a	a	c
c	b_1	a
b_1	$\vdots$	b_1
$\vdots$	c	$\vdots$

(f) P_6

k^+	k^-
a	c
c	a
b_1	b_1
$\vdots$	$\vdots$

(g) P_7

Figure 3.1.: Profiles used in the second part of the proof of Theorem 3.6.

The claim has established that $f(P_*) = b_1$. Now, consider profile P_2. By Pareto, $f(P_2) \in \{a, b_1\}$. Suppose $f(P_2) = a$. Then, let each of the k^- voters successively replace their vote by $b_1 \dots ca$. At each step, by Pareto, either a or b_1 is selected, and by strategyproofness in fact a is selected. At the end of process we have reached P_*, so $f(P_*) = a$, contradiction. Thus, $f(P_2) = b_1$. Now let each of the k^+ voters of P_2 replace their vote by $acb_1 \dots$; by Pareto at each step either a or b_1 is selected, and by strategyproofness in fact b_1 is selected. Hence $f(P_3) = b_1$.

Next, consider profile P_7. By Pareto, $f(P_7) \in \{a, c\}$. Suppose first that $f(P_7) = a$. Let each of the k^- voters replace their votes by $cb_1 \dots a$. At each step, by Pareto, either a or c is selected, and by strategyproofness in fact a is selected, and thus $f(P_4) = a$. Next, let each of the k^- voters in P_4 replace their vote by $b_1 \dots ac$; by strategyproofness, at each step a is selected. Hence $f(P_3) = a$, a contradiction. Thus, $f(P_7) = c$. Next, let k^- of the k^+ voters (i.e., all but two) in P_7 replace their vote by $ab_1 \dots c$. By Pareto, at each step a or c is selected, and by strategyproofness in fact c is selected, so $f(P_6) = c$. Next, let the k^- voters reporting $ab_1 \dots c$ in P_6 change their vote to $b_1 \dots ac$; by strategyproofness, c is still selected, so $f(P_5) = c$.

However, we have established $f(P_3) = b_1$. Suppose that k^- of the k^+ voters in P_3 change their vote from $acb_1 \dots$ to $cab_1 \dots$. At each step, we must still select b_1 by Pareto and strategyproofness. The resulting profile is the same as P_5 after permuting voters, so by anonymity $f(P_5) = b_1$, a contradiction to $f(P_5) = c$.

In case that n is odd, write $n = 2k + 1$, and set $k^+ := k + 1$ and $k^- := k$. The proof goes through unchanged, except that in P_5 and P_6 as shown in Figure 3.1, only 1 instead of 2 voters report the ranking in the left-most column (since now $k^+ - k^- = 1$ rather than 2). □

The theorem implies that if an anonymous and Pareto voting rule is not a Condorcet extension, then it is needlessly manipulable.

The first part of the proof (establishing the 'claim') shows that in order to prove $f = f_{\text{Condorcet}}$, it is sufficient to prove that $f(P_*) = f_{\text{Condorcet}}(P_*)$ for a particular profile P_*. This part of the proof is similar to arguments appearing in the characterisations of Campbell and Kelly [2003, 2015]. Once the reduction to P_* is in place, we can formulate our problem in propositional logic in a similar way to previous chapters. In this way, we obtain a formula whose models correspond to voting rules on $\mathcal{D}_{\text{Condorcet}}$ that are anonymous, Pareto, and strategyproof. This formula is satisfiable (namely by $f_{\text{Condorcet}}$), but once we add a clause specifying $f(P_*) \neq f_{\text{Condorcet}}(P_*)$, we obtain unsatisfiability. Extracting and analysing an MUS yields the shape of the second part of the proof (operating on profiles P_* and P_2 to P_7). The SAT approach only works for fixed n and m; we generalised the argument by hand by inspecting the MUS proofs for different n.

One might ask why it is necessary to first prove the claim about P_*, before using a SAT solver. After all, the condition $f \neq f_{\text{Condorcet}}$ can be encoded easily (as a single clause $\bigvee_P f(P) \neq f_{\text{Condorcet}}(P)$), and gives an unsatisfiability result. The problem with this attempt is that the constraint $f \neq f_{\text{Condorcet}}$ is a global one, and its presence forces any MUS to include clauses about every profile in $\mathcal{D}_{\text{Condorcet}}$. This does not yield an interpretable, compact proof. In contrast, the condition $f(P_*) \neq f_{\text{Condorcet}}(P_*)$ is local, and allows for small MUSes.

3.3. A Dilemma Theorem

Let us now combine some of the theorems we have obtained into a single statement. In the previous section, we proved that every voting rule (except Condorcet extensions) satisfying anonymity and Pareto is manipulable in a specific way (namely between two profiles which both admit a Condorcet winner). In the previous chapter, we saw that every Condorcet extension is also manipulable, namely by reversing one's preferences. Because any voting rule either is or is not a Condorcet extension, we obtain the following "disjunctive Gibbbard–Satterthwaite theorem".

Theorem 3.7. *Suppose there are at least 4 alternatives and at least 24 voters. Let f be a voting rule satisfying anonymity and Pareto. Either f is manipulable on $\mathcal{D}_{Condorcet}$, or f is manipulable by preference reversal.*

Proof. If f is a Condorcet extension then this follows from Theorems 2.5 and 2.6; otherwise it follows from Theorem 3.6. □

Comparing this result with the seminal result of Gibbard [1973] and Satterthwaite [1975], we see that the disjunctive version makes stronger assumptions (larger minimum values for n and m, anonymity instead of non-dictatorship, and Pareto instead of onto), but it also has a stronger conclusion: It more explicitly identifies the nature of manipulability: they are either *needless* (because they could be avoided by selecting Condorcet winners) or *egregious* (because there is a successful manipulation on which every pairwise comparison is misreported).

The disjunctive version of Gibbard–Satterthwaite we have obtained was first proposed by Zwicker [2016, Corollary 2.8] using slightly different assumptions. One could imagine other results of this type showing that every voting rule is manipulable using misreports of certain types. One such result appears in the literature: the Gibbard–Satterthwaite Theorem holds even if we allow voters to report only preferences that are obtained by at most one (adjacent) swap from the truthful vote [Sato, 2013, Caragiannis et al., 2012].

Part II.

Budgeting with Divisible Projects

4. Aggregating Budget Proposals

We consider a participatory budgeting problem in which each voter submits a proposal for how to divide a single divisible resource (such as money or time) among several possible projects and these proposals must be aggregated. We assume ℓ_1 preferences, for which a voter's disutility is given by the ℓ_1 distance between the aggregate division and the voter's proposed division. In this model, the social-welfare-maximising mechanism, which minimises the average ℓ_1 distance between the outcome and each voter's proposal, is strategyproof [Goel et al., 2019]. However, it fails to satisfy a natural fairness notion of proportionality, placing too much weight on majority preferences. Leveraging a connection between market prices and the generalised median rules of Moulin [1980], we introduce the *independent markets* mechanism, which is both strategyproof and proportional. We unify the social-welfare-maximising mechanism and the independent markets mechanism by defining a broad class of *moving phantom* mechanisms that includes both. We show that every moving phantom mechanism is strategyproof. Finally, we characterise the social-welfare-maximising mechanism as the unique Pareto-optimal mechanism in this class, suggesting an inherent tradeoff between Pareto-optimality and proportionality.

4.1. Introduction

Throughout Part II, we consider the problem of dividing a continuous budget among several possible uses, where each use can receive any fraction of the budget: the budget is perfectly divisible. Examples might include an organisation splitting its monetary budget among departments or a team of event organisers deciding what fraction of the event length to devote to various activities. Another example would be a government that needs to decide on a target energy mix (that is, how much energy should come from fossil fuels, nuclear, or renewable sources) and wishes to aggregate expert opinions. In this chapter, we will study preference aggregation rules that ask each voter for the voter's most-preferred division of the budget. Thus, each voter proposes one possible way of dividing the budget, and we have to combine these proposals into an aggregate division.

A first idea for aggregating the proposals would be to take the *mean*: view the proposals as vectors of fractions summing to 1, and then calculate the average. An advantage of the mean is that, intuitively, it is very fair: in effect, the mean splits the budget into pieces of size $1/n$, and lets each person decide what to do with their piece. However, the mean has a serious flaw: it is easily manipulated. Suppose that four voters propose dividing the budget (50%, 50%) across two projects, and a fifth voter prefers a (60%, 40%) split. The mean of these proposals is (52%, 48%). If the fifth voter instead pretends to prefer a (100%, 0%) split, the resulting mean is (60%, 40%), and so the voter has obtained the actually preferred outcome.

In this chapter, we seek mechanisms that are resistant to such manipulation. In particular, we require that no voter can, by lying, move the aggregate division toward the voter's preference on one project without moving it away from it by an equal or greater amount on other projects. In other words, we seek budget aggregation mechanisms that are strategyproof under ℓ_1 preferences, with each voter's disutility for a budget division equal to the ℓ_1 distance between that division and the division the voter prefers most.

Goel et al. [2019] showed that choosing an aggregate budget division that maximises the utilitarian welfare of the voters (that is, a division that minimises the total ℓ_1 distance from each voter's report) is both strategyproof and Pareto-optimal under this utility model. However, this utilitarian rule has a tendency to overweight majority preferences. For example, imagine that sixty voters propose (100%, 0%) while forty propose (0%, 100%). The utilitarian aggregate is (100%, 0%) while the mean is (60%, 40%). In many scenarios, the latter solution is more in the spirit of consensus. For example, suppose a city uses participatory budgeting, and imagine that each family votes for all education dollars to go to their own neighbourhood school. The utilitarian aggregate would earmark the entire budget to the most populous school district, while we may prefer that funds are split in proportion to the districts' populations. To capture this idea of fairness, we define a notion of *proportionality*, requiring that when voters are single-minded (as in this example), the fraction of the budget assigned to each project is equal to the proportion of voters who favour that project. Do there exist aggregators that are both strategyproof and proportional?

For the case of two projects, ℓ_1 preferences are a special case of *single-peaked* preferences, well-studied in the voting literature. The seminal results of Moulin [1980] imply that, in this setting, all strategyproof voting schemes correspond to inserting $n + 1$ "phantom" proposals, where n is the number of voters, and returning the median of the n true proposals and the $n + 1$ phantoms. We show that there exists a way of placing the phantoms that results in a proportional mechanism for two projects.

Generalising Moulin's phantom median mechanisms to allow for more than two projects is difficult. Existing proposals for multidimensional generalisations take a median in each dimension independently [Border and Jordan, 1983, Peters et al., 1992, Barberà and Jackson, 1994]. Unfortunately, this strategy fails in our application which includes a normalisation constraints (total aggregate spending must sum to 1). Unlike the mean, taking a coordinate-wise median will usually fail to normalise. However, we find a way to extend Moulin's idea of phantom voters to our higher-dimensional setting, by allowing the set of phantoms to continuously shift upwards or downwards, thereby increasing or decreasing the sum of the median until the aggregate becomes normalised. This idea allows us to define a very general class of *moving phantom* mechanisms. Although one might think that allowing the final phantom locations to depend on voters' reports might give voters an incentive to misreport, we prove that every moving phantom mechanism is strategyproof under ℓ_1 preferences.

Among this large family of strategyproof mechanisms, we find one that satisfies our proportionality requirement. This moving phantom mechanism is obtained when phantoms are placed uniformly between 0 and a value $x \geqslant 0$ which increases until the coordinate-wise medians sum to 1. To analyse this mechanism, we prove that the aggregate found by this mechanism can be interpreted as the clearing prices in a market system, and hence call it the *independent markets* mechanism. This reveals an unexpected connection between market prices and generalised medians that may be of broader interest.

In contrast, the independent markets mechanism unfortunately fails to satisfy Pareto-optimality. We show that this is unavoidable, as no proportional moving phantom mechanism is Pareto-optimal. In fact, we prove that there is a *unique* Pareto-optimal moving phantom mechanism. In this mechanism, all phantoms start at 0 and then, one by one, transition to 1, with no two phantoms moving at the same time. This mechanism turns out to also have a phantom-free interpretation: it is equivalent to selecting the maximum-entropy budget division out of all those that maximise social welfare, the same mechanism studied by Goel et al. [2019] up to the choice of tie-breaking rules.

At the end of the chapter, we consider a slight tweak to our model, where projects come with a minimum level of funding: they should either receive no funding, or at least the minimum amount, because the projects do not make sense with less funding. We show that adding this complication can make it impossible to achieve strategyproofness.

Related Work Several recent papers study voting rules for participatory budgeting, considering both axiomatics and computational complexity, but under the assumption that indivisible projects can either be fully funded or not funded at all [Goel et al., 2019, Benade et al., 2017, Lu and Boutilier, 2011, Aziz et al., 2018a]. The setting in which partial funding of projects is permitted has also been studied, but generally under a different utility model in which voters assign utility scores to the projects rather than having an ideal distribution [Fain et al., 2016, Bogomolnaia et al., 2005, Aziz et al., 2019a]. We will discuss this setting in Chapters 5 and 6.

The work of Goel et al. [2019] is closely related to this chapter. The primary focus of their paper is on *knapsack voting*, in which each voter submits a preferred set of indivisible projects to fully fund. However, they also consider the use of *fractional* knapsack voting in a setting in which partial funding of projects is permitted and voters have ℓ_1 preferences. This corresponds exactly to our setting. They show that the mechanism that maximises social welfare (with some fixed tie-breaking) is strategyproof. We replicate this result by showing that the welfare-maximising mechanism (with an arguably more natural way to break ties) is a member of the large class of moving phantom mechanisms, all of which are strategyproof. Goel et al. do not consider other mechanisms for the fractional case.

The strategyproof aggregation of preferences over numerical values (such as the temperature for an office) has been extensively studied. A famous result of Moulin [1980] characterises the set of strategyproof voting rules under the assumption that voters have single-peaked preferences over values in $[0, 1]$. These voting rules are generalised median schemes. The best-known example is the standard median, in which each voter reports an ideal point in $[0, 1]$ and the median report is selected. Other voting rules in this class insert "phantom voters" who report a fixed top choice. Barberà et al. [1993] obtained a multi-dimensional analogue of this result for $[0, 1]^m$, and there are further generalisations that characterise strategyproof rules if other constraints are imposed on the feasible set [Barberà et al., 1997]. Crucially, the constraints allowed by Barberà et al. [1997] do not include the normalisation constraint that is fundamental to our setting. Several other papers [Border and Jordan, 1983, Peters et al., 1992, Barberà and Jackson, 1994] introduced multidimensional models in which one can achieve strategyproofness by taking a generalised median in each coordinate, but such a strategy does not work with normalisation constraints. We are not aware of results (prior to this work) that extend generalised medians to multiple dimensions without using a mechanism that decomposes into one-dimensional mechanisms.

In the computer science literature, the above-mentioned generalised median schemes have also been studied in the context of strategyproof facility location [Procaccia and Tennenholtz, 2013, Alon et al., 2010a]. In this context, the aim is to approximate social welfare subject to strategyproofness.

One could apply our results to the aggregation of probabilistic beliefs. There is a large literature on *probabilistic opinion pooling* [Genest and Zidek, 1986, French, 1985, Clemen, 1989, Intriligator, 1973] which studies aggregators in this context. The main focus of that literature is to preserve stochastic and epistemic properties. To the best of our knowledge, strategic aspects have not been considered.

Finally, recently proposed rules for crowdsourcing societal tradeoffs [Conitzer et al., 2015, 2016] can be used to aggregate budget divisions (with full support) after converting them into pairwise ratios of funding amounts, but this setting has also not been analysed from a strategic viewpoint.

4.2. Preliminaries

Let $N = \{1, \dots, n\}$ be a set of voters and $M = \{1, \dots, m\}$ be a set of *projects*. Voters have structured preferences over budget *divisions* (or *distributions*) $\mathbf{p} \in [0, 1]^m$, with $\sum_{j \in [m]} p_j = 1$, where p_j is the fraction of a public resource (such as money or time) allocated to project j. Each voter $i \in N$ has a most-preferred distribution $\mathbf{p_i} = (p_{i,1}, \dots, p_{i,m})$, with their preference over other distributions induced by ℓ_1 distance from $\mathbf{p_i}$. Specifically, each voter i has a disutility for distribution $\mathbf{q}$ equal to $d(\mathbf{p_i}, \mathbf{q})$, where $d(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^{m} |x_i - y_i|$ denotes the ℓ_1 distance between $\mathbf{x}$ and $\mathbf{y}$. Note that a voter's complete preference over all possible distributions can be deduced from the voter's most-preferred distribution $\mathbf{p_i}$.

A *preference profile* $\mathbf{P} = (\mathbf{\hat{p}_1}, \mathbf{\hat{p}_2}, \dots, \mathbf{\hat{p}_n})$ consists of a reported distribution $\mathbf{\hat{p}_i}$ for each voter i. We use $\mathbf{P_{-i}}$ to denote the reports of all voters other than i. A *budget aggregation mechanism* $\mathcal{A}$ takes as input a preference profile $\mathbf{P}$, and outputs an aggregate distribution $\mathcal{A}(\mathbf{P})$. A mechanism is *continuous* if it is continuous when considered as a function $\mathcal{A} : (\mathbb{R}^m)^n \to \mathbb{R}^m$. We say that a mechanism is *anonymous* if its output is fixed under permutations of the voters, and *neutral* if a permutation of the projects in voters' inputs permutes the output in the same way.

We are interested in mechanisms that satisfy *strategyproofness*. Voters should not be able to change the aggregate distribution in their favour by misrepresenting their preference.

Definition 4.1. A budget aggregation mechanism $\mathcal{A}$ satisfies strategyproofness if, for all preference profiles $\mathbf{P}$, voters i, and distributions $\mathbf{p_i}$ and $\mathbf{\hat{p}_i}$, $d(\mathcal{A}(\mathbf{P}_{-i}, \mathbf{\hat{p}_i}), \mathbf{p_i}) \geqslant d(\mathcal{A}(\mathbf{P_{-i}}, \mathbf{p_i}), \mathbf{p_i})$.

We are also interested in the basic efficiency notion of *Pareto-optimality*. It should not be possible to change the aggregate so that some voter is strictly better off but no other voter is worse off.

Definition 4.2. A budget aggregation mechanism $\mathcal{A}$ satisfies Pareto-optimality if, for all preference profiles $\mathbf{P}$, and all distributions $\mathbf{q}$, if $d(\mathcal{A}(\mathbf{P}), \mathbf{\hat{p}_i}) > d(\mathbf{q}, \mathbf{\hat{p}_i})$ for some voter i, then there exists a voter j for which $d(\mathcal{A}(\mathbf{P}), \mathbf{\hat{p}_j}) < d(\mathbf{q}, \mathbf{\hat{p}_j})$.

We also consider a fairness property that we call proportionality: Suppose each voter is single-minded, in that they prefer a distribution in which the entire resource goes to a single project. Then it is natural to split the resource in proportion to the number of voters supporting each project. For example, if 6 voters report $(1, 0, 0)$, 3 voters report $(0, 1, 0)$, and 1 voter reports $(0, 0, 1)$, then the aggregate should be $(0.6, 0.3, 0.1)$.

Definition 4.3. A voter is *single-minded* if their preferred distribution is a unit vector. A budget aggregation mechanism $\mathcal{A}$ is *proportional* if, for every preference profile $\mathbf{P}$ consisting of only single-minded voters, and every project j, $\mathcal{A}(\mathbf{P})_j = n_j/n$, where n_j is the number of voters that support project j.

Proportionality is a fairly weak definition, only applying to a small subset of possible profiles. However, as we will see later, it is already strong enough to be incompatible with Pareto-optimality within the class of moving phantom mechanisms that we introduce in this chapter.

4.3. Two Projects

To build intuition, we begin by considering the case in which $m = 2$. Due to the normalisation of inputs and of the output, and with ℓ_1 preferences, the problem is perfectly one-dimensional in this case. This allows us to directly import Moulin's [1980] famous characterisation of *generalised median* rules as the only strategyproof mechanisms for voters with single-peaked preferences over a single-dimensional quantity.[1]

[1] Our preference model using ℓ_1 imposes slightly more structure than just single-peakedness, namely that voters are indifferent between points that are equidistant to their peak. However, this restriction does not enlarge the class of strategyproof mechanisms, at least if we impose continuity [Massó and de Barreda, 2011].

Theorem 4.4 (Moulin, 1980)**.** *For $m = 2$, an anonymous and continuous budget aggregation mechanism $\mathcal{A}$ is strategyproof if and only if there are $\alpha_0 \geqslant \alpha_1 \geqslant \cdots \geqslant \alpha_n$ in $[0, 1]$ such that, for all profiles $\mathbf{P}$,*

$$\mathcal{A}(\mathbf{P})_1 = \mathrm{med}(p_{1,1}, p_{2,1}, \ldots, p_{n,1}, \alpha_0, \alpha_1, \ldots, \alpha_n),$$
$$\mathcal{A}(\mathbf{P})_2 = \mathrm{med}(p_{1,2}, p_{2,2}, \ldots, p_{n,2}, 1 - \alpha_0, 1 - \alpha_1, \ldots, 1 - \alpha_n).$$

The numbers α_k are known as *phantoms*. Each mechanism described by Theorem 4.4 can be understood as taking the coordinate-wise median of the reported distributions, after inserting $n + 1$ phantom voters (whose report is fixed and independent of the input profile).

One can check that $\alpha_0, \ldots, \alpha_n$ define a neutral mechanism if and only if the phantom placements are symmetric, that is if and only if $(\alpha_0, \ldots, \alpha_n) = (1 - \alpha_n, \ldots, 1 - \alpha_0\}$. Note that there are $n + 1$ phantoms but only n voters, so that the phantoms can outweigh the voters. For example, when $\alpha_k = 1/2$ for all $k \in \{0, \ldots, n\}$ then the mechanism is just the constant mechanism returning $(1/2, 1/2)$. However, if we take $\alpha_0 = 1$ and $\alpha_n = 0$, then these two phantoms "cancel out" and there are only $n - 1$ phantoms left. In fact, one can check that the mechanism is Pareto-optimal if and only if $\alpha_0 = 1$ and $\alpha_n = 0$ [Moulin, 1980].

A particularly interesting example is the *uniform phantom mechanism*, obtained when placing the phantoms uniformly over the interval $[0, 1]$, so that $\alpha_k = 1 - k/n$ for each $k \in \{0, \ldots, n\}$. This placement of phantom voters appears in a paper by Caragiannis et al. [2016b]. They were aiming for mechanisms whose output is close to the mean, and they prove that the uniform phantom mechanism yields an aggregate that is closer to the mean than that obtained from any other phantom placements, in the worst case over inputs. The uniform phantom mechanism has other attractive properties, including being proportional in the sense of Definition 4.3.

Proposition 4.5. *For $m = 2$, the uniform phantom mechanism is the unique (anonymous and continuous) budget aggregation mechanism $\mathcal{A}$ that is both strategyproof and proportional.*

Proof. Theorem 4.4 gives us that $\mathcal{A}$ is strategyproof if and only if it can be written in terms of phantom medians. We therefore need only to consider the additional requirement of proportionality. The uniform phantom mechanism is proportional, because if $\mathbf{P}$ consists of $n - k$ voters reporting $(1, 0)$ and k voters reporting $(0, 1)$, then $\mathbf{A}(\mathbf{P})_1 = \alpha_k = (n - k)/n$, as required.

For uniqueness, suppose $\alpha_0, \ldots, \alpha_n$ are phantom positions that induce a proportional mechanism. Let $k \in \{0, \ldots, n\}$. We show that $\alpha_k = 1 - k/n$. Let $\mathbf{P}$ be a profile consisting of only single-minded voters with $n_1 = n - k$ voters reporting $\hat{\mathbf{p}}_{\mathbf{i}} = (1, 0)$. Then α_k is the median, and proportionality requires that $\alpha_k = n_1/n = (n - k)/n = 1 - k/n$. □

Another natural way to place the phantoms is one that takes the coordinate-wise median. When $n + 1$ is even, this is achieved by placing half the phantoms at 0 and the other half at 1, outputting precisely the median of the reported values on each coordinate. When $n + 1$ is odd, we place $n/2$ phantoms at 0, $n/2$ phantoms at 1, and we place a single phantom at $1/2$ to preserve neutrality. This mechanism outputs the point between the left and right medians that is closest to $1/2$. The resulting mechanism returns an aggregate $\mathbf{p}$ that minimises the sum of distances between the reports $(\mathbf{p_1}, \mathbf{p_2}, \ldots, \mathbf{p_n})$ and $\mathbf{p}$. We will generalise this mechanism for larger m in Section 4.6.

4.4. Moving Phantom Mechanisms

For $m = 2$, we have a complete picture of strategyproof mechanisms, thanks to Moulin's characterisation. For $m \geqslant 3$, it is less clear how to construct examples of strategyproof mechanisms. One could try to take a generalised median for each project independently, but the result of such a mechanism would not respect the normalisation constraint.

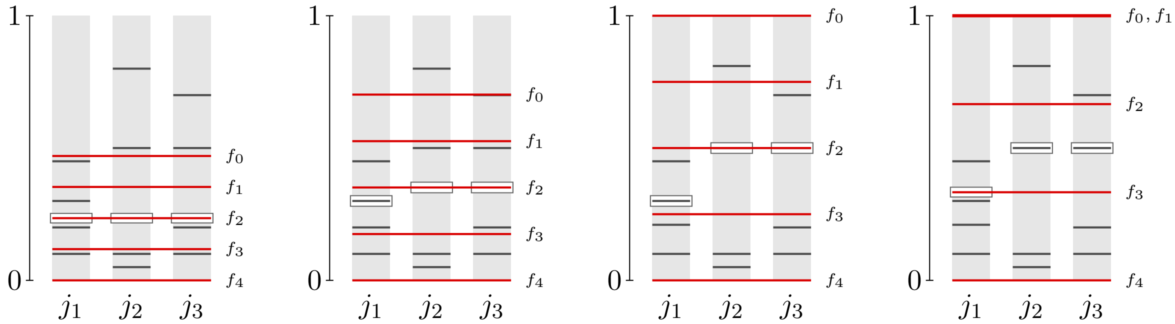


Figure 4.1.: A moving phantom mechanism operating on an instance with $n = 4$ and $m = 3$.

However, there is a way of extending the idea of generalised medians to the higher-dimensional setting. The basic idea is that if a coordinate-wise generalised median violates the normalisation constraint, then we can adjust the placement of the phantoms, increasing or decreasing the sum of the generalised medians as needed. Such a procedure might, in principle, give voters incentives to manipulate in order to affect the phantom placements. However, our class of *moving phantom mechanisms* manages to avoid this problem.

Definition 4.6. Let $\mathcal{F} = \{f_k : k \in \{0, \ldots, n\}\}$ be a family of functions, called a *phantom system*, where $f_k : [0,1] \to [0,1]$ is a continuous, weakly increasing function with $f_k(0) = 0$ and $f_k(1) = 1$ for each k, and we have $f_0(t) \geqslant f_1(t) \geqslant \cdots \geqslant f_n(t)$ for all $t \in [0,1]$. Then, the *moving phantom mechanism* $\mathcal{A}^{\mathcal{F}}$ is defined so that for all profiles $\mathbf{P}$ and all $j \in [m]$,

$$\mathcal{A}^{\mathcal{F}}(\mathbf{P})_j = \text{med}(f_0(t^*), \ldots, f_n(t^*), \hat{p}_{1,j}, \ldots, \hat{p}_{n,j}), \tag{4.1}$$

where t^* is chosen so that $t^* \in \{t : \sum_{j\in[m]} \text{med}(f_0(t), \ldots, f_n(t), \hat{p}_{1,j}, \ldots, \hat{p}_{n,j}) = 1\}$.

For brevity, we write $\mathcal{F}(t) = (f_0(t), \ldots, f_n(t))$ and abbreviate the median in (4.1) to $\text{med}(\mathcal{F}(t), \mathbf{P}_{i\in[n],j})$.

Let us examine the definition. Each f_k represents a phantom, and the phantom system $\mathcal{F}$ represents a "movie" in which all phantoms continuously increase from 0 to 1, with the function argument t defining an instantaneous snapshot of the phantom positions. The moving phantom mechanism $\mathcal{A}^{\mathcal{F}}$ defined by $\mathcal{F}$ identifies a particular snapshot in time, t^*, for which the sum of generalised medians over all coordinates is exactly 1. One can check that at least one such t^* exists, and that the output of the mechanism is independent of which of these t^* is chosen.

Proposition 4.7. *The moving phantom mechanism $\mathcal{A}^{\mathcal{F}}$ is well-defined for every phantom system $\mathcal{F}$ satisfying the conditions of Definition 4.6.*

Proof. First note that the function $t \mapsto \sum_{j\in[m]} \text{med}(\mathcal{F}(t), \mathbf{P}_{i\in[n],j})$ is continuous and increasing in t, because f_k is continuous and increasing, and these properties are preserved under taking the median and sum. This implies that, provided the set $\{t : \sum_{j\in[m]} \text{med}(\mathcal{F}(t), \mathbf{P}_{i\in[n],j}) = 1\}$ is non-empty, the aggregate $\mathcal{A}^{\mathcal{F}}(\mathbf{P})$ does not depend on the choice of t^*.

When $t = 0$, $\sum_{j\in[m]} \text{med}(\mathcal{F}(t), \mathbf{P}_{i\in[n],j}) = 0$, since all $n+1$ phantom entries are 0. When $t = 1$, $\sum_{j\in[m]} \text{med}(\mathcal{F}(t), \mathbf{P}_{i\in[n],j}) = m > 1$, since all $n+1$ phantom entries are 1. By the Intermediate Value Theorem, using continuity, there exists $t \in [0,1]$ with $\sum_{j\in[m]} \text{med}(\mathcal{F}(t), \mathbf{P}_{i\in[n],j}) = 1$. □

To build intuition, we consider an example moving phantom mechanism in Figure 4.1. There are three projects, each occupying a column on the horizontal axis, and four voters. Voter reports are indicated by grey horizontal line segments, with their magnitude $\hat{p}_{i,j}$ indicated by their vertical position. The phantom placements are indicated by the red lines and labelled $f_0, \ldots, f_4$.

For each project, the median of the four agent reports and the five phantoms is indicated by a rectangle.

The four snapshots shown in Figure 4.1 display increasing values of t. Observe that the position of each phantom (weakly) increases from left to right, as does the median on each project. Although the vertical axis is not labelled, for simplicity of presentation, normalisation here occurs in the second image from the left. In the leftmost image, the sum of the highlighted entries is less than 1, while in the two rightmost images it is more than 1.

For simplicity, the definition of moving phantom mechanisms treats the number of voters n as fixed. To allow n to vary, it is necessary to define a family of phantom systems, one for each n. In the next two sections, we give two examples of such families, but for this section we keep the presentation simple by considering only a fixed n.

Moving phantom mechanisms satisfy some important basic properties. They are all anonymous and neutral. Here neutrality is a design choice, and one could imagine defining moving phantom mechanisms for which the movement of the phantoms depends on the project. All moving phantom mechanisms are also continuous.

Given a profile, we can efficiently approximate the output of a moving phantom mechanism, assuming oracle access to its defining functions $\mathcal{F}$, by performing a binary search on t. In principle, the precise time $t^* \in [0,1]$ at which the output of the mechanism is normalised may have many decimal digits, and for badly-behaved $\mathcal{F}$ it may even be irrational. For the same reason, the mechanism may return an irrational distribution, so the precise computation of the output may not be possible. However, for the mechanisms studied in the following sections, we can show that t^* has few digits and the output is always rational, so polynomial-time computation is possible.

We now show our main result in this section, that every moving phantom mechanism is strategyproof. Before proving the result formally, we provide some intuition. If i changes i's report from $\mathbf{p_i}$ to $\hat{\mathbf{p}}_\mathbf{i}$, the effect on the aggregate can be decomposed into two parts. First, we can think of holding the phantoms fixed at the snapshot dictated by the truthful instance, while changing i's report to $\hat{\mathbf{p}}_\mathbf{i}$. Second, we can think of repositioning the phantoms to the snapshot required to guarantee normalisation of the aggregate vector after i reports $\hat{\mathbf{p}}_\mathbf{i}$. To prove strategyproofness, we show that any change that the aggregate distribution undergoes in the first stage can only be bad for voter i, pushing the aggregate away from $\mathbf{p_i}$. Change in the second stage can push the aggregate towards $\mathbf{p_i}$, helping voter i, but the magnitude of this change is upper bounded by the magnitude of the harmful change in the first stage.

Theorem 4.8. *Every moving phantom mechanism is strategyproof.*

Proof. Let $\mathcal{F}$ define a moving phantom mechanism $\mathcal{A}^{\mathcal{F}}$. Consider some report $\hat{\mathbf{p}}_\mathbf{i} \neq \mathbf{p_i}$, and fix the reports of all other voters $\mathbf{P_{-i}}$. Let t^* determine the phantom placement for reports $(\mathbf{p_i}, \mathbf{P_{-i}})$ and $\hat{t}^*$ for reports $(\hat{\mathbf{p}}_\mathbf{i}, \mathbf{P_{-i}})$.

Consider the effect of i's misreport from $\mathbf{p_i}$ to $\hat{\mathbf{p}}_\mathbf{i}$ while *holding the phantom placement fixed at* $\mathcal{F}(t^*)$. Then, because phantom placements are fixed on each project, any change that voter i can cause on project j by misreporting must be away from i's preference $p_{i,j}$. For each $j \in [m]$,

- if $p_{i,j} \leqslant \mathcal{A}^f(\mathbf{p_i}, \mathbf{P_{-i}})_j < \hat{p}_{i,j}$, then $\text{med}(\mathcal{F}(t^*), \hat{p}_{i,j}, \mathbf{P}_{-i,j}) \geqslant \text{med}(\mathcal{F}(t^*), p_{i,j}, \mathbf{P}_{-i,j})$;
- if $\hat{p}_{i,j} \leqslant \mathcal{A}^f(\mathbf{p_i}, \mathbf{P_{-i}})_j < p_{i,j}$, then $\text{med}(\mathcal{F}(t^*), \hat{p}_{i,j}, \mathbf{P}_{-i,j}) \leqslant \text{med}(\mathcal{F}(t^*), p_{i,j}, \mathbf{P}_{-i,j})$;
- if $\hat{p}_{i,j}$ and $p_{i,j}$ lie on same side of $\mathcal{A}^f(\mathbf{p_i}, \mathbf{P_{-i}})_j$, $\text{med}(\mathcal{F}(t^*), \hat{p}_{i,j}, \mathbf{P}_{-i,j}) = \text{med}(\mathcal{F}(t^*), p_{i,j}, \mathbf{P}_{-i,j})$.

Let $y_j = \text{med}(\mathcal{F}(t^*), \hat{p}_{i,j}, \mathbf{P}_{-i,j}) - \text{med}(\mathcal{F}(t^*), p_{i,j}, \mathbf{P}_{-i,j})$ denote the change caused on project j by voter i's misreport, subject to holding the phantom placement fixed at $\mathcal{F}(t^*)$. By the above, the ℓ_1 distance from i's preferred distribution has increased by $\sum_{j\in[m]} |y_j|$ as a result of i's misreport.

Next, we consider the change that results from moving the phantoms from $\mathcal{F}(t^*)$ to $\mathcal{F}(\hat{t}^*)$. Assume that $\sum_{j\in[m]} y_j \geqslant 0$ (otherwise, a very similar argument applies). Then we have that $\sum_{j\in[m]} \text{med}(\mathcal{F}(t^*), \hat{p}_{i,j}, \mathbf{P}_{-i,j}) \geqslant 1$, which implies that $\hat{t}^* \leqslant t^*$ since the sum is monotonic in t (see the proof of Proposition 4.7). This produces aggregate distribution $\mathcal{A}^{\mathcal{F}}(\hat{\mathbf{p}}_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})$ with $\mathcal{A}^{\mathcal{F}}(\hat{\mathbf{p}}_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})_j = \text{med}(\mathcal{F}(\hat{t}^*), \hat{p}_{i,j}, \mathbf{P}_{-i,j}) \leqslant \text{med}(\mathcal{F}(t^*), \hat{p}_{i,j}, \mathbf{P}_{-i,j})$ for all j, and $\sum_{j\in[m]}(\text{med}(\mathcal{F}(t^*), \hat{p}_{i,j}, \mathbf{P}_{-i,j}) - \mathcal{A}^{\mathcal{F}}(\mathbf{P})_j) = \sum_{j\in[m]} y_j$. That is, the ℓ_1 distance between taking generalised medians with phantoms defined by t^* and doing so with phantoms defined by $\hat{t}^*$, conditioned on voter i reporting $\hat{\mathbf{p}}_\mathbf{i}$, is at most $\sum_{j\in[m]} y_j$.

Therefore, $d(\mathcal{A}^{\mathcal{F}}(\hat{\mathbf{p}}_\mathbf{i}, \mathbf{P}_{-\mathbf{i}}), \mathbf{p}_\mathbf{i}) \geqslant d(\mathcal{A}^{\mathcal{F}}(\mathbf{p}_\mathbf{i}, \mathbf{P}_{-\mathbf{i}}), \mathbf{p}_\mathbf{i}) + \sum_j |y_j| - \sum_j y_j \geqslant d(\mathcal{A}^{\mathcal{F}}(\mathbf{p}_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})$. □

In addition to strategyproofness, moving phantom mechanisms satisfy a natural monotonicity property that says that if some voter increases the report on project j, and decreases the report on all other projects, then the aggregate weight on project j should not decrease.

Definition 4.9. A budget aggregation mechanism $\mathcal{A}$ satisfies *monotonicity* if, for all $\mathbf{p}_\mathbf{i}$, $\mathbf{p}'_\mathbf{i}$ with $p_{i,j} > p'_{i,j}$ for some j and $p_{i,k} \leqslant p'_{i,k}$ for all $k \neq j$,

$$\mathcal{A}(\mathbf{p}_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})_j \geqslant \mathcal{A}(\mathbf{p}'_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})_j.$$

Theorem 4.10. *Every moving phantom mechanism satisfies monotonicity.*

Proof. Let $\mathbf{p}_\mathbf{i}$, $\mathbf{p}'_\mathbf{i}$ be such that $p_{i,j} > p'_{i,j}$ for some j and $p_{i,k} \leqslant p'_{i,k}$ for all $k \neq j$. Let t^* determine the phantom placement for reports $(\mathbf{p}_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})$ and t'^* for reports $(\mathbf{p}'_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})$.

Suppose that $t'^* < t^*$. We have

$$\mathcal{A}(\mathbf{p}'_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})_j = \text{med}(\mathcal{F}(t'^*), p'_{i,j}, P_{-i,j}) \leqslant \text{med}(\mathcal{F}(t^*), p_{i,j}, P_{-i,j}) = \mathcal{A}(\mathbf{p}_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})_j$$

where the inequality holds because $p_{i,j} > p'_{i,j}$ and $f_k(t^*) \geqslant f_k(t'^*)$ for all $k \in \{0, \ldots, n\}$.

Next, suppose that $t'^* > t^*$. Then

$$\begin{aligned}\mathcal{A}(\mathbf{p}'_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})_j = 1 - \sum_{k\neq j} \mathcal{A}(\mathbf{p}'_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})_k &= 1 - \sum_{j'\neq j} \text{med}(\mathcal{F}(t'^*), p'_{i,j'}, P_{-i,j'}) \\ &\leqslant 1 - \sum_{j'\neq j} \text{med}(\mathcal{F}(t^*), p_{i,j'}, P_{-i,j'}) = \mathcal{A}(\mathbf{p}_\mathbf{i}, \mathbf{P}_{-\mathbf{i}})_j\end{aligned}$$

where the inequality holds because $p_{i,j'} < p'_{i,j'}$ for all $j' \neq j$ and $f_k(t^*) \leqslant f_k(t'^*)$ for all $k \in \{0, \ldots, n\}$. □

Before we move on to particular moving phantom mechanisms, let us end this section with a tantalising open question: Does there exist an (anonymous, neutral, continuous) strategyproof budget aggregation mechanism that is *not* a moving phantom mechanism? We have not been able to construct any example, and have found that some mechanisms that on first sight seem to have nothing to do with medians end up having an equivalent description as a moving phantom mechanism. For the simpler two-project case, we already have a characterisation of all strategyproof mechanisms (Theorem 4.4). This class can equivalently be described in terms of moving phantoms, and so the answer to our question for $m = 2$ is *no*.

Theorem 4.11. *For $m = 2$, moving phantom mechanisms are the only budget aggregation mechanisms that satisfy anonymity, neutrality, continuity, and strategyproofness.*

Proof. Certainly all moving phantom mechanisms satisfy these properties. For the other direction, we know from Theorem 4.4 that any mechanism $\mathcal{A}$ satisfying these properties can be described as a generalised median with phantoms $\alpha_0, \ldots, \alpha_n$ satisfying, due to neutrality, $\{\alpha_0, \ldots, \alpha_n\} = \{1 - \alpha_0, \ldots, 1 - \alpha_n\}$. We show that $\mathcal{A}$ is equivalent to a moving phantom mechanism. Define $\mathcal{A}^{\mathcal{F}}$ using a phantom system $\mathcal{F}$ for which there exists a $t^* \in [0, 1]$ with $f_k(t^*) = \alpha_k$ for every $k \in \{0, \ldots, n\}$. Then, for every preference profile $\mathbf{P}$, we have that $\mathcal{A}^{\mathcal{F}}(\mathbf{P})_1 = \text{med}(\mathcal{F}(t^*), \mathbf{P}_{i\in[n],j}) = (\alpha_0, \ldots, \alpha_n, \mathbf{P}_{i\in[n],j})$, and $\mathcal{A}^{\mathcal{F}}(\mathbf{P})_2 = 1 - \mathcal{A}^{\mathcal{F}}(\mathbf{P})_1$, matching the output of $\mathcal{A}$. □

4.5. The Independent Markets Mechanism

We have seen that uniform phantoms is uniquely proportional for $m = 2$. By a similar argument to the proof of Proposition 4.5, it is easy to see that any family of functions $\mathcal{F}$ that generates uniform phantoms at some snapshot will be proportional, and will reduce to the uniform phantom mechanism for $m = 2$. However, this leaves a large class of moving phantom mechanisms to choose from. In this section, we identify a particular moving phantom mechanism that generalises the uniform phantom mechanism for arbitrary m. Its output can be interpreted as a market equilibrium.

Definition 4.12. The independent markets mechanism ($\mathcal{A}^{IM}$) is the moving phantom mechanism defined by the phantom system $f_k(t) = \min\{t(n-k), 1\}$ for each $k \in \{0, \ldots, n\}$.

To visualise the phantom placement, observe that for any $t \leqslant 1/n$, phantoms are being placed at $0, t, 2t, \ldots, nt$. Once t reaches $1/n$, phantoms continue to grow in the same manner, but the higher phantoms get capped at 1.[2] This is actually the mechanism that we displayed in Figure 4.1. Note that, when $t = 1/n$, the phantom placement is uniform on $[0, 1]$ (as is the case in the third panel of Figure 4.1); thus, $\mathcal{A}^{IM}$ reduces to the uniform phantom mechanism for $m = 2$.

Example 4.13. Let us consider a simple numerical example. Let $n = m = 3$, and suppose voter reports are $\mathbf{p_1} = (0, 0.5, 0.5), \mathbf{p_2} = (0.5, 0.5, 0)$, and $\mathbf{p_3} = (0.9, 0, 0.1)$. Consider the placement of the $n+1 = 4$ phantoms when $t = 0.6$. They are placed at $f_0(t) = 0.6, f_1(t) = 0.4, f_2(t) = 0.2, f_3(t) = 0$. On the first project,

$$\text{med}\{f_0(t), f_1(t), f_2(t), f_3(t), p_{1,1}, p_{2,1}, p_{3,1}\} = \text{med}\{0.6, 0.4, 0.2, 0, 0, 0.5, 0.9\} = 0.4.$$

Similarly, it is easy to check that the generalised median on the second project is 0.4 and on the third project is 0.2. Because these are normalised, $t^* = 0.6$ is a valid choice of t^*, and the outcome $\mathcal{A}^{IM}(\mathbf{P}) = (0.4, 0.4, 0.2)$.

4.5.1. Market Interpretation

Why do we call this mechanism the independent markets mechanism? To explain this, we first establish a connection between the market clearing price in a simple single-good market and the median of some familiar-looking numbers.

Suppose we are selling a single divisible good, of which a total amount of $x \in [0, \infty)$ is available. Each of n voters has a budget of 1, and a value $v_i \in [0, \infty)$ per unit of the good. At a price $\pi \geqslant 0$ per unit of the good, the *demand* of voter i, $D_i(\pi)$ is given by the following function:

$$D_i(\pi) = \begin{cases} \infty & \pi = 0, \\ \frac{1}{\pi} & 0 < \pi < v_i, \\ 0 & \pi \geqslant v_i \text{ and } \pi > 0. \end{cases}$$

Thus, each voter demands as much of the good as their budget of 1 allows at price π, as long as the price per unit is lower than their value per unit. The *market clearing price* c is the price at which the supply of the good (x) equals the total demand. Formally,

$$c = \sup\{\pi : \textstyle\sum_{i \in [n]} D_i(\pi) > x\}, \tag{4.2}$$

where the supremum is necessary because, due to discontinuities in the demand function, supply and demand may never be exactly equal.

[2] As written, $f_n(1) = 0$, but Definition 4.6 requires $f_k(1) = 1$ for all k. This detail does not matter here, since normalisation is always achieved without moving phantom n, but one could write f_n in a different form to satisfy Definition 4.6 without it changing the behaviour of the mechanism.

It turns out that the market clearing price c is equal to the median of the n voter values v_i and the $n+1$ "phantom values" which are uniformly distributed on the interval $[0, n/x]$. To the best of our knowledge, this connection has not previously been appreciated in the literature.

Lemma 4.14. *In the market defined above, the market clearing price c equals*

$$\text{med}(0, 1/x, \ldots, (n-1)/x, n/x, v_1, \ldots, v_n).$$

Proof. We distinguish the cases that the median is a phantom entry or a voter entry. Suppose that the median is a/x for some a. Then we can partition the (real and phantom) entries, with the exception of the phantom at a/x, into sets A and B with $|A| = |B| = n$, where A consists only of entries less than or equal to a/x, and B consists only of entries greater than or equal to a/x.

The set B contains $n-a$ phantoms, so $n-(n-a) = a$ voter reports. At any price $\pi < a/x$, each voter $i \in B$ has demand $D_i(\pi) = 1/\pi > x/a$. The total demand of all voters in B is therefore greater than x. At price $\pi = a/x$, each voter $i \in B$ has demand $D_i(\pi) = 1/\pi = x/a$ (if $v_i > a/x$) or $D_i(\pi) = 0$ (if $v_i = a/x$), and each voter $i \notin B$ has demand 0. Therefore the total demand of all voters is at most x, so the market clearing price is a/x.

Next, suppose that the generalised median is $a/x < y < (a+1)/x$ for some $a \leqslant n-1$ (note that the generalised median cannot be greater than n/x, because it cannot be higher than the largest phantom value). Then we can partition the (real and phantom) entries, not including a single voter with $v_i = y$ (one such voter must exist because the median coincides with some entry, and no phantom entry lies at y), into sets A and B each of size n, where A consists only of entries less than or equal to y, and B consists only of entries greater than or equal to y.

Again, B contains $n-a$ phantom reports, so a voter reports. At all prices $\pi < y$, each of these a voters, as well as voter i with $\pi_i = y$, has demand $1/\pi > 1/y$. The total demand is thus greater than $(a+1)/y > x$. At price $\pi = y$, the total demand is at most $a/y < x$ (since the number of voters with $v_i > y$ is at most the number voter reports in set B). The market clearing price is therefore y. ☐

The "market" connection to independent markets is now clear: For each project j, we set up a market in which we sell an amount x of a good; this amount is the same across markets. Voter $i \in [n]$ has value $\hat{p}_{i,j}$ for the good sold in market j, and has a budget of 1 in each market. The markets are "independent" because, while each voter is engaged in every market, the budget of 1 for each market can only be used to buy the good sold in that market. Using Lemma 4.14, we can derive the market clearing prices in each of these markets. If we write $t = n/x$, then these prices correspond exactly to the output of $\mathcal{A}^{IM}$ with the phantoms as placed at time t. Changing the phantom placement by varying t to normalise the output is equivalent to varying the amount x of the good sold in each market until the clearing prices across markets sum to 1. While we prevent phantoms from moving above 1 in the definition of independent markets to comply with Definition 4.6, the exact positions of these phantoms do not affect the clearing price since all reports are at most 1.

Returning to Example 4.13, we can verify the outcome using the market interpretation, by setting the quantity of goods to be sold in each market to $x^* = n/t^* = 5$. In the market corresponding to project 1, the market clears at price $\pi_1 = 0.4$, at which price voters 2 and 3 demand $1/\pi_1 = 2.5$ goods each, matching supply, and voter 1 demands nothing as $p_{1,1} = 0 < \pi$. It can be checked that the market prices also match the independent markets outcome for projects 2 and 3.

The market system we have described yields an strategyproof aggregator, since it corresponds to a moving phantom mechanism. There are other market-based aggregation mechanisms described in the literature, most famously the parimutuel consensus mechanism of Eisenberg and Gale

[1959]. That mechanism differs from ours in that voters have only a single budget of 1 which they can use in all of the markets. (The supply of goods can be fixed at $x = n$, which guarantees that prices are normalised, because total spending is fixed.) For the case $m = 2$, it does not matter whether markets are independent or not, and our mechanism is equivalent to the one of Eisenberg and Gale [1959]. It follows that the parimutuel consensus mechanism is strategyproof for $m = 2$ (in our ℓ_1 sense). However, for $m \geqslant 3$, the mechanism is manipulable,[3] and hence cannot be represented as a moving phantom mechanism. We point the reader to the work of Garg et al. [2018] for a detailed overview of other settings in which market mechanisms have been used in the context of public decision making.

4.5.2. Other Properties of Independent Markets

The independent markets mechanism can be seen as generalising the uniform phantom placement that guaranteed proportionality in the $m = 2$ case, and for a similar reason, the independent markets mechanism satisfies proportionality.

Proposition 4.15. *$\mathcal{A}^{IM}$ satisfies proportionality.*

Proof. When $\mathcal{A}^{IM}$ is run on a profile of single-minded reports, and we stop the phantom movement at $t^* = \frac{1}{n}$, then the generalised medians correspond to the proportional output. Since this is clearly normalised, this is what $\mathcal{A}^{IM}$ returns. □

We next check that the independent markets outcome is always rational and can be described in polynomially many bits, thus ensuring that it can be computed efficiently as suggested in Section 4.4. Our argument proceeds by showing that the outcome is a solution of a linear program, similar to a proof of rationality for the parimutuel consensus mechanism [Vazirani, 2007, Thm. 5.1]. Consider the outcome $\mathbf{p}$ of the independent markets, and write $N_j = \{i \in N : p_j < \hat{p}_{i,j}\}$ for the set of voters that purchase good $j \in [m]$ since their value is lower than its price. Now, if x is the supply of each good, then the amount xp_j of money spent on j equals the budget of the demanders, which is $|N_j|$. Introducing a variable $z \equiv 1/x$, we can write this as $p_j = z \cdot |N_j|$. Thus, $\mathbf{p}$ is the solution of maximising ε subject to

$$\begin{aligned}
\hat{p}_{i,j} &\leqslant p_j - \varepsilon && \text{for } j \in [m] \text{ and } i \in N_j,\\
\hat{p}_{i,j} &\geqslant p_j && \text{for } j \in [m] \text{ and } i \in N \setminus N_j,\\
p_j &= z \cdot |N_j| && \text{for } j \in [m],\\
\textstyle\sum_{j\in[m]} p_j &= 1, \quad p_j \geqslant 0, \quad z \geqslant 0 && \text{for } j \in [m].
\end{aligned}$$

Using standard encoding techniques, one can also calculate the independent markets mechanism using an ILP with binary variables encoding "$i \in N_j$."

4.6. Pareto-Optimality and Social Welfare

The independent markets mechanism has several natural interpretations, and it satisfies proportionality. However, it is not Pareto-optimal. If voter 1 reports $(0.8, 0.2, 0)$ and voter 2 reports $(0.8, 0, 0.2)$, then independent markets returns $(0.6, 0.2, 0.2)$, which is dominated by $(0.8, 0.1, 0.1)$. On this example, independent markets even fails to be *range-respecting*, which requires that $\min_{i\in[n]} \hat{p}_{i,j} \leqslant \mathcal{A}(\mathbf{P})_j \leqslant \max_{i\in[n]} \hat{p}_{i,j}$ for all $j \in [m]$.

The failure to be range-respecting can be fixed, if desired, by changing the positions of phantoms 0 and n. One can show that a moving phantom mechanism $\mathcal{A}^{\mathcal{F}}$ is range-respecting if and only if

[3] Let $\mathbf{p_1} = (0, 0.5, 0.5)$, $\mathbf{p_2} = (0.5, 0.5, 0)$. Parimutuel consensus yields prices $(1/3, 1/3, 1/3)$, at distance $2/3$ from $\mathbf{p_1}$. If voter 1 instead reports $\hat{\mathbf{p}}_\mathbf{1} = (0, 0, 1)$, the price vector is $(0.25, 0.25, 0.5)$, at distance 0.5 from $\mathbf{p_1}$.

$f_0(t) = 1$ and $f_n(t) = 0$ for all $t \in [0,1]$ except for an initial period where phantom 0 moves from 0 to 1 while all other phantoms remain at 0, and a period at the end where phantom n moves from 0 to 1 while all other phantoms are at 1. This mirrors a result in Section 4.3; if the outer two phantoms are at 0 and 1, the $n-1$ remaining phantoms cannot outweigh the n voter reports.

While many moving phantom mechanisms are range-respecting, it is much more difficult to find a mechanism in this class which is Pareto-optimal. Usually, it is possible to construct a profile in which the mechanism returns a vector $\mathbf{p}$ all of whose entries are phantom reports, and then a Pareto-improvement can be obtained by perturbing this vector in the directions where the majority of voter reports lie. Such a perturbation is not possible if the phantoms lie at 0 or 1, which turns out to be the only escape. As we prove below, no mechanism $\mathcal{A}^{\mathcal{F}}$ can be Pareto-optimal if there is any time point t when two phantoms are both strictly between 0 and 1.

This condition is extremely restrictive, and a moment's thought reveals that there is only one legal phantom system which avoids having two interior phantoms: All phantoms start at 0, and then, one by one, one of the phantoms is moved to 1. At each t, at most one phantom lies strictly between 0 and 1 while travelling. We call this phantom system $\mathcal{F}^*$. It can be formalised as

$$f_k(t) = \begin{cases} 0 & 0 \leqslant t \leqslant \frac{k}{n+1}, \\ t(n+1) - k & \frac{k}{n+1} < t < \frac{k+1}{n+1}, \\ 1 & \frac{k+1}{n+1} \leqslant t \leqslant 1. \end{cases}$$

Below we will show that $\mathcal{A}^{\mathcal{F}^*}$ precisely corresponds to the budget aggregation mechanism that maximises voter welfare, breaking ties in favour of the maximum entropy distribution. It will immediately follow that $\mathcal{A}^{\mathcal{F}^*}$ is indeed Pareto-optimal. Combined with Theorem 4.16 below, which shows that all other moving phantom mechanisms are Pareto-inefficient, this implies that the welfare-maximising mechanism is the unique Pareto-optimal moving phantom mechanism.

4.6.1. Characterising Pareto-Optimality

The proof of Theorem 4.16 shows, by induction, that each phantom needs to move all the way to 1 before the next phantom can leave its position at 0. In case this does not happen, based on the approximate phantom positions, we construct a profile where the mechanism is Pareto-inefficient. These constructions are of two kinds: an easier case when the interior phantoms are low (lying below $\frac{1}{n(n-1)}$), and a more involved case when one of the phantoms has moved higher. In both cases, our constructions utilise two types of projects. More voters report "high" probabilities on projects of the first type than on projects of the second type. The constructions work so that if two phantoms simultaneously take values between 0 and 1, then the mechanism outputs middling values on all projects. Social welfare can be improved by increasing the output on projects of the first type, and decreasing the output on projects of the second type. By incorporating enough symmetry between voters, we guarantee that social welfare gains are shared equally, so obtain a Pareto-improvement.

Theorem 4.16. *A moving phantom mechanism $\mathcal{A}^{\mathcal{F}}$ cannot be Pareto-optimal for any $m \geqslant n^2$ unless $\mathcal{A}^{\mathcal{F}} = \mathcal{A}^{\mathcal{F}^*}$.*

Proof. We first show that any Pareto-optimal moving phantom mechanism $\mathcal{A}^{\mathcal{F}}$ for which there exists a t with $f_0(t) < 1$ and $f_1(t) > 0$ can be equivalently expressed as a moving phantom mechanism that does not have such a t. Suppose that $\mathcal{A}^{\mathcal{F}} = \text{med}(f_0(t^*), \dots, f_n(t^*), \hat{p}_{1,j}, \dots, \hat{p}_{n,j}) = f_0(t^*) < \bar{p}_{1,j}$ for some j. Then $\mathcal{A}^{\mathcal{F}}$ is not Pareto-optimal, because increasing $\mathcal{A}^{\mathcal{F}}(\mathbf{P})_j$ and decreasing $\mathcal{A}^{\mathcal{F}}(\mathbf{P})_{j'}$ for any coordinate with $\mathcal{A}^{\mathcal{F}}(\mathbf{P})_{j'} > \bar{p}_{1,j'}$ is a Pareto-improvement (such a coordinate must exist, because $\sum_j \bar{p}_{1,j} \leqslant 1$). Therefore, for all preference profiles $\mathbf{P}$, $\mathcal{A}^{\mathcal{F}} = \text{med}(f_0(t^*), \dots, f_n(t^*), \hat{p}_{1,j}, \dots, \hat{p}_{n,j}) \geqslant \bar{p}_{1,j}$. This implies that $f_0(t^*) \geqslant \bar{p}_{1,j}$ and so the exact position of $f_0(t^*)$ has no effect on the mechanism. It would be equivalent to move phantom f_0 to position 1 before moving phantom f_1.

A very similar argument can be used to show that there cannot exist a t for which $f_{n-1}(t) < 1$ and $f_n(t) > 0$. For the rest of the proof, we focus on the intermediate phantoms. Suppose that there exists some index $1 \leqslant k \leqslant n-2$ for which $f_k(t) < 1$ and $f_{k+1}(t) > 0$ for some t. If no such k exists, then phantom system $\mathcal{F} = \mathcal{F}^*$.

We next show that if $f_k(t) < \frac{1}{n(n-1)}$, it must be the case that $x_{k+1}(t) = 0$. Define an instance with $m = n^2$ projects. Voter $i \in [n]$ reports $\hat{p}_{i,j} = \frac{1}{n^2-kn-1}$ for projects $j \in \{((i-1)(n-1)+1, \ldots, (i-1)(n-1)+n^2-n-kn+k) \mod n(n-1)\}$ and for projects $j \in \{n(n-1)+(i, i+1, i+n-k-2) \mod n\}$, and $\hat{p}_{i,j} = 0$ for all other projects. Note that $\sum_{j=1}^{m} \hat{p}_{i,j} = 1$. Further, note that among projects $1, ..., n(n-1)$, each voter makes $n^2-n-kn+k = (n-1)(n-k)$ non-zero reports and each project has $n-k$ non-zero reports, while among projects $n(n-1)+1, \ldots, n^2$, each voter makes $n-k-1$ non-zero reports and each project has $n-k-1$ non-zero reports. Therefore, if $f_k(t) < \frac{1}{n^2-kn-1}$, the generalised median on project j is $f_k(t)$ for $j \in \{1, \ldots, n(n-1)\}$ and the median on project j is $f_{k+1}(t)$ for $j \in \{n+1, \ldots, 2n\}$.

Suppose that there exists t for which $f_k(t) < \frac{1}{n(n-1)} < \frac{1}{n^2-kn-1}$ and $f_{k+1}(t) > 0$. Then, since f is increasing and continuous, and the aggregate distribution $\mathcal{A}^{\mathcal{F}}(\mathbf{P})$ is normalised, it will necessarily be the case that for all $j \in \{1, ..., n(n-1)\}$, $\mathcal{A}^{f}(\mathbf{P})_j = f_k(t^*) < \frac{1}{n(n-1)}$, and for all $j \in \{1, ..., n(n-1)\}$, $\mathcal{A}^{f}(\mathbf{P})_j = f_{k+1}(t^*) > 0$, with $n(n-1)f_k(t^*) + nf_{k+1}(t^*) = 1$. But this is not Pareto-optimal. Consider, for some small enough ε, increasing $\mathcal{A}^{\mathcal{F}}(\mathbf{P})_j$ by ε on projects $j \in \{1, \ldots, n(n-1)\}$, and decreasing $\mathcal{A}^{\mathcal{F}}(\mathbf{P})_j$ by $\varepsilon(n-1)$ on projects $j \in \{n(n-1)+1, \ldots, n^2\}$. For every voter i, there are $n^2-n-kn+k$ projects on which the aggregate moves ε closer to i's report, $kn-k$ projects for which the aggregate moves ε farther from i's report, $n-k-1$ projects on which the aggregate moves $\varepsilon(n-1)$ further from i's report, and $k+1$ projects for which the aggregate moves $\varepsilon(n-1)$ closer to i's report. Summing these up, the change moves the aggregate $(2n-2)\varepsilon$ closer to $\hat{\mathbf{p}}_{\mathbf{i}}$ in ℓ_1 distance.

We now show that if $f_k(t) < 1$ then it must be the case that $f_{k+1}(t) = 0$. For contradiction suppose otherwise. Let $\bar{t} = \sup\{t : f_{k+1}(t) = 0\}$ be the final snapshot at which $f_{k+1}(t) = 0$. By assumption, $f_k(\bar{t}) < 1$. We define an instance similar to that above. Let $\delta > 0$ (we will determine the exact value of δ later). For every voter i, let $\hat{p}_{i,j} = \frac{1-f_k(\bar{t})-\delta}{n(n-1)-1}$ for every $j \in \{(i-1)(n-1)+1, \ldots, (i-1)(n-1)+n-1\}$ and $\hat{p}_{i,j} = \frac{1-f_k(\bar{t})}{n(n-1)-1}$ for every $j \in \{((i-1)(n-1)+n, \ldots, (i-1)(n-1)+n^2-n-kn+k) \mod n(n-1)\}$. However, for $j = 1$, for every voter i with $p_{i,1} = \frac{1-f_k(\bar{t})-\delta}{n(n-1)-1}$ we instead set $p_{i,1} = f_k(\bar{t}) + \delta$, overriding the earlier setting. Because we know that $f_k(\bar{t}) \geqslant \frac{1}{n(n-1)}$, we have that $\frac{1-f_k(\bar{t})-\delta}{n(n-1)-1} < \frac{1-\frac{1}{n(n-1)}}{n(n-1)-1} \leqslant \frac{1}{n(n-1)} \leqslant x_k$, therefore the new value of $\hat{p}_{i,1}$ is higher than the one it replaces. To set δ, choose some value that guarantees $\sum_{j=1}^{n(n-1)} \hat{p}_{i,j} < 1$ for all i. In particular, by the previous observation, it is sufficient to set δ so that

$$f_k(\bar{t}) + \delta + (n-2)\tfrac{1-f_k(\bar{t})-\delta}{n(n-1)-1} + (n-k-1)(n-1)\tfrac{1-f_k(\bar{t})}{n(n-1)-1} < 1. \tag{4.3}$$

To see that such a value of δ exists, note that Equation 4.3 is continuous in δ and takes value strictly less than 1 when $\delta = 0$:

$$\begin{aligned} f_k(\bar{t}) + (n-2)\tfrac{1-f_k(\bar{t})}{n(n-1)-1} + (n-k-1)(n-1)\tfrac{1-f_k(\bar{t})}{n(n-1)-1} &< f_k(\bar{t}) + n(n-2)\tfrac{1-f_k(\bar{t})}{n(n-1)-1} \\ &< f_k(\bar{t}) + 1 - f_k(\bar{t}) = 1, \end{aligned}$$

where we may assume $n \geqslant 3$ because the case of $n = 2$ has only a single phantom that is not f_0 or f_n. For every $j \in \{1, \ldots, n(n-1)\}$ for which $\hat{p}_{i,j}$ is not explicitly set greater than 0, we set it to 0.

For all i, we evenly distribute the remaining (positive) mass $1 - \sum_{j=1}^{n(n-1)} p_{i,j}$ evenly among $j \in \{n(n-1) + (i, i+1, i+n-k-2) \mod n\}$.

When $t = \bar{t}$, the generalised median on each project is $f_k(t) = f_k(\bar{t})$ for project 1, $\frac{1-f_k(\bar{t})-\delta}{n(n-1)-1}$ for all $j \in \{2, \ldots, n(n-1)$, and $x_{k+1}(\bar{t}) = 0$ for all $j \in \{n(n-1)+1, \ldots, n^2\}$. The sum of these generalised medians is $1-\delta$. Therefore t needs to increase to achieve normalisation. By the definition of $\bar{t}$, for any $t > \bar{t}$, we have that $f_{k+1}(t) > 0$, and therefore the generalised median on all projects $j \in \{n(n-1)+1, \ldots, n^2\}$ is greater than 0. It is therefore impossible for $f_k(t)$ to reach $f_k(\bar{t}) + \delta$, because then the sum of generalised medians would exceed 1. It is also impossible for $f_{k+1}(t)$ to reach $\frac{1-f_k(\bar{t})}{n(n-1)-1}$. If it does, then the generalised median on project 1 is at least $f_k(\bar{t})$, on $j \in \{2, \ldots, n(n-1)\}$ is $\frac{1-f_k(\bar{t})}{n(n-1)-1}$ and on $j \in \{n(n-1)+1, \ldots, n^2\}$ is strictly greater than 0. Therefore the aggregate is not normalised. To summarise, we are guaranteed that $\frac{1}{n(n-1)} \leqslant \mathcal{A}^{\mathcal{F}}(\mathbf{P})_1 < f_k(\bar{t}) + \delta$, $\frac{1-f_k(\bar{t})-\delta}{n(n-1)-1} \leqslant \mathcal{A}^{\mathcal{F}}(\mathbf{P})_j < \frac{1-f_k(\bar{t})}{n(n-1)-1}$ for all $j \in \{2, \ldots, n(n-1)\}$, and $\mathcal{A}^{\mathcal{F}}(\mathbf{P})_j > 0$ for all $j \in \{n(n+1)+1, \ldots, n^2\}$.

Now we can define a Pareto-improvement to $\mathcal{A}^{\mathcal{F}}(\mathbf{P})$ of the same form as previously. For some small enough ε, increase the aggregate by ε on projects $j \in \{1, \ldots, n(n-1)\}$, and decrease the aggregate by $\varepsilon(n-1)$ on projects $j \in \{n(n-1)+1, \ldots, n^2\}$. For voter 1, with $p_{1,1} = f_k(\bar{t}) + \delta$, the new aggregate is ε better than $\mathcal{A}^{\mathcal{F}}(\mathbf{P})$ on project 1, ε worse on projects $j \in \{2, \ldots, n-1\}$, with $p_{1,j} = \frac{1-f_k(\bar{t})-\delta}{n(n-1)-1}$, ε better on projects $j \in \{n, \ldots, n^2 - n - kn + k\}$, with $p_{i,j} = \frac{1-f_k(\bar{t})}{n(n-1)-1}$, and ε worse on projects $j \in \{n^2 - n - kn + k + 1, \ldots, n(n-1)\}$, with $p_{i,j} = 0$. On the final n projects, there are at least $k+1$ projects on which the aggregate moves $\varepsilon(n-1)$ closer to voter 1's report (on projects for which voter 1 reports $p_{i,j} = 0$), and at most $n-k-1$ projects for which the aggregate moves $\varepsilon(n-1)$ farther from voter 1's report. Summing these up, we get the aggregate has moved $1-(n-2)+(n-k-1)(n-1)-k(n-1)-(n-1)(n-k-1)+(n-1)(k+1) = 2$ towards $\mathbf{p_1}$.

For all other voters, the new aggregate is ε worse than $\mathcal{A}^{\mathcal{F}}(\mathbf{P})$ on projects $j \in \{(i-1)(n-1)+1, \ldots, n-1\}$, with $p_{i,j} = \frac{1-f_k(\bar{t})-\delta}{n(n-1)-1}$, and ε better on projects $j \in \{((i-1)(n-1)+n, \ldots, (i-1)(n-1)+n^2-n-kn+k) \mod n(n-1)\}$, with $p_{i,j} = \frac{1-f_k(\bar{t})}{n(n-1)-1}$, and ε worse on projects $j \in \{((i-1)(n-1)+n^2-n-kn+k+1, \ldots, (i-1)(n-1)+n(n-1)) \mod n(n-1)\}$, with $p_{i,j} = 0$. On the final n projects, there are at least $k+1$ projects on which the aggregate moves $\varepsilon(n-1)$ closer to voter i's report (on projects for which voter i reports $p_{i,j} = 0$), and at most $n-k-1$ projects for which the aggregate moves $\varepsilon(n-1)$ farther from voter i's report. Summing these up, we get that the new and old aggregates are equal ℓ_1 distances from $\mathbf{p_i}$.

Finally, our construction uses $m = n^2$ projects, but we can extend it to larger m by adding dummy projects that no voter puts any weight on. □

On profiles consisting of single-minded voters, $\mathcal{A}^{\mathcal{F}^*}$ selects a distribution that is also single-minded, following the plurality. Hence, it is not proportional, which gives the following corollary.

Corollary 4.17. *For $m \geqslant n^2$, no moving phantom mechanism is proportional and Pareto-optimal.*

4.6.2. Maximising Social Welfare

Having narrowed down the space of Pareto-optimal moving phantom mechanisms to at most one mechanism, let us examine the behaviour of $\mathcal{A}^{\mathcal{F}^*}$ with the assistance of Figure 4.2, which takes the same form as Figure 4.1. On every project, order the entries $\{p_{i,j}\}$ from largest to smallest. We denote the relabelled entries $\bar{p}_{1,j} \geqslant \cdots \geqslant \bar{p}_{n,j}$. At the snapshot of $\mathcal{F}^*$ for which $f_0(t) = \cdots = f_k(t) = 1$ and $f_{k+1}(t) = \cdots = f_n(t) = 0$, the generalised median selects the order

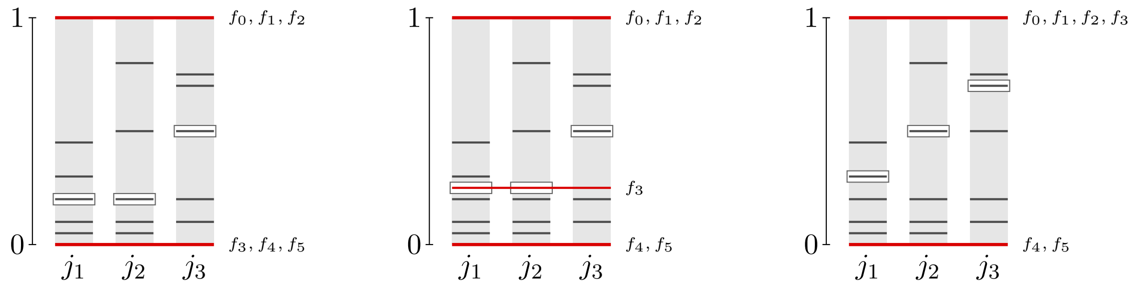


Figure 4.2.: Snapshots of the phantom system $\mathcal{F}^*$ with $t < t^*$ (left), $t = t^*$ (center), and $t > t^*$ (right) on an instance with $n = 5$, $m = 3$.

statistic $\bar{p}_{n-k,j}$ for all j. We see this in Figure 4.2 where, in the left image, $k = 2$ and the generalised median is the $n - k = 3$rd highest report on each project, and in the right image $k = 3$ and the $n - k = 2$nd highest reports are chosen.

We can think of $\mathcal{F}^*$ as partitioning the phantom "movie" into periods defined by which phantom is moving. Initially, all phantoms are at 0, and the generalised medians are 0 for each $j \in [m]$. Then phantom f_0 moves to 1, and the generalised medians are $\bar{p}_{n,j}$. As phantom f_k moves from 0 to 1, the generalised medians progress from $\bar{p}_{n-k+1,j}$ to $\bar{p}_{n-k,j}$, until all phantoms reach 1 and the generalised medians are uniformly 1. By (a discrete analogue of) the intermediate value theorem, there must exist some value I for which $\sum_{j\in[m]} \bar{p}_{I+1,j} \leqslant 1$ and $\sum_{j\in[m]} \bar{p}_{I,j} \geqslant 1$, and this transition is made during the period in which phantom $n - I$ is moving. In Figure 4.2, we have $I = 2$ because the sum of the third-highest entries is less than one (see the left image), while the sum of the second-highest entries is more than one (the right image).

Normalisation therefore occurs during the movement of phantom f_{n-I}, and the final value $\mathcal{A}^{\mathcal{F}^*}(\mathbf{P})_j$ lies in the interval $[\bar{p}_{I+1,j}, \bar{p}_{I,j}]$. If $f_{n-I}(t^*)$ lies in this interval, then $\mathcal{A}^{\mathcal{F}^*}(\mathbf{P})_j = f_{n-I}(t^*)$, otherwise $\mathcal{A}^{\mathcal{F}^*}(\mathbf{P})_j$ is equal to the endpoint of the interval closest to $f_{n-I}(t^*)$. This is depicted in the center image of Figure 4.2, where $f_3(t^*)$ lies between the second and third-highest reports on the first two projects, but below the third-highest report on the third project.

Finding the exact value of $f_{n-I}(t^*)$, and therefore the output $\mathcal{A}^{\mathcal{F}^*}(\mathbf{P})$, can be thought of as finding the "most equal" distribution, subject to interval constraints on each project. This problem has been studied before, and the (unique) value of $f_{n-I}(t^*)$ can be found in $O(m \log m)$ time by the Divvy algorithm of Gulati et al. [2012].

Given a profile $\mathbf{P}$, the *social cost* of an outcome $\mathbf{p}$ is $\sum_{i\in[n]} d(\hat{\mathbf{p}}_i, \mathbf{p})$, and the (utilitarian) social welfare of $\mathbf{p}$ is the negation of the social cost. In general, there may be multiple distributions that maximise social welfare. For example, if $m = 2$, one voter reports $(1, 0)$ and another reports $(0, 1)$, then all distributions have the same social cost of 2. As it turns out, any distribution that satisfies the upper and lower bound constraints of $\bar{p}_{I,j}$ and $\bar{p}_{I+1,j}$ maximises social welfare.

Lemma 4.18. *A distribution* $\mathbf{q}$ *maximises social welfare if and only if* $\bar{p}_{I+1,j} \leqslant q_j \leqslant \bar{p}_{I,j}$ *for all* j.

Proof. Let $\mathbf{q}$ be a distribution with $\bar{p}_{I+1,j} \leqslant q_j = \bar{p}_{I+1,j} + \varepsilon_j \leqslant \bar{p}_{I,j}$ for all j, with normalisation of $\mathbf{q}$ implying that $\sum_{j\in[m]} \varepsilon_j = 1 - \sum_{j\in[m]} \bar{p}_{I+1,j}$. Then the social cost of $\mathbf{q}$ is

$$\sum_{j\in[m]} \sum_{i\in[n]} |\bar{p}_{i,j} - q_j| = \sum_{j\in[m]} \left(\sum_{i\in[n]} |\bar{p}_{i,j} - \bar{p}_{I+1,j}| + \sum_{i\geqslant I+1} \varepsilon_j - \sum_{i\leqslant I} \varepsilon_j \right)$$

$$= \sum_{j\in[m]} \sum_{i\in[n]} |\bar{p}_{i,j} - \bar{p}_{I+1,j}| + (n - 2I) \left(1 - \sum_{j\in[m]} \bar{p}_{I+1,j} \right)$$

Because this expression does not depend on ε_j, all such distributions $\mathbf{q}$ have the same social cost.

We now show that this distance is minimal. Let $\mathbf{q}$ be a distribution that does not satisfy $\bar{p}_{I+1,j} \leqslant q_j \leqslant \bar{p}_{I,j}$ for some j. Suppose $q_j > \bar{p}_{I,j}$ (the case where $q_j < \bar{p}_{I+1,j}$ can be handled similarly). By the definition of I, there must exist some project j' for which $q_{j'} < \bar{p}_{I,j'}$. Now, consider the distribution $\mathbf{q}'$ defined by $q'_j = q_j - \varepsilon > \bar{p}_{I,j}$ and $q'_{j'} = q_{j'} + \varepsilon < \bar{p}_{I,j'}$, with $\mathbf{q}'$ and $\mathbf{q}$ equal on all other coordinates. Compare $\mathbf{q}$ and $\mathbf{q}'$ in terms of ℓ_1 distance from the reports. They are indistinguishable on all projects other than j and j'. On project j, $\mathbf{q}'$ is ε closer than $\mathbf{q}$ to all entries $\bar{p}_{i,j}$ with $i \geqslant I$, and at most ε farther from all other entries. On project j', $\mathbf{q}'$ is ε closer than $\mathbf{q}$ to all entries $\bar{p}_{i,j'}$ with $i \leqslant I$, and at most ε farther from all other entries. Therefore, of the $2n$ entries on projects j and j', $\mathbf{q}'$ is ε closer than $\mathbf{q}$ to at least $n+1$ of them, and no more than ε farther than $\mathbf{q}$ from the other $n-1$. Therefore, $\mathbf{q}$ does not maximise social welfare. □

As a corollary of Lemma 4.18, we immediately obtain that $\mathcal{A}^{\mathcal{F}^*}$ maximises social welfare, and is therefore Pareto-optimal. Social-welfare-maximising mechanisms have been considered before; all that is needed is a suitable tiebreaking procedure to select a single distribution from the set of maximisers. Goel et al. [2019] suggest breaking ties by selecting the lexicographically largest welfare maximiser, but this is not neutral. We propose a different way to break ties, which is neutral: select the welfare maximiser $\mathbf{p}$ with largest Shannon entropy $-\sum_{j\in[m]} p_j \log p_j$. Because the set of welfare-maximisers is convex, and Shannon entropy is a convex function, existence and uniqueness of $\mathbf{p}$ is guaranteed.

Theorem 4.19. *For every profile* $\mathbf{P}$*,* $\mathcal{A}^{\mathcal{F}^*}$ *selects the entropy-maximising distribution among those that maximise social welfare.*

Proof. From the earlier discussion, we know that $\mathcal{A}^{\mathcal{F}^*}(\mathbf{P})_j = \operatorname{med}\{\bar{p}_{I+1,j}, \bar{p}_{I,j}, f_{n-I}(t^*)\} \in [\bar{p}_{I+1,j}, \bar{p}_{I,j}]$ for all $j \in [m]$. Therefore, it maximises social welfare.

It remains to show that, subject to these constraints, $\mathcal{A}^{\mathcal{F}^*}(\mathbf{P})$ maximises Shannon entropy. Consider any other distribution $\mathbf{q} \neq \mathcal{A}^{\mathcal{F}^*}(\mathbf{P})$ with $\bar{p}_{I+1,j} \leqslant q_j \leqslant \bar{p}_{I,j}$. Then there must exist a project j for which $\bar{p}_{I+1,j} \leqslant \mathcal{A}^{\mathcal{F}^*}(\mathbf{P})_j < q_j \leqslant \bar{p}_{I,j}$. Further, because $\mathcal{A}^{\mathcal{F}^*}(\mathbf{P})_j < \bar{p}_{I,j}$, it must be the case that $f_{n-I}(t^*) \leqslant \mathcal{A}^{\mathcal{F}^*}(\mathbf{P})_j = \operatorname{med}\{\bar{p}_{I+1,j}, \bar{p}_{I,j}, f_{n-I}(t^*)\}$. There must also be a project j' for which $\bar{p}_{I+1,j'} \leqslant q_{j'} < \mathcal{A}^{\mathcal{F}^*}(\mathbf{P})_{j'} \leqslant \bar{p}_{I,j'}$, with $\mathcal{A}^{\mathcal{F}^*}(\mathbf{P})_{j'} \leqslant f_{n-I}(t^*)$.

Putting these together, we have that $q_{j'} < f_{n-I}(t^*) < q_j$. We also know that $\bar{p}_{I+1,j} < q_j$ and $q_{j'} < \bar{p}_{I,j'}$. Therefore, adjusting q_j to $q_j - \varepsilon$ and $q_{j'}$ to $q_{j'} + \varepsilon$, for ε small enough that none of the above strict inequalities are violated, both (1) decreases $|q_j - q_{j'}|$, which it is easy to check increases Shannon entropy, and (2) respects social-welfare maximisation. Therefore, $\mathbf{q}$ is not the unique entropy-maximising distribution among social welfare maximisers, so $\mathcal{A}^{\mathcal{F}^*}$ is. □

4.7. Minimum Spending Requirements

The model of this chapter assumes that projects are feasible no matter what fraction of the budget is spent on them. However, many projects have some fixed costs, and so they only make sense when spending exceeds a certain minimum level. In this section, we will show that with these additional constraints it can become impossible to satisfy strategyproofness.

We study a specific setting to illustrate the problem. Suppose there are $m = 3$ projects, and we impose the following constraint: a distribution $\mathbf{p}$ is *feasible* if $p_x > 0$ implies $p_x \geqslant \frac{1}{3}$. We only allow reports of ideal distributions which are themselves feasible (this restriction strengthens the following negative result).

Recall that a mechanism $\mathcal{A}$ is *range-respecting* if $\min_{i\in[n]} \hat{p}_{i,j} \leqslant \mathcal{A}(\mathbf{P})_j \leqslant \max_{i\in[n]} \hat{p}_{i,j}$ for all $j \in [m]$. We prove that there is no anonymous, range-respecting, truthful mechanism which selects feasible distributions, for $n = 2$ voters. Suppose for a contradiction that $\mathcal{A}$ is an aggregator satisfying these conditions.

We will mainly reason about profiles where all reported fractions are $\frac{1}{2}$. An important observation: The output at such a profile must be one of the voter reports. For example, $\mathcal{A}((0, 0.5, 0.5), (0.5, 0, 0.5))$ is either $(0, 0.5, 0.5)$ or $(0.5, 0, 0.5)$. To see this, note that the third value needs to be exactly 0.5 for $\mathcal{A}$ to be range-respecting, and at most one of the remaining values can be positive by the minimum spending assumption.

Lemma 4.20. *We have*

(a) $\mathcal{A}((0, 0.5, 0.5), (0.5, 0, 0.5)) = (0, 0.5, 0.5)$ *iff* $\mathcal{A}((0, 0.5, 0.5), (0.5, 0.5, 0)) = (0, 0.5, 0.5)$*;*

(b) $\mathcal{A}((0.5, 0, 0.5), (0, 0.5, 0.5)) = (0.5, 0, 0.5)$ *iff* $\mathcal{A}((0.5, 0, 0.5), (0.5, 0.5, 0)) = (0.5, 0, 0.5)$*;*

(c) $\mathcal{A}((0.5, 0.5, 0), (0, 0.5, 0.5)) = (0.5, 0.5, 0)$ *iff* $\mathcal{A}((0.5, 0.5, 0), (0.5, 0, 0.5)) = (0.5, 0.5, 0)$.

Proof. We only do (a), the other statements are symmetric. Suppose the statement is false, and exactly one of the conditions is satisfied. Suppose (without loss of generality) that the former is satisfied and the latter is not, so

$$\mathcal{A}((0, 0.5, 0.5), (0.5, 0, 0.5)) = (0, 0.5, 0.5) \tag{4.4}$$
$$\mathcal{A}((0, 0.5, 0.5), (0.5, 0.5, 0)) = (0.5, 0.5, 0). \tag{4.5}$$

In situation (4.4), the second voter is unhappy, and misreports $(\frac{2}{3}, 0, \frac{1}{3})$. Write

$$\mathcal{A}((0, 0.5, 0.5), (\tfrac{2}{3}, 0, \tfrac{1}{3})) = (a, 0.5 - b, 0.5 - c).$$

for some $a, b, c \geqslant 0$ (non-negativity comes from range-respecting). By truthfulness, $c \geqslant a + b$. Since the output must be normalized, $a = b + c$. Combining these, $0 \geqslant 2b$, so $b \leqslant 0$. Hence $b = 0$, so $a = c$. From the minimum spending requirement, either $a = 0$ or $a \geqslant \frac{1}{3}$. The latter case is impossible since then $c \geqslant \frac{1}{3}$ so that $0.5 - c \leqslant \frac{1}{6}$, which contradicts that $\mathcal{A}$ is range-respecting. Thus $a = b = c = 0$, so

$$\mathcal{A}((0, 0.5, 0.5), (\tfrac{2}{3}, 0, \tfrac{1}{3})) = (0, 0.5, 0.5).$$

The second voter is at distance $\frac{2}{3} + \frac{1}{2} + \frac{1}{6} = \frac{8}{6}$. If the second voter manipulates and reports $(0.5, 0.5, 0)$, then by (4.5), the output is only at distance $\frac{1}{6} + \frac{1}{2} + \frac{1}{6} = \frac{5}{6} < \frac{8}{6}$, a successful manipulation, contradicting truthfulness of $\mathcal{A}$. ☐

Now we apply the lemma several times, while implicitly using anonymity and our initial observation, to obtain a contradiction.

Start by supposing, without loss of generality since we could reorder voters and projects, that $\mathcal{A}((0, 0.5, 0.5), (0.5, 0, 0.5)) = (0, 0.5, 0.5)$. By (a), we have $\mathcal{A}((0, 0.5, 0.5), (0.5, 0.5, 0)) = (0, 0.5, 0.5)$. This violates the left-hand part of (c), and thus its right-hand part is also violated, so $\mathcal{A}((0.5, 0.5, 0), (0.5, 0, 0.5)) = (0.5, 0, 0.5)$. This satisfies the right-hand part of (b), so its left-hand part is also satisfied, so $\mathcal{A}((0.5, 0, 0.5), (0, 0.5, 0.5)) = (0.5, 0, 0.5)$. This contradicts our initial assumption.

4.8. Conclusion

We considered the problem of aggregating budget proposals for participatory budgeting. Inspired by the generalised median mechanisms of Moulin [1980], we introduced the broad class of moving phantom mechanisms and proved that all mechanisms in this class are strategyproof under ℓ_1 voter preferences. We analysed two moving phantom mechanisms in detail: one that maximises social welfare while violating proportionality, and another that satisfies proportionality while violating Pareto-optimality.

We have implemented both mechanisms. Some preliminary simulation results suggest that even when voters are not single-minded, independent markets is "more proportional" than $\mathcal{A}^{\mathcal{F}^*}$, in the sense that it better reflects all voters' opinions and not just that of the majority. However, independent markets also has a tendency to shift the aggregate towards the uniform distribution, relative to what we might intuitively expect. This deserves more investigation.

There are many other moving phantom mechanisms that we have not considered. It would be interesting to investigate what other properties can be achieved by other phantom systems. And finally, as we mentioned earlier, when there are only two outcomes, we know that all (anonymous, neutral, and continuous) strategyproof budget aggregation mechanisms can be represented as moving phantom mechanisms. It remains an open question whether this continues to hold when the number of outcomes m is more than 2.

5. Aggregating Approval Preferences

We consider the problem of dividing a perfectly divisible common budget among several uses, employing the user-friendly input format of approval ballots: voters can indicate for each project whether they approve it or not. We give a short overview of several natural aggregation rules for this input type. We then prove a conjecture of Bogomolnaia, Moulin, and Stong [2005], saying that there is no aggregation mechanism that is efficient, strategyproof, and satisfies an extremely mild fairness condition. This axioms in the impossibility are independent, so dropping any of the three axioms makes the impossibility disappear.

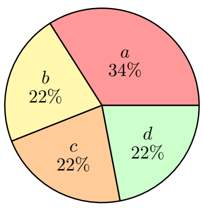
5.1. Introduction

In this chapter, we consider the same setting as in the previous chapter: a common divisible budget needs to be divided among several projects, and each project can receive any fraction of the budget. However, we consider a different utility model and hence a different input format: we allow voters to indicate, for each project, whether they approve it or disapprove it. Given a budget division, we take the voter's utility to be equal to the fraction of the budget spent on approved projects.

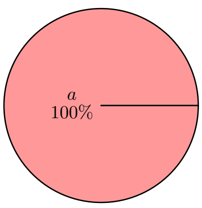
This model was proposed by Bogomolnaia et al. [2002, 2005]. Let us introduce some of their aggregation rules and discuss their properties on an example. Suppose that there are four projects, $A = \{a, b, c, d\}$ and five voters $N = \{1, 2, 3, 4, 5\}$ with the profile

$$P = (\{a\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}).$$

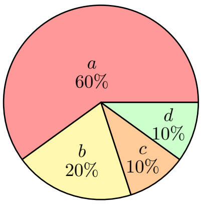
Thus, for example, voter 2 approves projects a and c, but disapproves projects b and d. Write A_i for the approval set of voter i. The outcome space is the set $\Delta(A)$ of distributions $p : A \to [0, 1]$ with $\sum_{a \in A} p_a = 1$. For a voter $i \in N$ and a given distribution p, the utility of i is $u_i(p) = \sum_{a \in A_i} p_a$.



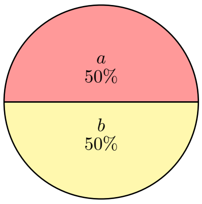
One possible way of obtaining a distribution is to take the spending on a project to be proportional to its approval score. The approval score of a project is the number of voters who approve it. In our example, the approval scores are 3, 2, 2, 2, and the corresponding distribution is $(\frac{3}{9}, \frac{2}{9}, \frac{2}{9}, \frac{2}{9})$. A problem with this approach is that it will spend a positive fraction on every project that is approved by at least one voter, and this includes dominated projects. In our example, no project is dominated by another, but still this distribution is inefficient: every voter weakly prefers $(\frac{5}{9}, \frac{4}{9}, 0, 0)$, and voter 1 strictly prefers it.



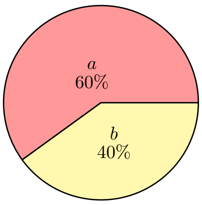
A standard way to achieve Pareto efficiency is to maximise some type of social welfare. For example, we could take the distribution p for which $\sum_{i \in N} u_i(p)$ is maximised. This is the *utilitarian rule*. On our example, the optimum is the distribution $(1, 0, 0, 0)$ which spends the entire budget on the project a with the highest approval score, that is, on the approval winner. To see this, consider any distribution which spends a positive amount on a project which is not an approval winner; then redistributing this spending towards an approval winner increases $\sum_{i \in N} u_i(p)$. While this choice is certainly Pareto-efficient, it seems unfair, since it gives utility 0 to voters 4 and 5, who form 40% of the electorate.



To achieve fairness while keeping some of the utilitarian spirit, Duddy [2015] introduces the *conditional utilitarian rule*, further discussed by Aziz et al. [2019a] and Brandl et al. [2019b]. Each voter is given an equal share of the budget (here 20%) and can decide what to do with it. Each voter i spends the share only on approved projects, but chooses those projects in A_i which have the highest approval score. In our example, voter 1 spends 20% on a since this is the only approved project. Voters 2 and 3 also spend only on a, since the approval score of a is higher than the score of b or of c. For voter 4 and 5, their two approved projects have the same approval score, and so we let them put 10% on each. Summing up, we obtain $(0.6, 0.2, 0.1, 0.1)$. By design, no voter's interests are ignored, but the resulting distribution can be inefficient. Here, every voter weakly prefers $(0.7, 0.3, 0, 0)$, and voter 1 strictly prefers it.



Another way to ensure that no voter is ignored is following the Rawlsian principle of maximising the welfare of the worst-off agent. This would correspond to choosing a distribution p such that the egalitarian welfare $\min_{i \in N} u_i(p)$ is maximised. In our example, the distribution $(0.5, 0.5, 0, 0)$ gives every voter a utility of at least 50%, and it is easy to see that this is optimal. By standard arguments, one can show that there is always a Pareto-efficient distribution that maximises egalitarian welfare (by using the leximin criterion).



A possible criticism of the egalitarian rule is that it ignores the relative sizes of different groups. For example, if we added a hundred voters to P who all only approve a, the egalitarian rule would not change its output. One rule that is often seen as a compromise is to maximise the Nash product $\prod_{i \in N} u_i(p)$. This rule does take into account how frequently each vote occurs in a profile, but it retains some of the spirit of egalitarianism: if some voter obtains a low utility in an outcome this substantially reduces the Nash product, so it prioritises improving the utility of badly represented voters. In particular, unlike the utilitarian rule, it will never return a distribution in which a voter obtains utility 0, since that would give a Nash product of 0. The Nash rule always returns a Pareto optimum, and in our case the output is $(0.6, 0.4, 0, 0)$.

One can prove that the first three rules we mentioned (proportional to approval scores, utilitarian rule, conditional utilitarian rule) are strategyproof: an agent with dichotomous 0/1 preferences cannot misreport to obtain higher utility. This mirrors the result that single-winner Approval Voting is strategyproof [Brams and Fishburn, 1983]. The egalitarian rule and the Nash product are not strategyproof: If voter 2 reports the approval set $\{c\}$ instead of $\{a, c\}$, then the egalitarian (leximin) rule returns $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0)$ and the Nash rule returns $(\frac{2}{6}, \frac{1}{6}, \frac{2}{6}, \frac{1}{6})$. In both cases, the fraction spent on a and c together has strictly increased from the previous output.

A rule satisfies *positive share* if for all profiles, in the output distribution p, we have $u_i(p) > 0$ for every $i \in N$. Thus, we never spend 100% of the budget on projects a voter disapproves. The utilitarian rule violates positive share, even though it is an extremely weak property. Indeed, positive share does not seem strong enough to adequately capture fairness. However, in this chapter we are after a negative result, and thus a weak notion makes for a stronger argument.

Let us summarise our discussion so far in a table that indicates, for each of the rules we mentioned, whether it satisfies Pareto efficiency, positive share, and strategyproofness.

	approval scores	utilitarian	conditional utilitarian	egalitarian	Nash
Pareto-efficient	×	✓	×	✓	✓
positive share	✓	×	✓	✓	✓
strategyproof	✓	✓	✓	×	×

Suspiciously, no column features three checkmarks. Is there some other rule which satisfies all three of these axioms?

5.2. Impossibility Theorem

In 2005, Bogomolnaia, Moulin, and Stong [2005] conjectured that, like the utilitarian rule, all efficient and strategyproof mechanisms will fail positive share – and hence many other desirable properties which are stronger, such as the individual fair share property [Bogomolnaia et al., 2005] or the core [Aziz et al., 2019a]. They "submit as a challenging conjecture the following statement: there is no strategyproof and ex ante efficient mechanism guaranteeing positive shares." Bogomolnaia et al. were able to prove impossibility theorems of this type only when substituting much stronger versions of strategyproofness or of positive share. Still, their proofs were rather involved, and one of them required that there are $m \geqslant 17$ projects.[1] As to whether a mechanism satisfying the original conditions exists, they left it as "a challenging open question to which we suspect the answer is negative when $[m]$ and $[n]$ are large enough."

Here, we confirm Bogomolnaia et al.'s conjecture.

Theorem 5.1. *No mechanism satisfies efficiency, strategyproofness, and positive share when $m \geqslant 4$ and $n \geqslant 6$.*

To our surprise, Bogomolnaia et al.'s suspicion that an impossibility would require a large number of voters and projects turned out to be false.

We proved Theorem 5.1 using the computer-aided technique based on SAT solving that we saw in Part I. On first sight, our problem has a continuous flavour, since the rules in this context return real-valued distributions. This suggests an encoding into integer linear programming like we did in Section 4.7, or into SMT, which has previously been used to prove an impossibility theorem in the formally equivalent setting of probabilistic social choice [Brandl et al., 2018]. A drawback of these continuous methods is that they can (presently) only handle comparatively small instances. Solving times tend to become prohibitive once we search for an impossibility for a domain of more than a few thousand profiles. Discrete encodings of social choice problems into SAT can often be solved for hundreds of thousands of profiles.

Our problem can be discretised by only considering the *support* of the distribution returned by our mechanism; the support of a distribution is the set of projects on which a positive fraction is spent. For 4 projects, there are only $2^4 - 1 = 15$ possible supports per profile (compared to infinitely many distributions). Clearly, for every rule that assigns an output distribution to every input profile, there is an induced function that assigns a support to every input profile. Note that the positive share axiom only refers to the support. Less obviously, Aziz, Brandl, and Brandt [2014] have proved that whether a distribution is efficient or not depends only on its support. The only remaining axiom is strategyproofness, which depends on the precise distributions returned by the mechanism. However, it turns out that impossibility still holds when only considering particularly clear-cut manipulations. In the encoding, we *only* consider manipulations in which the manipulator enforces a distribution in which the *entire* budget is distributed across the manipulator's approved projects, so that by manipulating the agent obtains the maximum possible utility of 1. If we obtain an impossibility with these axioms for support-selecting rules, this implies the same impossibility for rules selecting a distribution.

Even after discretising, the formulas involved are very big, and further reduction techniques are needed. There are $15^6 \approx 11$ million different profiles with $n = 6$ and $m = 4$, and we need to use 15 variables for each profile (one for each support), giving 170 million variables in total. It is much easier to obtain a result when we impose anonymity and neutrality, which was also done by Bogomolnaia et al. [2005]. A mechanism is *anonymous* if it is invariant under renaming agents, and it is *neutral* if a permutation of the projects induces the same permutation in the mechanism's output.

[1] Duddy [2015] proved a related impossibility using a group fairness notion. His result, like Theorems 5.1 and 5.2, works for $m \geqslant 4$ projects, and he shows that this bound is tight.

When we consider anonymous and neutral mechanisms, the number of essentially different profiles reduces to 2197. In fact, with these extra axioms, the impossibility holds even for $n = 5$, for which there are only 736 essentially different profiles. Solving the resulting formula is almost instantaneous with a modern SAT solver. After extracting a minimal unsatisfiable set, we were astonished to find that it only referred to two different profiles, giving a short and elegant proof.

Theorem 5.2. *No anonymous and neutral rule satisfies efficiency, strategyproofness, and positive share when* $m \geqslant 4$ *and* $n \geqslant 5$.

Proof. We prove the incompatibility for $m = 4$ and $n = 5$. The proof can be adapted to larger values by adding agents approving all projects or by adding projects which no-one approves.

Assume there is a strategyproof mechanism satisfying efficiency and positive share. Consider the following profile:

$$Q = (\{a\}, \{a, c\}, \{a, d\}, \{b, c\}, \{a, b\}).$$

Let q be the distribution returned by the rule when given Q as input. Because the mechanism is anonymous and neutral, since b and c are symmetric, we must have $q_b = q_c$, and this value must be positive by positive share for voter 4. It follows that $u_5(q) < 1$ because a positive amount is spent on project c, which voter 5 does not approve.

Suppose voter 5 reports the approval set $\{b, d\}$ instead of $\{a, b\}$. The resulting profile is

$$P = (\{a\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}),$$

which is the same profile P we looked at before. Let p be the distribution returned by the rule when given P as input. Suppose first that both p_c and p_d are positive, say $p_c \geqslant \varepsilon$ and $p_d \geqslant \varepsilon$ for some $\varepsilon > 0$. Then consider the distribution p' with

$$p'_a = p_a + \varepsilon, \quad p'_b = p_b + \varepsilon, \quad p'_c = p_c - \varepsilon, \quad p'_d = p_d - \varepsilon.$$

One can check that, in profile P, $u_i(p') \geqslant u_i(p)$ for all $i \in N$, and $u_1(p') > u_1(p)$. Thus, p is not Pareto-efficient, which contradicts efficiency of the rule. Hence either $p_c = 0$ or $p_d = 0$. Now c and d are symmetric projects in P, and thus we must have $p_c = p_d$ by anonymity and neutrality of the rule. Thus $p_c = p_d = 0$. So the entire budget is split between projects a and b, and so $u_5(p) = 1$, where we take voter 5's utility as reported in profile Q, and in particular $u_5(p) > u_5(q)$.

Hence, voter 5 has successfully manipulated, which contradicts strategyproofness. □

This short proof relies heavily on symmetry arguments. However, the SAT solver indicates that the theorem remains true without imposing anonymity and neutrality, confirming a second conjecture of Bogomolnaia et al. [2005]. However, without anonymity and neutrality, proofs become much more complicated and consider potential manipulations between hundreds of profiles. Thus, we do not include a proof of Theorem 5.1 here. To obtain the result, one can first find a proof (via minimal unsatisfiable subsets) that uses anonymity but not neutrality. Then, run the SAT solver on the (manageably small) domain of profiles obtained by permuting the profiles in the minimal unsatisfiable subset in all possible ways.

5.3. Subset Manipulations

Rules satisfying notions such as positive share aim for an outcome that makes every agent reasonably happy. There is an obvious strategy for manipulators to try to exploit this tendency: they can pretend to be less happy than they are. In our setting, this would correspond to approving fewer projects, i.e. to manipulate by reporting a subset of the truthful approval set.[2] We can show, by a proof similar to the one above, that every efficient mechanism that satisfies positive share can be manipulated using this technique. The proof uses anonymity and neutrality and, in contrast to Theorem 5.1, we do not know whether this can be dropped.

Theorem 5.3. *Every anonymous and neutral rule which satisfies efficiency and positive share can be manipulated by an agent reporting a subset of their truthful approval set, when $m \geqslant 5$ and $n \geqslant 5$.*

Proof. We prove the incompatibility for $m = 5$ and $n = 5$, and the proof can be adapted to larger values as before.

Assume that there is a rule f satisfying anonymity, neutrality, efficiency, and positive share. Now consider the profile

$$P = (\{a\}, \{a, b, c\}, \{a, b, d\}, \{a, c, e\}, \{d, e\}).$$

Let p be the distribution returned by the mechanism when given profile P. Since f is efficient, we must have $p_b = p_c = 0$, because otherwise a Pareto improvement can be obtained by redistributing spending from either of these projects to a. Since the profile is symmetric under the permutation $\sigma = (b\ c)(d\ e)$, we must have $p_b = p_c$ and $p_d = p_e$ because f is anonymous and neutral. By positive share for voter 5, we must have $p_d = p_e > 0$. It follows that $u_4(p) < C$ because a positive amount is spent on project d, which voter 4 does not approve.

Now, suppose that voter 4 pretends not to approve a, so we get the profile Q:

$$Q = (\{a\}, \{a, b, c\}, \{a, b, d\}, \{c, e\}, \{d, e\}).$$

Let q be the distribution now returned by the mechanism when given profile Q. Again, by efficiency, we must have $q_b = 0$ since we can otherwise redistribute resources from b to a to get a Pareto improvement. Next, suppose that both q_c and q_d are positive, say $q_c \geqslant \varepsilon$ and $q_d \geqslant \varepsilon$ for some $\varepsilon > 0$. Then q is Pareto dominated by the distribution q' defined as

$$q'_a = q_a + \varepsilon, \quad q'_b = q_b, \quad q'_c = q_c - \varepsilon, \quad q'_d = q_d - \varepsilon, \quad q'_e = q_e + \varepsilon.$$

This contradicts efficiency of f, so either $q_c = 0$ or $q_d = 0$. Since projects c and d are symmetric in Q, we must have $q_c = q_d = 0$. Hence, q distributes the entire budget between projects a and e, and so $u_4(q) = 1$.

Thus, voter 4 has successfully manipulated f by reporting a subset of the true approval set. □

Theorems 5.1 and 5.3 are both strong impossibilities, because of the weak versions of strategyproofness and fairness used in their proofs. However, the table at the end of Section 5.1 shows that the impossibility disappears once we drop any of the axioms (except anonymity and neutrality). For example, if we drop efficiency, we obtain the conditional utilitarian rule, and if we drop strategyproofness, we obtain the Nash rule. Both of these rules are arguably attractive.

[2] We will also study this notion of "subset-strategyproofness" in the context of proportional multiwinner elections in Chapter 7. The corresponding notion of *superset*-strategyproofness has been studied by Aziz, Bogomolnaia, and Moulin [2019a], who found that the conditional utilitarian and the egalitarian rules satisfy it, while the Nash rule fails it.

6. Aggregating Ranking Preferences

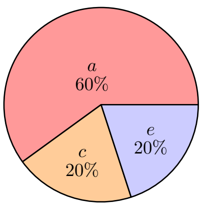
A public divisible resource is to be divided among projects. We study rules that decide on a distribution of the budget when voters have ordinal preference rankings over projects. We introduce a family of rules for portioning, inspired by positional scoring rules. Rules in this family are given by a scoring vector (such as plurality or Borda) associating a positive value with each rank in a vote, and an aggregation function such as leximin or the Nash product. Our family contains well-studied rules, but most are new. We discuss computational and normative properties of our rules. We focus on fairness, and introduce the SD-core, a group fairness notion. Our Nash rules are in the SD-core, and the leximin rules satisfy individual fairness properties. Both are Pareto-efficient.

6.1. Introduction

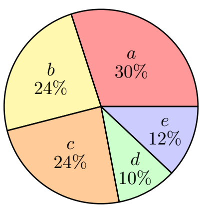
In this chapter, we ask voters to report their preferences over projects as *rankings*, the most common format considered in social choice. If a project is ranked more highly, the voter thinks it is more worthwhile and should receive a larger fraction of the budget.

The space of sensible aggregation rules for this input format is large. Let us illustrate some important design considerations by an example, in a similar spirit as the running example of Chapter 5.

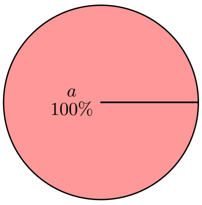
An Example A family is planning a road trip by car. The family members have different musical tastes; they need to decide which type of music to play for how long. The genres under consideration are a, b, c, d, e. The three children all think $a \succ b \succ c \succ d \succ e$; mother thinks $e \succ b \succ c \succ d \succ a$; and father thinks $c \succ a \succ e \succ d \succ b$.



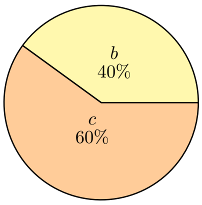
One simple way to split the time is to allocate each person the same share of time (20%) and let them decide what music to play, as a temporary dictator. During their time, each person plays their favourite music. To the social choice theorist, this rule sounds familiar: it is formally identical to *Random Dictatorship*, whose output is usually seen not as a division of a budget, but as a probability distribution. Indeed, any probabilistic social choice function can be repurposed to divide budgets; but these are often not attractive for portioning since many of them were designed as tie-breaking devices.



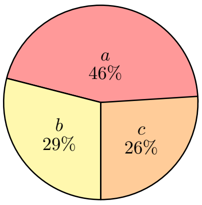
The output of random dictatorship can be a good choice, especially if our family strongly prefers their top choice to any other music. But it is also plausible that mother and the children agree that b is good common ground. Random Dictatorship, using plurality scores, ignores this. Instead, we could impute *Borda scores* on our family: for example, the children give 4 ‘utility’ points to a, 3 to b, 2 to c, 1 to d, and 0 to e. *Proportional Borda* then allocates time in proportion to the total Borda score of the genres. This leads to a significant time share for b. On the other hand, the family now also listens to d, which is dominated: everyone agrees that c is better than d! So Proportional Borda is inefficient.



To restore efficiency, it makes sense to maximise a notion of social welfare. Suppose the utility enjoyed by a family member is the weighted average of the Borda scores of the music played on the trip, where the weights come from the fraction of time spent on each genre. *Utilitarian Borda* then picks the distribution where the sum of utilities is greatest. In our example, we listen to a during 100% of the time. While this is Pareto-efficient, it is unfair to mother, who only gets to listen to her least-preferred style. In fact, many rules suffer from this phenomenon of completely overriding some voters' preferences: For example, the 'maximal lotteries' rule also only plays a since it is the Condorcet winner.



To avoid frustration during the trip, we may take a more egalitarian approach, and try to give each family member a significant share. *Borda-Egalitarian* picks the distribution maximising the utility of the worst-off passenger. In our example, we can give every passenger an average Borda-utility of 2.4. In general, this approach does not give a Pareto-efficient outcome, but we can ensure Pareto-efficiency by using leximin maximisation: once we have maximised the utility of the worst-off passenger, we then maximise the utility of the second-worst-off person and so on. On this example, using leximin gives the same outcome as egalitarian.



We can also maximise *Nash social welfare*, the product of utilities. This is often seen as a compromise between maximising utilitarian and egalitarian welfare notions. While egalitarian rules perform well when we wish to be fair to each *individual*, Nash rules tend to be fair to *groups*. In our example, the children form a large group, and Borda-Nash plays a almost half the time. If there were more children with the same preferences in the car, Borda-Nash would increase the time share of a. In contrast, Borda-Leximin avoids playing a to benefit the mother, and the output of egalitarian rules does not change with the number of children. Depending on the context, either of these behaviours might be more appropriate.

Our Contributions We introduce a class of aggregation rules called *positional social decision schemes*. Rules in this class first convert each input ranking into scores for the projects, using a scheme such as plurality or Borda scores. Then, they select a distribution of the budget maximising social welfare given those scores, where different notions of welfare can be used; classically, we consider utilitarian, egalitarian (leximin), and Nash welfare. Our class contains known rules such as random dictatorship, but most have not been studied.

We begin by noting basic properties of the rules in our class, giving closed forms and equivalent definitions in some cases. We also show that the rules in this class can be calculated or approximated in polynomial time. For rules based on Nash welfare, we show that their output can involve irrational percentages; we prove that those rules are guaranteed to be rational if the scoring vector used is plurality or veto, but that no other scoring vector guarantees rational output.

We then formalise intuitive notions of fairness in the budgeting context. The axioms we propose require that no individual is ignored by the procedure, in the sense of having none of the budget allocated to favoured causes. We also give some group fairness notions. Our strongest axiom is the *SD-core* which, roughly, requires that a group of α% of the voters can control what happens with α% of the budget. We show that the rules in our class based on Nash welfare satisfy the SD-core, while the egalitarian rules satisfy the individual fairness notions.

We close by studying the performance of our rules on standard social choice properties, such as Pareto-efficiency, strategyproofness, and monotonicity. While the first is usually satisfied by our rules, the latter two are mostly failed.

Related Work In Chapter 5, we saw related work when voters have dichotomous (approval) preferences. The paper by Aziz et al. [2019a] considers some rules based on welfare maximisation, and it introduces new fairness axioms (including a core notion) which are related to the fairness axioms we discuss in Section 6.4.

Fain et al. [2016] study the divisible budgeting problem in a *cardinal* model which allows agents to give a full utility function over distributions (which may also feature decreasing returns). They study the core and connect it to the Lindahl equilibrium from the study of public goods, and prove that a core solution always exists. For a broad class of utility functions, they show that a core solution can be found in polynomial time by solving a suitable convex program. They also use differential privacy to design a mechanism for this setting which satisfies approximate versions of efficiency, truthfulness, and the core.

With rankings as input, this setting has been studied in the formally isomorphic guise of *probabilistic social choice* [see Brandt, 2019 for a recent survey]. In this literature, the outcome distribution is interpreted as a random device, which is used to eventually implement a single outcome. This makes notions of fairness and proportionality less relevant, and it is seen as desirable for a rule to randomise as little as possible. For example, the *maximal lotteries* rule [Kreweras, 1965, Brandl et al., 2016], while attractive according to consistency axioms, spends the entire budget on the Condorcet winner if it exists. This is usually undesirable in a budgeting context. On the other hand, results like Gibbard's [1977] random dictatorship theorem are as interesting in our setting as they are within probabilistic social choice. Some papers on probabilistic social choice also discuss fairness concerns [see, e.g., Aziz et al., 2018b; Aziz and Stursberg, 2014].

The literature on cake-cutting and item allocation is mostly *unrelated* to our problem: in those settings, goods are allocated to specific agents for their exclusive use. In our setting, resources are spent on projects which can be enjoyed by all agents. On a technical level, the idea of scoring followed by aggregation has been explored in fair division [Brams and King, 2005, Darmann and Schauer, 2015, Baumeister et al., 2017].

6.2. Positional Social Decision Schemes

Preliminaries Let $A = \{x_1, \dots, x_m\}$ be a set of projects and $N = \{1, \dots, n\}$ be a set of voters. Let $A!$ be the set of linear orders over A. For $\succ \in A!$, the *rank* of project x_j is $r(\succ, x_j) = 1 + |\{x_k \in A : x_k \succ x_j\}|$. A *profile* $P = (\succ_1, \dots, \succ_n) \in A!^n$ is a collection of linear orders, one for each voter. We write abc as shorthand for $a \succ b \succ c$. Let $\Delta(A) = \{p : A \to [0,1] : \sum_{x \in A} p_x = 1\}$ be the set of (probability) distributions over A. We use notations such as $\frac{1}{2}x_1 + \frac{1}{2}x_2$ to specify a distribution, and just x_j for the degenerate distribution with $p_{x_j} = 1$. We say that $z : A \to [0,1]$ is a *partial distribution* if $\sum_{x \in A} z_x \leqslant 1$. A *social decision scheme* (SDS) is a function F assigning to each $P \in A!^n$ a nonempty subset of $\Delta(A)$, usually a singleton.

Positional SDSes A *scoring vector* for m projects is a vector $\mathbf{s} = (s_1, \dots, s_m)$ of numbers with $s_1 \geqslant s_2 \dots \geqslant s_m$ and $s_1 > s_m$. We usually assume $s_m = 0$. A scoring vector $\mathbf{s}$ is *strictly decreasing* if $s_j > s_{j+1}$ for all $j < m$. The Borda vector is $\mathbf{bor} = (m-1, m-2, \dots, 0)$; the plurality vector is $\mathbf{plu} = (1, 0, \dots, 0)$; the veto vector is $\mathbf{vet} = (1, \dots, 1, 0)$.

For a fixed profile P, we write $\mathbf{s}[i,j] = s_{r(\succ_i, x_j)}$ for the $\mathbf{s}$-score that voter $i \in N$ assigns to project $x_j \in A$. These scores can be lifted to distributions in a natural way; the $\mathbf{s}$-score of $p \in \Delta(A)$ for i is $\mathbf{s}[i,p] = \sum_{j=1}^{m} p_j \mathbf{s}[i,j]$. Finally, define the utility vector $\mathbf{s}[p] = (\mathbf{s}[1,p], \dots, \mathbf{s}[n,p])$.

A *welfare ordering* is a weak order $\geqslant_W$ ordering utility vectors $(\alpha_1, \dots, \alpha_n) \in \mathbb{R}^n_{\geqslant 0}$. The main examples are utilitarianism which orders vectors by their sum, egalitarianism which uses the minimum, the Nash product which uses multiplication, and leximin which sorts the components of the utility vector and then orders sorted vectors lexicographically.

By combining a scoring vector and a welfare ordering, we can define a positional social decision scheme.

Definition 6.1. For scoring vector $\mathbf{s}$ and a welfare ordering $\geqslant_W$, define the social decision scheme $F_{\mathbf{s},\geqslant_W}$ so that for all P,

$$F_{\mathbf{s},\geqslant_W}(P) = \{p \in \Delta(A) : \mathbf{s}[p] \geqslant_W \mathbf{s}[q] \text{ for all } q \in \Delta(A)\}.$$

For the specific $\geqslant_W$ mentioned, we usually call these rules $\mathbf{s}$-utilitarianism, $\mathbf{s}$-egalitarianism, $\mathbf{s}$-leximin, and $\mathbf{s}$-Nash.

Example 6.2. Consider the profile $P = (ab, ab, ba)$ over two projects, with $\mathbf{s} = (1, 0)$. Then $\mathbf{s}$-utilitarianism selects a, $\mathbf{s}$-egalitarianism selects $\frac{1}{2}a + \frac{1}{2}b$, and $\mathbf{s}$-Nash selects $\frac{2}{3}a + \frac{1}{3}b$.

Extending preferences For normative analysis, it is useful to extend voters' rankings of the projects to (partial) preferences over distributions. We assume *linear preferences*: there is an unknown utility function $u_i : A \to \mathbb{R}$ consistent with $\succ_i$ such that i prefers those distributions p with higher average utility $\sum_{x\in A} u_i(x)p_x$. A classical way of ranking distributions despite not knowing u_i uses *stochastic dominance* (SD).

If p and q are (possibly partial) distributions, we write

$$p \succcurlyeq_i^{\text{SD}} q \iff \textstyle\sum_{x_k \succ_i x_j} p_{x_k} \geqslant \sum_{x_k \succ_i x_j} q_{x_k} \text{ for all } x_j \in A.$$

This definition is justified by the following standard equivalence: We have $p \succcurlyeq_i^{\text{SD}} q$ if and only if $\sum_{x\in A} u_i(x)p_x \geqslant \sum_{x\in A} u_i(x)q_x$ for all utility functions $u_i : A \to \mathbb{R}$ satisfying $\min_{x\in A} u_i(x) = 0$ and $u_i(x_j) > u_i(x_k)$ iff $x_j \succ_i x_k$. The condition that the utility of the worst project is 0 is necessary to allow SD-comparisons of partial distributions: we assume that voters are indifferent between not spending part of the budget or spending it on their worst project. This is crucial for the definition of the SD-core in Section 6.4.

6.3. Computation and Basic Properties

In this section, we look at elementary properties of the family of rules we have defined. We will note that several of the rules are familiar from the probabilistic context. We also study the computational complexity of finding an optimal distribution.

Utilitarianism From a utilitarian perspective, it never pays to spend part of the budget on projects whose total $\mathbf{s}$-score is not maximal: shifting that spending to an $\mathbf{s}$-maximal project increases utilitarian welfare. Thus, up to ties, $\mathbf{s}$-utilitarianism never mixes and spends all resources on the $\mathbf{s}$-winner. Formally, $\mathbf{s}$-utilitarianism selects those distributions p for which $p_{x_j} > 0$ only if the score $\sum_{i\in N} \mathbf{s}[i, j]$ is maximum.

Since the behaviour of $\mathbf{s}$-utilitarianism is familiar from work on scoring rules in voting, we will not study it in much detail.

Egalitarianism Plurality-egalitarianism is easy to understand: it returns the uniform distribution over all projects that are ranked top by at least one voter. In the probabilistic context, this rule is known as *egalitarian simultaneous reservation* [Aziz and Stursberg, 2014]. For other scoring vectors, $\mathbf{s}$-egalitarianism is less simple, and it need not return a uniform distribution (see the example of Section 6.1). However, one can easily evaluate $\mathbf{s}$-egalitarianism using linear programming:

$$\begin{aligned} \text{maximise } t^* \text{ s.t. } \quad & \textstyle\sum_{j=1}^m \mathbf{s}[i, j] \cdot p_j \geqslant t^* \text{ for } i \in N \\ & \textstyle\sum_{j=1}^m p_j = 1, \text{ and } p_j \geqslant 0 \text{ for } x_j \in A \end{aligned}$$

Algorithm 2 Computing an $\mathbf{s}$-leximin distribution

Input: A profile given by utilities u_i^j of voter i for project j.
Let $N' \leftarrow \emptyset$ be the set of agents whose utility has been fixed
while $N' \neq N$ **do**
  Using linear programming, find the maximum value t^* such that there exists a distribution $(p_1, \dots, p_m)$ satisfying

$$\begin{aligned} &\textstyle\sum_{j=1}^m \mathbf{s}[i,j]p_j \geqslant t^* \quad && \text{for } i \in N \setminus N' \\ &\textstyle\sum_{j=1}^m \mathbf{s}[i,j]p_j = t_i \quad && \text{for } i \in N' \end{aligned}$$

  for each $i' \in N \setminus N'$ **do**
    Using linear programming, find the maximum ε such that there exists a distribution $(p_1, \dots, p_m)$ satisfying

$$\begin{aligned} &\textstyle\sum_{j=1}^m \mathbf{s}[i',j]p_j \geqslant t^* + \varepsilon && \\ &\textstyle\sum_{j=1}^m \mathbf{s}[i,j]p_j \geqslant t^* \quad && \text{for } i \in N \setminus N' \\ &\textstyle\sum_{j=1}^m \mathbf{s}[i,j]p_j = t_i \quad && \text{for } i \in N' \end{aligned}$$

    If $\varepsilon = 0$, add i' to N' and set $t_{i'} \leftarrow t^*$.
return the solution $(p_1, \dots, p_m)$ of the last LP solved

Now, $\mathbf{s}$-egalitarianism is not very decisive, and may select Pareto-inferior outcomes. When $P = (abcd, acbd, bdac)$, and $\mathbf{s} = (1,1,0,0)$, it selects all distributions of the form

$$p \cdot a + q \cdot b + (\tfrac{1}{2} - p) \cdot c + (\tfrac{1}{2} - q) \cdot d$$

where $0 \leqslant p, q \leqslant 1$ and $\frac{1}{2} \leqslant p + q \leqslant 1$. Note that d can get a positive fraction even though every voter prefers b to d (so that d is Pareto-dominated). A standard way of making egalitarianism more decisive and more efficient is by using leximin instead. In the above example, $\mathbf{s}$-leximin uniquely selects $\frac{1}{2}a + \frac{1}{2}b$. It is easy to see that $\mathbf{s}$-leximin will never give a positive fraction to a Pareto-dominated project.

It is still possible to evaluate $\mathbf{s}$-leximin in polynomial time, by solving $O(n^2)$ linear programs successively. Our algorithm uses the convexity of $\Delta(A)$, which allows it to greedily fix the identity of the agent who is worst-off in the current iteration.

Theorem 6.3. *For every* $\mathbf{s}$*, one can compute a distribution selected by* $\mathbf{s}$*-leximin in polynomial time.*

Proof. The algorithm is specified as Algorithm 2 that requires running at most $n(n+1)/2$ linear programs.

We argue by induction on $|N'|$ that every distribution p selected by $\mathbf{s}$-leximin satisfies $\sum_{j=1}^m u_i^j p_j = t_i$ for each $i \in N'$. This is vacuously true if $N' = \emptyset$. Suppose it is true at some point in the algorithm. Let p be a distribution selected by $\mathbf{s}$-leximin. Because p satisfies the inductive hypothesis, from the upper LP, we know that the least $\mathbf{s}$-score obtained by a voter in $N \setminus N'$ under p is t^*. But which voter? The lower LP tests, for each $i' \in N \setminus N'$, whether i' obtains $\mathbf{s}$-score exactly t^* in all leximin distributions. Such a voter must exist: suppose not, and for each $i' \in N \setminus N'$, let $p_{(i')}$ be a leximin distribution where i' obtains $\mathbf{s}$-score strictly higher than t^*. Write $p' = \sum_{i' \in N \setminus N'} \frac{1}{|N \setminus N'|} p_{(i')}$. Then p' satisfies the inductive hypothesis (because each $p_{(i')}$ does), but the least $\mathbf{s}$-score obtained by a voter in $N \setminus N'$ under p' is strictly higher than t^*, contradicting choice of t^*. Thus, there is a voter $i^* \in N \setminus N'$ who obtains $\mathbf{s}$-score t^* in all leximin distributions. Such a voter is found by the algorithm and added to N', establishing the inductive step. $\square$

Nash product The defining optimisation problem

$$\begin{aligned} \text{maximise} \quad & \textstyle\sum_{i \in N} \log \left(\sum_{j=1}^{m} \mathbf{s}[i,j] \cdot p_j \right) \\ \text{s.t.} \quad & \textstyle\sum_{j=1}^{m} p_j = 1, \text{ and } p_j \geqslant 0 \text{ for } x_j \in A \end{aligned}$$

of **s**-Nash is a convex program which can be efficiently solved using standard solvers. Formally, one can approximate optimum Nash welfare within an additive factor of ε in time polynomial in n, m, and $1/\varepsilon$. Thus, all the usual decision problems associated with computing **s**-Nash are easy. However, writing down the precise output in decimal expansion is impossible, as there are instances where **s**-Nash uniquely returns a distribution with irrational fractions. For instance, for $P = (abc, acb, cab, cab)$ and $\mathbf{s} = (2, 1, 0)$, **s**-Nash uniquely returns $\frac{1+\sqrt{33}}{8}a + \frac{7-\sqrt{33}}{8}c$.

To further understand **s**-Nash, let us analyse the first-order conditions of the convex program. Write down the Lagrangian

$$\mathcal{L} = \textstyle\sum_{i \in N} \log \left(\sum_{j=1}^{m} \mathbf{s}[i,j] \cdot p_j \right) - \lambda \cdot (1 - \sum_{j=1}^{m} p_j).$$

At an optimal solution p, we have

$$\tfrac{\partial \mathcal{L}}{\partial p_j} = \textstyle\sum_{i \in N} \frac{\mathbf{s}[i,j]}{\mathbf{s}[i,p]} - \lambda \leqslant 0, \text{ with equality if } p_j > 0.$$

This implies $\lambda p_j = \sum_{i \in N} \frac{\mathbf{s}[i,j]}{\mathbf{s}[i,p]} p_j$. Summing over all j, thus

$$\lambda = \lambda(p_1 + \cdots + p_m) = \textstyle\sum_{j=1}^{m} \sum_{i \in N} \frac{\mathbf{s}[i,j]}{\mathbf{s}[i,p]} p_j = n,$$

since $\mathbf{s}[i,p] = \sum_{j=1}^{m} \mathbf{s}[i,j] \cdot p_j$ by definition. It follows that

$$n \geqslant \textstyle\sum_{i \in N} \frac{\mathbf{s}[i,j]}{\mathbf{s}[i,p]}, \text{ with equality if } p_j > 0. \tag{6.1}$$

For example, using (6.1), we can characterise plurality-Nash [see also Moulin, 2003, Example 3.6]:

Theorem 6.4. *Plurality-Nash selects p with $p_j = \mathrm{pl}(x_j)/n$ for all j, where $\mathrm{pl}(x_j)$ is the number of voters placing x_j top.*

Proof. Let p be optimal for plurality-Nash. If some voter i puts x_j top then $p_j > 0$, or else $\mathbf{s}[i,p] = 0$ and the Nash product equals 0. By (6.1), we get $n = \sum_{i \in N} \frac{\mathbf{s}[i,j]}{\mathbf{s}[i,p]} = \mathrm{pl}(x_j)/p_j$, and so $p_j = \mathrm{pl}(x_j)/n$. It follows that $p_j = 0$ whenever no voter places x_j top. □

Thus, we see that plurality-Nash is the same rule as *random dictatorship*, familiar from the probabilistic context.

The *veto-Nash* rule seems sensible when projects are nuisances, where each agent wants to minimise the amount spent on the worst option. In some sense, veto-Nash for nuisances is as relevant as plurality-Nash for goods, in the portioning context. Mathematically, veto-Nash is also well-behaved. While we do not provide a closed formula, the following result shows that an exact optimum for veto-Nash can be found in polynomial time (and that it is rational). It gives a collection of at most m different explicit rational distributions, and guarantees that the veto-Nash optimum is among them.

Theorem 6.5. *Let P be a profile, and let $\mathrm{vt}(x_j)$ be the number of voters placing x_j bottom. Relabel projects so that $\mathrm{vt}(x_1) \leqslant \cdots \leqslant \mathrm{vt}(x_m)$. If $\mathrm{vt}(x_j) = 0$ for some x_j, veto-Nash selects all distributions over such projects. Otherwise, there is some $k \in [m]$ with $(k-1)\mathrm{vt}(x_k) < \sum_{j=1}^{k} \mathrm{vt}(x_j)$, such that veto-Nash selects the distribution p with*

$$p_j = 1 - \frac{(k-1)\mathrm{vt}(x_j)}{\sum_{l=1}^{k} \mathrm{vt}(x_l)} \text{ if } j \in [k], \text{ and } p_j = 0 \text{ otherwise.}$$

Proof. If $\mathrm{vt}(x_j) = 0$ for some x_j, then the best-possible Nash product of 1 can be achieved, and is achieved precisely by distributions whose support consists of never-vetoed projects.

Now suppose that $\mathrm{vt}(x_j) > 0$ for all $x_j \in A$. Let p be a distribution selected by veto-Nash, and take k maximal such that $p_k > 0$. Then we must also have $p_j > 0$ for all $j = 1, \dots, k-1$. (If not, and $p_j = 0$ for some j, consider the distribution q with $q_l = p_l$ for all l, except that $q_j = q_k = \frac{1}{2}p_k$. Then, since $\mathrm{vt}(x_j) \leqslant \mathrm{vt}(x_k)$, q has strictly higher Nash product than p, contradiction.) Thus, the support of p is $\{x_1, \dots, x_k\}$.

For $i = 1, \dots, k$, equation (6.1) applies and can be written as

$$n = \sum_{j \in [k] \setminus \{i\}} \frac{\mathrm{vt}(x_j)}{1 - p_j} + \sum_{j=k+1}^{m} \mathrm{vt}(x_j). \tag{6.2}$$

Summing the equations (6.2) for $i = 1, \dots, k$, we get

$$nk = (k-1) \sum_{j \in [k]} \frac{\mathrm{vt}(x_j)}{1 - p_j} + k \sum_{j=k+1}^{m} \mathrm{vt}(x_j).$$

Using $n = \sum_{j=1}^{k} \mathrm{vt}(x_j) + \sum_{j=k+1}^{m} \mathrm{vt}(x_j)$, rearrange this as

$$\sum_{j \in [k]} \frac{\mathrm{vt}(x_j)}{1 - p_j} = \frac{k}{k-1} \sum_{j=1}^{k} \mathrm{vt}(x_j).$$

From the symmetry of the equations (6.2), the values $\frac{\mathrm{vt}(x_i)}{1-p_i}$ must be equal for all $i \in [k]$. Since we know their sum, we get

$$\frac{\mathrm{vt}(x_i)}{1 - p_i} = \frac{1}{k-1} \sum_{j=1}^{k} \mathrm{vt}(x_j) \quad \text{for all } i \in [k].$$

Rearranging, we arrive at the conclusion that

$$p_i = 1 - \frac{k-1}{\sum_{j=1}^{k} \mathrm{vt}(x_j)} \mathrm{vt}(x_i) \quad \text{for all } i \in [k].$$

These values sum to 1, and are non-negative provided that $(k-1)\mathrm{vt}(x_k) < \sum_{j=1}^{k} \mathrm{vt}(x_j)$. If this condition is not satisfied, the choice of k cannot lead to a veto-Nash optimum. □

This gives an algorithm for computing veto-Nash exactly: if some projects are never vetoed, return any distribution over these. Otherwise iterate over all $k \in [m]$ satisfying the condition of the theorem and calculate the corresponding distribution, and return the one with highest Nash product.

Example 6.6. If 2, 3, 3 and 5 voters rank x_1, x_2, x_3 and x_4 last, respectively, then $k = 2$ and 3 satisfy the condition of Thm. 6.5. Thus, either $p = \frac{3}{5}x_1 + \frac{2}{5}x_2$ or $p' = \frac{1}{2}x_1 + \frac{1}{4}x_2 + \frac{1}{4}x_3$ is optimal. The former has higher Nash product, so p is optimal.

Theorems 6.4 and 6.5 show that both plurality-Nash and veto-Nash are rational. Are there any other score vectors $\mathbf{s}$ such that $\mathbf{s}$-Nash is guaranteed to be rational? The answer is no: for every $\mathbf{s}$ other than plurality and veto, we can construct a profile where $\mathbf{s}$-Nash uniquely returns an irrational distribution. This result suggests that a convex programming solver is the best way of computing $\mathbf{s}$-Nash for $\mathbf{s}$ other than plurality and veto.

Theorem 6.7. *Let $m \geqslant 3$, and let $\mathbf{s} = (s_1, \dots, s_m) \in \mathbb{Q}^m$ be a score vector with $s_m = 0$ and normalised so that $s_1 = 1$. Unless $\mathbf{s} = (1, 0, \dots, 0)$ or $\mathbf{s} = (1, \dots, 1, 0)$, there exists a profile $P \in A!^n$ for some $n \in \mathbb{N}$ such that $\mathbf{s}$-Nash returns a unique distribution p with $p \notin \mathbb{Q}^m$.*

Proof. We construct four infinite families of examples, for different shapes of score vectors $\mathbf{s}$. We only consider the case $m = 3$ here, and only sketch the algebra required. The other families require a more involved construction, but work using similar calculus.

Suppose $m = 3$, and let $\mathbf{s} = (1, \frac{r}{s}, 0)$, where $0 < \frac{r}{s} < 1$ and $\frac{r}{s}$ is in lowest terms. Let c be a large-enough integer. Consider the following profile: c voters with abc, one voter bac, one voter with bca. Note that b Pareto-dominates c, so that $p_c = 0$. Let $(x, 1-x, 0)$ be the distribution selected by $\mathbf{s}$-Nash. One can show that $0 < x < 1$ if c is large enough. Now, the Nash product obtained by this distribution is

$$(x + \tfrac{r}{s}(1-x))^c \cdot ((1-x) + \tfrac{r}{s}x) \cdot (1-x).$$

By optimality, x must make the derivative d/dx vanish. After a calculation, cancelling non-zero factors, this implies that

$$\begin{aligned} &((c+2)(r-s)^2) \cdot x^2 \\ +\, &(-(r-s)((c+3)r - 2(c+1)s)) \cdot x \\ +\, &(r^2 - 2rs - crs + cs^2) = 0 \end{aligned}$$

This is a quadratic equation with integer coefficients. Solutions to the equation $ax^2 + bx + c = 0$ involve the term $\sqrt{b^2 - 4ac}$; thus, they are rational if and only if the discriminant $b^2 - 4ac$ is a perfect square. In this case, the discriminant simplifies to

$$(c+1)^2 r^2 + 4(r+1).$$

The first summand is a large perfect square, and the second summand is a constant. Since the distance between consecutive perfect squares is large (in the sense that $(z+1)^2 - z^2 = 2z + 1 = \Theta(z)$), the discriminant cannot be a perfect square for large enough c. Hence, x is irrational. □

6.4. Fairness, Proportionality, and the SD-core

Usually, $\mathbf{s}$-utilitarianism spends 100% on a single project. Some agents might rank this project in a very low position, or even in last place. In some contexts, this is unfair and might rule out $\mathbf{s}$-utilitarianism. In this section, we formalise several notions of fairness, and show that $\mathbf{s}$-egalitarianism satisfies individual fairness, and that $\mathbf{s}$-Nash satisfies group fairness.

A minimal fairness axiom is *positive share* [adapted from Bogomolnaia et al., 2005] which requires that if voter i ranks x in last position, then $p_x < 1$. Hence, for every voter, a positive amount is spent on projects that they do not rank in last position. As suggested above, $\mathbf{s}$-utilitarianism fails positive share for any $\mathbf{s}$.[1] However, provided that $s_m = 0$, positive share is satisfied by $\mathbf{s}$-egalitarianism, $\mathbf{s}$-leximin, and $\mathbf{s}$-Nash. To see this, note that the uniform distribution has positive egalitarian and Nash welfare, whereas a distribution violating positive share has zero egalitarian and Nash welfare.

We can strengthen positive share to *individual fair share*, requiring that if voter i ranks x in last position, then $p_x \leqslant 1 - \frac{1}{n}$. Thus, for each voter, at least $\frac{1}{n}$ is spent on projects not ranked last. Note that the distribution identified by random dictatorship satisfies this condition and has egalitarian welfare at least $\frac{1}{n}$, normalizing $s_1 = 1$. Thus, the optimum $\mathbf{s}$-egalitarian welfare is at least $\frac{1}{n}$, and hence $\mathbf{s}$-egalitarianism and $\mathbf{s}$-leximin satisfy individual fair share (recalling that $s_m = 0$). Below, we show that $\mathbf{s}$-Nash also satisfies it.

Consider $A = \{a, b\}$, with 9 voters ab and 1 voter ba. Then $\mathbf{s}$-egalitarianism returns $\frac{1}{2}a + \frac{1}{2}b$. While this is individually fair, the group of 9 voters is underrepresented. If we desire fairness

[1] Take $(m-1)!$ voters ranking x_1 top and the other projects in all possible ways. Copy these voters sufficiently often. Add a voter ranking x_1 bottom. Then the $\mathbf{s}$-score of x_1 is strictly highest, so $\mathbf{s}$-utilitarianism spends 100% on x_1, violating positive share.

to groups, we need a stronger axiom. One option is this: if k out of n voters rank x last, then $p_x \leqslant 1 - \frac{k}{n}$, so at least $\frac{k}{n}$ is spent on projects other than x. This condition is failed by **s**-egalitarianism and **s**-leximin, but **s**-Nash satisfies it. In our example, **s**-Nash picks $\frac{9}{10}a + \frac{1}{10}b$.

All the notions above focus on avoiding voters' last-ranked project. Despite working in an ordinal setting, using the SD-extension, we can define a group fairness notion that uses more than just the last-ranked project. An important underlying intuition is that agents are "entitled" to $1/n$ of the budget, and this share should be spent in accordance to their preferences. Similarly, a group $S \subseteq N$ of k agents could pool together and be entitled to k/n of the budget.

The intuitive notion of entitlement can be formalised using a core-style concept. A coalition $S \subseteq N$ of voters is supposed to be able to 'control' a fraction of $|S|/n$ of the entire budget. The notion of control is ambiguous since coalitions may overlap and each share of the budget is simultaneously controlled by several coalitions. However, the entitlement of S is certainly violated under p if S can come up with a way of using only its entitlement $|S|/n$ which all members prefer to the way that p uses the entire budget.

Definition 6.8. A coalition $S \subseteq N$ *SD-blocks* a distribution p if there exists a partial distribution z with $\sum_{x\in A} z_x = |S|/n$ such that $z \succcurlyeq_i^{\mathrm{SD}} p$ for all $i \in S$, and $z \succ_j^{\mathrm{SD}} p$ for some $j \in S$. A distribution p is in the SD-core if no coalition SD-blocks p.

If a distribution p lies in the SD-core, then it also satisfies all of our previous conditions: Suppose not, and consider the coalition S of voters that rank x last, where $p_x > 1 - |S|/n$. Then the coalition S can SD-block p: Write $p_x = 1 - |S|/n + \varepsilon$ for some $\varepsilon > 0$, and define the partial distribution z as follows:

$$z_y = p_y + \varepsilon/(m-1) \quad \text{for all } y \in A \setminus \{x\}, \text{ and} \quad z_x = 0.$$

Then $\sum_{a\in A} z_a = \varepsilon + \sum_{y\in A\setminus\{x\}} p_y = \varepsilon + (1 - p_x) = |S|/n$, so that z has the required total weight. It is easy to check that $z \succ_i^{\mathrm{SD}} p$ for all $i \in S$. Thus, p is not in the SD-core.

For an example, consider the profile with three voters abc, acb, bca. Which distributions p are in the SD-core? First, singleton coalitions $\{i\}$ block p if $p_x > \frac{2}{3}$ for any x, using $z = \frac{1}{3}y$ where y is i's top project (this is individual fair share). Also, the coalition consisting of abc and acb blocks all p with $p_a + p_b \leqslant \frac{2}{3}$ and $p_a + p_c \leqslant \frac{2}{3}$ (one inequality strict), using $z = \frac{2}{3}a$. All other distributions are in the SD-core. This is drawn in Figure 6.1, which shows the simplex of all distributions, where the SD-core is shaded. Note that the SD-core is not convex for this profile.

Figure 6.1 shows the outputs of **s**-Nash for all **s** as a blue line. The blue line is entirely contained in the SD-core. In fact, **s**-Nash is always in the SD-core. We give a direct argument using equation (6.1). The result can also be obtained via the theory of Lindahl equilibrium [Fain et al., 2016, Foley, 1970].

Theorem 6.9. *For any* **s** *with* $s_m = 0$*, any distribution selected by* **s***-Nash is in the SD-core.*

Proof. Suppose p is selected by **s**-Nash. For a contradiction, assume that $S \subseteq N$ is a blocking coalition of agents, deviating using $(z_1, \dots, z_m) \in [0,1]^m$ with $\sum_{j=1}^m z_j = |S|/n$, such that $z \succcurlyeq_i^{\mathrm{SD}} p$ for all $i \in S$, and $z \succ_j^{\mathrm{SD}} p$ for some $j \in S$. Now, **s** defines utilities compatible with the voters' ordinal preferences, and thus $\mathbf{s}[i,z] \geqslant \mathbf{s}[i,p]$ for all $i \in S$, and $\mathbf{s}[j,z] > \mathbf{s}[j,p]$ for some $j \in S$. Then

$$|S| = n \cdot \sum_{j=1}^{m} z_j \overset{(6.1)}{\geqslant} \sum_{i\in N} \frac{\sum_{j=1}^m \mathbf{s}[i,j] z_j}{\mathbf{s}[i,p]} = \sum_{i\in N} \frac{\mathbf{s}[i,z]}{\mathbf{s}[i,p]} > |S|.$$

The last inequality follows because the sum contains only non-negative terms, $|S|$ of which are at least 1, and one of which is strictly larger than 1. This is a contradiction. □

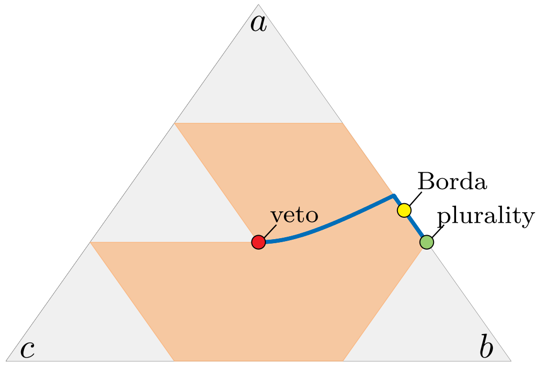


Figure 6.1.: The SD-core of the profile (abc, acb, bca) within the simplex of all distributions. The shaded area shows the distributions that are in the SD-core. The blue line shows the output of **s**-Nash for all $\mathbf{s} = (1, q, 0)$ with $q \in [0, 1]$. Plurality-Nash selects $\frac{2}{3}a + \frac{1}{3}b$, Borda-Nash selects $0.58a + 0.42b$, and veto-Nash selects $\frac{1}{3}a + \frac{1}{3}b + \frac{1}{3}c$.

Thus, the **s**-Nash rules are particularly fair to groups. The SD-core can also be seen as a *proportionality* requirement: the common resource should be divided so that the share of a project is proportional to its support. Such a property is of particular interest in political contexts, for example when we are dividing parliament seats among parties.

6.5. Other Axiomatic Properties

We now briefly study other axiomatic properties of our rules. We ignore ties when defining strategyproofness and monotonicity.

Pareto-efficiency A distribution q *SD-dominates* p if $q \succcurlyeq_i^{\mathrm{SD}} p$ for all $i \in N$, and $q \succ_j^{\mathrm{SD}} p$ for some $j \in N$. A distribution p is *SD-efficient* if no distribution dominates it. Note that SD-core implies SD-efficiency (with $S = N$), and so **s**-Nash rules are SD-efficient when $s_m = 0$. More generally, one can show that **s**-utilitarianism, **s**-leximin, and **s**-Nash are SD-efficient provided that **s** is strictly decreasing, because if a distribution q were to SD-dominate the output p of any of these rules, then the **s**-score of q would be higher than the **s**-score of p for each voter, which would contradict that p maximises the notion of welfare implicit in the rule.

Strategyproofness A social decision scheme is *(strongly) SD-strategyproof* if, when a voter misreports their ranking, the SDS selects a distribution that the voter believes is weakly SD-worse than the distribution resulting from a truthful report. Plurality-Nash (i.e., random dictatorship) is strategyproof in this sense. A well-known result of Gibbard [1977] shows that this is the only SDS that is strategyproof and also anonymous and Pareto-efficient. Hence, all other SD-efficient rules we have considered are manipulable. (Outside of our class of rules, Barberà [1979] shows that *proportional Borda* is a strategyproof rule, but it is not efficient.)

Monotonicity An SDS F is *monotone* if, when we change a profile P into P' by moving up a project x in a voters' ranking (by swapping), then the share of x weakly increases, i.e., $F(P')_x \geqslant F(P)_x$. This is clearly satisfied by **s**-utilitarianism, and also by plurality-Nash. However, other **s**-Nash rules (and also **s**-leximin) may fail it. If $\mathbf{s} = (2, 1, 0)$ and $P = (abc, abc, abc, acb, bac, cba)$, then **s**-Nash selects an irrational distribution which rounds to $0.642a + 0.333b + 0.024c$. If the bac voter moves c up one place (to get bca), then **s**-Nash selects $0.5a + 0.5b$. Thus, c's share has strictly decreased. Monotonicity is a kind of fairness to projects (x gets more if it performs better), while our rules aim for fairness to *voters*.

6.6. Conclusions

We have introduced a class of aggregation rules which can be used to make budget decisions. We have found that our rules are attractive on efficiency and fairness grounds. Formally, these rules can be seen as outputting probabilities; thus, they may be of interest to probabilistic social choice. More generally, the connections and differences between randomisation and splitting a common resource need to be discussed further.

We have introduced concepts such as the SD-core which is a group fairness and proportionality notion. Our rules based on maximising Nash welfare satisfy this property. However, one can criticise these rules as making an arbitrary choice of the score vector used. We are not aware of any other rules in the literature satisfying the SD-core, but maybe attractive other examples of such rules can be found.

Part III.

Budgeting with Indivisible Projects: Committee Elections

7. Strategyproof Committee Selection

Multiwinner voting rules can be used to select a fixed-size committee from a larger set of candidates. We consider approval-based committee rules, which allow voters to approve or disapprove candidates. In this setting, several voting rules such as Proportional Approval Voting (PAV) and Phragmén's rules have been shown to produce committees that are proportional, in the sense that they proportionally represent voters' preferences; all of these rules are strategically manipulable by voters. On the other hand, a generalisation of Approval Voting gives a non-proportional but strategyproof voting rule. We show that there is a fundamental tradeoff between these two properties: we prove that no multiwinner voting rule can simultaneously satisfy a weak form of proportionality (a weakening of justified representation) and a weak form of strategyproofness. Our impossibility is obtained using a formulation of the problem in propositional logic and applying SAT solvers; a human-readable version of the computer-generated proof is obtained by extracting a minimal unsatisfiable set (MUS).

7.1. Introduction

The theory of multiwinner elections is concerned with designing and analysing procedures that, given preference information from a collection of voters, select a fixed-size *committee* consisting of k members, drawn from a larger set of m candidates. Often, we will be interested in picking a *representative* committee whose members together cover the diverse interests of the voters. We may also aim for this representation to be *proportional*; for example, if a group of 20% of the voters have similar interests, then about 20% of the members of the committee should represent those voters' interests.

Historically, much work in mathematical social science has tried to formalise the latter type of proportionality requirement, in the form of finding solutions to the *apportionment problem*, which arises in settings where voters express preferences over *parties* which are comprised of many candidates [Balinski and Young, 1982]. More recently, theorists have focussed on cases where there are no parties, and preferences are expressed directly over the candidates [Faliszewski et al., 2017a]. The latter setting allows for applications in areas outside the political sphere, such as in group recommendation systems.

To formalise the requirement of proportionality in this party-free setting, it is convenient to consider the case where input preferences are given as *approval ballots*: each voter reports a set of candidates that they find acceptable. Even for this simple setting, there is a rich variety of rules that exhibit different behaviour [Kilgour, 2010].

One natural way of selecting a committee of k candidates when given approval ballots is to extend *Approval Voting* (AV): for each of the m candidates, count how many voters approve them (their *approval score*), and then return the committee consisting of the k candidates whose approval score is highest. Notably, this rule can produce committees that fail to represent large groups of voters. Consider, for example, an instance where $k = 3$, and where 5 voters approve candidates a, b and c, while 4 other voters approve only the candidate d. Then AV would select the committee $\{a, b, c\}$, leaving almost half of the electorate unrepresented. Intuitively, the latter group of 4 voters, consisting of more than a third of the electorate, should be represented by at least 1 of the 3 committee members.

Aziz et al. [2017] introduce an axiom called *justified representation* (JR) which formalises this intuition that a group of n/k voters should not be left without any representation; a stronger version of this axiom called *proportional justified representation* (PJR) has also been introduced and studied [Sánchez-Fernández et al., 2017]. While AV fails these axioms, there are appealing rules which satisfy them. An example is *Proportional Approval Voting* (PAV), first proposed by Thiele [1895]. The intuition behind this rule is that voters prefer committees which contain more of their approved candidates, but that there are decreasing marginal returns; specifically, let us presume that voters gain 1 'util' in committees that contain exactly 1 approved candidates, $1 + \frac{1}{2}$ utils with 2 approved candidates, and in general $1 + \frac{1}{2} + \dots + \frac{1}{r}$ utils with r approved candidates. PAV returns the committee that maximises utilitarian social welfare with this choice of utility function. PAV satisfies a strong form of justified representation [Aziz et al., 2017].

When voters are strategic, PAV has the drawback that it can be manipulated. Indeed, suppose a voter i approves candidates a and b. If a is also approved by many other voters, PAV is likely to include a in its selected committee anyway, but it might not include b because voter i is already happy enough due to the inclusion of a. However, if voter i pretends not to approve a, then it may be utility-maximising for PAV to include both a and b, so that i successfully manipulated the election.[1] Besides PAV, there exist several other proportional rules, such as rules proposed by Phragmén [Janson, 2016, Brill et al., 2017], but all of them can be manipulated using a similar strategy.

That voting rules are manipulable is familiar to voting theorists; indeed the Gibbard–Satterthwaite theorem shows that for single-winner voting rules and strict preferences, *every* non-trivial voting rule is manipulable. However, in the approval-based multiwinner election setting, we have the tantalising example of Approval Voting (AV): this rule is strategyproof in the sense that voters cannot induce AV to return a committee including more approved candidates by misrepresenting their approval set. This raises the natural question of whether there exist committee rules that combine the benefits of AV and PAV: are there rules that are simultaneously proportional and strategyproof?

The contribution of this chapter is to show that these two demands are incompatible. No approval-based multiwinner rule satisfies both requirements. This impossibility holds even for very weak versions of proportionality and of strategyproofness. The version of proportionality we use is much weaker than JR. It requires that if there is a group of at least n/k voters who all approve a certain candidate c, and none of them approve any other candidate, and no other voters approve c, then c should be part of the committee. Strategyproofness requires that a voter cannot manipulate the committee rule by dropping candidates from their approval ballot; a manipulation would be deemed successful if the voter ends up with a committee that contains additional approved candidates. In particular, our notion of strategyproofness only requires that the committee rule be robust to *dropping* candidates; we do not require robustness against arbitrary manipulations that both add and remove candidates (we considered a similar property in Chapter 5). Additionally, we impose a mild efficiency axiom requiring that the rule not elect candidates who are approved by none of the voters.

The impossibility theorem is obtained using the computer-aided techniques that we discussed in earlier chapters. We encode the problem of finding a committee rule satisfying our axioms into propositional logic, and then use a SAT solver to check whether the formula is satisfiable. If the formula is unsatisfiable, this implies an impossibility, for a fixed number of voters, a fixed number of candidates, and a fixed k. We can then manually prove induction steps showing that the impossibility continues to hold for larger parameter values.

As mentioned in the Introduction of the thesis (Chapter 0), committee elections can be seen as

[1] For a specific example, consider $P = (abc, abc, abc, abd, abd)$ for which abc is the unique PAV-committee for $k = 3$. If the last voter instead reports to approve d only, then the unique PAV-committee is abd.

a special case of deciding how to spend a common budget when the choices are indivisible. In this model, there is a budget of B dollars and a collection C of projects. Each project $c \in C$ comes with a cost or price p_c, and we need to find a selection $W \subseteq C$ of projects such that $\sum_{c \in W} p_c \leqslant B$. This model captures the way in which many cities run their participatory budgeting projects. Committee elections are formally identical to the special case of *unit costs*, where $p_c = 1$ for all $c \in C$, in which case B is the desired committee size. Many of the proportionality notions we discuss below can be generalized to the non-unit-cost case [see, e.g., Aziz et al., 2018a], and so our result implies that fair methods for participatory budgeting cannot be strategyproof. For more on the budgeting problem without the unit cost assumption, see Benade et al. [2017], Fain et al. [2018], Fluschnik et al. [2019], and Faliszewski and Talmon [2019]. There is also a literature on *combinatorial public projects* [Papadimitriou et al., 2008], which considers VCG-like truthful mechanisms in the presence of money; this literature usually imposes the unit cost assumption. In our discussion, there is no money.

We begin this chapter by describing several possible versions of strategyproofness and proportionality axioms. We then present the proof of our main theorem. We end by discussing some extensions to this result, and contrast our result to a related impossibility theorem due to Duddy [2014b].

7.2. Preliminaries

Let C be a fixed finite set of m *candidates*, and let $N = \{1, \dots, n\}$ be a fixed finite set of n *voters*. An *approval ballot* is a proper[2] subset A_i of C, so that $\emptyset \neq A_i \subsetneq C$; let $\mathcal{B}$ denote the set of all ballots. For brevity, when writing ballots, we often omit braces and commas, so that the ballot $\{a, b\}$ is written ab. An (approval) *profile* is a function $P : N \to \mathcal{B}$ assigning every voter an approval ballot. For brevity, we write a profile P as an n-tuple, so that $P = (P(1), \dots, P(n))$. For example, in the profile (ab, abc, d), voter 1 approves candidates a and b, voter 2 approves a, b, and c, and voter 3 approves d only.

Let k be a fixed integer with $1 \leqslant k \leqslant m$. A *committee* is a subset of C of cardinality k. We write $\mathcal{C}_k$ for the set of committees, and again for brevity, the committee $\{a, b\}$ is written as ab. An (approval-based) *committee rule* is a function $f : \mathcal{B}^N \to \mathcal{C}_k$, assigning to each approval profile a unique winning committee. Note that this definition assumes that f is *resolute*, so that for every possible profile, it returns exactly one committee. In our proofs, we will implicitly restrict the domain of f to profiles P with $|\bigcup_{i \in N} P(i)| \geqslant k$, so that it is possible to fill the committee with candidates who are each approved by at least one voter. Since we are aiming for a negative result, this domain restriction only makes the result stronger.

Let us define two specific committee rules which will be useful examples throughout.

- *Approval Voting (AV)* is the rule that selects the k candidates with highest approval score, that is, the k candidates c for which $|\{i \in N : c \in P(i)\}|$ is highest. Ties are broken lexicographically.

- *Proportional Approval Voting (PAV)* is the rule that returns the set $W \subseteq C$ with $|W| = k$ which maximises
$$\sum_{i \in N} \left(1 + \frac{1}{2} + \dots + \frac{1}{|P(i) \cap W|} \right).$$
In case of ties, PAV returns the lexicographically first optimum.

[2]Nothing hinges on the assumption that ballots are *proper* subsets. Since we are mainly interested in impossibilities, this 'domain restriction' slightly strengthens the results.

Other important examples that we occasionally mention are Monroe's rule, Chamberlin–Courant, Phragmén's rules, and the sequential version of PAV. For definitions of these rules, we refer to the book chapter by Faliszewski et al. [2017a]; they are not essential for following our technical results.

7.3. Our Axioms

In this section, we discuss the axioms that will be used in our impossibility result. These axioms have been chosen to be as weak as possible while still yielding an impossibility. This can make them sound technical and unnatural in isolation. To better motivate them, we discuss stronger versions that may have more natural appeal.

7.3.1. Strategyproofness

A voter can *manipulate* a voting rule if, by submitting a non-truthful ballot, the voter can ensure that the voting rule returns an outcome that the voter strictly prefers to the outcome at the truthful profile. It is not obvious how to phrase this definition for committee rules, since we do not assume that voters have preferences over committees; we only have approval ballots over candidates.

One way to define manipulability in this context is to *extend* the preference information we have to preferences over committees. This is the approach also typically taken when studying set-valued (irresolute) voting rules [Taylor, 2005, Gärdenfors, 1979] or probabilistic voting rules [Brandt, 2017]. In our setting, there are several ways to extend approval ballots to preferences over committees, and hence several notions of strategyproofness. Our impossibility result uses the weakest notion.

For the formal definitions, let us introduce the notion of *i-variants*. For a voter $i \in N$, we say that a profile P' is an i-variant of profile P if P and P' differ only in the ballot of voter i, that is, if $P(j) = P'(j)$ for all $j \in N \setminus \{i\}$. Thus, P' is obtained after i manipulated in some way, assuming that P was the truthful profile.

One obvious way in which one committee can be better than another in a voter's view is if the former contains a larger number of approved candidates. Suppose at the truthful profile, we elect a committee of size $k = 5$, of which voter i approves 2 candidates. If i can submit a non-truthful approval ballots which leads to the election of a committee with 3 candidates who are approved by i, then this manipulation would be successful in the cardinality sense.

Cardinality-Strategyproofness If P' is an i-variant of P, then we do not have $|f(P') \cap P(i)| > |f(P) \cap P(i)|$.

One can check that AV with lexicographic tie-breaking satisfies cardinality-strategyproofness: it is neither advantageous to increase the approval score of a non-approved candidate, nor to decrease the approval score of an approved candidate.

Alternatively, we can interpret an approval ballot $A \in \mathcal{B}$ to say that the voter likes the candidates in A (and would like them to in the committee), and that the voter dislikes the candidates not in A (and would like them not to be in the committee). The voter's 'utility' derived from committee W would be the number of approved candidates in W plus the number of non-approved candidates not in W. Interpreting approval ballots and committees as bit strings of length m, the voter thus desires the *Hamming distance* between their ballot and the committee to be small. For two sets A, B, write $\mathcal{H}(A, B) = |A \,\Delta\, B| = |(A \cup B) \setminus (A \cap B)|$.

Hamming-Strategyproofness If P' is an i-variant of P, then we do not have $\mathcal{H}(f(P'), P(i)) < \mathcal{H}(f(P), P(i))$.

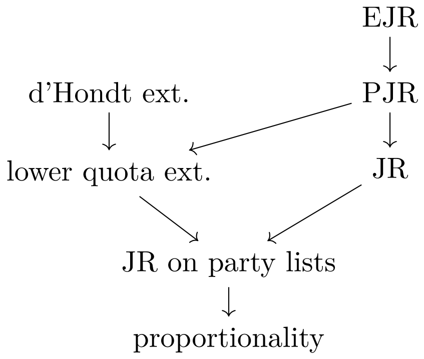


Figure 7.1.: Proportionality axioms and logical implications.

One can check that Hamming-strategyproofness and cardinality-strategyproofness are equivalent, because for a *fixed* ballot $P(i)$, a committee is Hamming-closer to $P(i)$ than another if and only if the number of approved candidates is higher in the former.

The notions of strategyproofness described so far make sense if we subscribe to the interpretation of an approval ballot as a dichotomous preference, with the voter being completely indifferent between all approved candidates (or being unable to distinguish between them). In some settings, this is not a reasonable assumption.

For example, suppose i approves $\{a, b, c\}$; still it might be reasonable for i to prefer a committee containing just a to a committee containing both b and c, maybe because i's underlying preferences are such that a is preferred to b and c, even though all three are approved. However, i should definitely prefer a committee that includes a strict superset of approved candidates. For example, a committee containing a and b should be better than a committee containing only a. This is the intuition behind superset-strategyproofness, which is a weaker notion than cardinality-strategyproofness.

Superset-Strategyproofness If P' is an i-variant of P, then we do not have $f(P') \cap P(i) \supsetneq f(P) \cap P(i)$.

Interestingly, PAV and other proportional rules are often manipulable in a particularly simple fashion: a manipulator can obtain a better outcome by dropping popular candidates from their approval ballot. Formally, these rules can be manipulated even through reporting a proper subset of the truthful ballot. Our final and official notion of strategyproofness is a version of subset-strategyproofness which only requires the committee rule to resist manipulators who report a subset of the truthful ballot.

Strategyproofness If P' is an i-variant of P with $P'(i) \subset P(i)$, then we do not have $f(P') \cap P(i) \supsetneq f(P) \cap P(i)$.

Manipulating by reporting a subset of one's truthful ballot is sometimes known as *Hylland free riding* [Hylland, 1992, Schulze, 2004]: the manipulator free-rides on others approving a candidate, and can pretend to be worse off than they actually are. This can then induce the committee rule to add further candidates from their ballot to the committee.

Interestingly, one can check that PAV cannot be manipulated by reporting a *super*set of one's ballot; such a manoeuvre never helps.

7.3.2. Proportionality

We now discuss several axioms formalising the notion that the committee rule f should be *proportional*, in the sense of proportionally representing different factions of voters: for example, a 'cohesive' group of 10% of the voters should be represented by about 10% of the members of the committee. The version of proportionality used in our impossibility is the last and weakest axiom we discuss. Figure 7.1 shows a Hasse diagram of all discussed axioms. Approval Voting (AV) fails all of them, as can be checked for the example profile $P = (abc, abc, d)$ and $k = 3$, where AV returns *abc*.

We say that a profile P is a *party-list profile* if for all voters $i, j \in N$, either $P(i) = P(j)$, or $P(i) \cap P(j) = \emptyset$. For example, (ab, ab, cde, cde, f) is a party-list profile, but (ab, c, c, abc) is not. A party-list profile induces a partition of the set C of candidates into disjoint *parties*, so that each voter approves precisely the members of exactly one party. The problem of finding a proportional committee given a party-list profile has been extensively studied as the problem of *apportionment*. Functions $g : \{\text{party-list profiles}\} \to \mathcal{C}_k$ are known as *apportionment methods*; thus any committee rule induces an apportionment by restricting its domain to party-list profiles [Brill et al., 2018]. Many proportional apportionment methods have been introduced and defended over the last few centuries. Given a committee rule f, one way to formalise the notion that f is proportional is by requiring that the apportionment method induced by f is proportional.

Given a party-list profile P, let us write $n_P(A) = |\{i \in N : P(i) = A\}|$ for the number of voters approving party A. An apportionment method g satisfies *lower quota* if for every party-list profile P, each party A in P gets at least $\lfloor n_P(A) \cdot \frac{k}{n} \rfloor$ seats, that is, $|g(P) \cap A| \geqslant \lfloor n_P(A) \cdot \frac{k}{n} \rfloor$. This notion gives us our first proportionality axiom.

Lower quota extension The apportionment method induced by f satisfies lower quota.

This axiom is satisfied by PAV, the sequential version of PAV, by Monroe's rule if k divides n, and Phragmén's rule [Brill et al., 2018].

We can strengthen this axiom by imposing stronger conditions on the induced apportionment method. For example, the apportionment method induced by PAV and by Phragmén's rule coincides with the *d'Hondt method* (aka Jefferson method, see [Brill et al., 2018] for a definition), so we could use the following axiom.

d'Hondt extension The apportionment method induced by f is the d'Hondt method.

Aziz et al. [2017] introduce a different approach of defining a proportionality axiom. Instead of considering only the case of party-list profiles, they impose conditions on *all* profiles. The intuition behind their axioms is that sufficiently large groups of voters that have similar preferences 'deserve' at least a certain number of representatives in the committee. They introduce the following axiom:

Justified Representation (JR) If P is a profile, and $N' \subseteq N$ is a group with $|N'| \geqslant \frac{n}{k}$ and $\bigcap_{i \in N'} P(i) \neq \emptyset$, then $f(P) \cap \bigcup_{i \in N'} P(i) \neq \emptyset$.

Thus, JR requires that no group of at least $\frac{n}{k}$ voters for which there is at least one candidate $c \in C$ that they all approve can remain unrepresented: at least one of the voters in the group must approve at least one of the committee members. This axiom is satisfied, for example, by PAV, Phragmén's rule, and Chamberlin-Courant [Aziz et al., 2017], but not by the sequential version of PAV unless $k \leqslant 5$ [Sánchez-Fernández et al., 2017, Aziz et al., 2017].

One may think that JR is too weak: even if there is a large majority of voters who all report the same approval set, JR only requires that *one* of their candidates be a member of the committee. But this group may deserve several representatives. The following strengthened version of JR is due to [Sánchez-Fernández et al., 2017]. It requires that a large group of voters for which there are several candidates that they all approve should be represented by several committee members.

Proportional Justified Representation (PJR) For any profile P and each $\ell = 1, \dots, k$, if $N' \subseteq N$ is a group with $|N'| \geqslant \ell \cdot \frac{n}{k}$ and $|\bigcap_{i \in N'} P(i)| \geqslant \ell$, then $|f(P) \cap \bigcup_{i \in N'} P(i)| \geqslant \ell$.

This axiom is also satisfied by PAV and Phragmén's rule [Sánchez-Fernández et al., 2017, Brill et al., 2017]. Brill et al. [2018] show that if a rule satisfies PJR, then it is also a lower quota extension. A yet stronger version of JR is *EJR*, introduced by Aziz et al. [2017]; EJR requires that there is at least one group member who has at least ℓ approved committee members.

Extended Justified Representation (EJR) For any profile P and each $\ell = 1, \dots, k$, if $N' \subseteq N$ is a group with $|N'| \geqslant \ell \cdot \frac{n}{k}$ and $|\bigcap_{i \in N'} P(i)| \geqslant \ell$, then $|f(P) \cap P(i)| \geqslant \ell$ for some $i \in N'$.

This axiom is satisfied by PAV [Aziz et al., 2017], but not by Phragmén's rule [Brill et al., 2017].

The proportionality axiom we use in our impossibility combines features of the JR-style axioms with the apportionment-extension axioms. Consider the following axiom.

JR on party lists Suppose P is a party-list profile, and some ballot $A \in \mathcal{B}$ appears at least $\frac{n}{k}$ times in P. Then $f(P) \cap A \neq \emptyset$.

This axiom only requires JR to hold for party-list profiles; thus, it only requires that we represent large-enough groups of voters who all report the exact same approval ballot [see also Behrens et al., 2014]. As an example, this axiom requires that $f(ab, ab, cd, cd) \in \{ac, ad, bc, bd\}$, because the ballots ab and cd both appear at least $\frac{n}{k} = \frac{4}{2} = 2$ times.

Our official proportionality axiom is still weaker, and only requires us to represent *singleton* parties with large-enough support.

Proportionality Suppose P is a party-list profile, and some singleton ballot $\{c\} \in \mathcal{B}$ appears at least $\frac{n}{k}$ times in P. Then $c \in f(P)$.

This axiom should be almost uncontroversial if we desire our committee rule to be proportional in any sense. A group of voters who all approve just a single candidate is certainly cohesive (there are *no* internal disagreements), it is clear what it means to represent this group (add their approved candidate to the committee), and the group is uniquely identified (because no outside voters approve sets that intersect with the group's approval ballot).

Since our proportionality axiom only refers to the apportionment method induced by f, our impossibility states that no reasonable apportionment method admits an extension to the 'open list' setting (where voters are not bound to a party) which is strategyproof.

A type of axiom related to proportionality are *diversity* requirements. These typically require that as many voters as possible should have a representative in the committee, but they do not insist that groups of voters be proportionally represented [Elkind et al., 2017a, Faliszewski et al., 2017a]. The Chamberlin–Courant rule [Chamberlin and Courant, 1983] is an example of a rule selecting diverse committees. Lackner and Skowron [2018] propose the following formulation of this requirement for the approval setting:

Disjoint Diversity Suppose P is a party-list profile with at most k different parties. Then $f(P)$ contains at least one member from each party.

Our main result (Theorem 7.1) also holds when replacing proportionality by disjoint diversity, since all profiles in its proof where proportionality is invoked feature at most k different parties.

7.3.3. Efficiency

We will additionally impose a mild technical condition, which can be seen as an efficiency axiom.[3] The axiom will only be used in one of the induction steps (Lemma 7.5).

[3] The conference version of this chapter did not use this axiom. Without it, the proof of Lemma 7.5 does not work. I thank Boas Kluiving, Adriaan de Vries, Pepijn Vrijbergen for pointing out the error.

Weak Efficiency If P is a profile with $|\bigcup_{i\in N} P(i)| \geqslant k$, and c is a candidate who is approved by no voters, then $c \notin f(P)$.

Thus, a rule satisfying weak efficiency should fill the committee with candidates who are approved by some voters, rather than electing candidates approved by no one. A similar axiom of the same name is used by Lackner and Skowron [2018]. As we declared in Section 7.2, in our proofs we will always restrict attention to profiles P with $|\bigcup_{i\in N} P(i)| \geqslant k$, so that weak efficiency applies to all relevant profiles.

7.4. The Impossibility Theorem

We are now in a position to state our main result, that there are no proportional and strategyproof committee rules.

Theorem 7.1. *Suppose $k \geqslant 3$, the number n of voters is divisible by k, and $m \geqslant k+1$. Then there exists no approval-based committee rule which satisfies weak efficiency, proportionality, and strategyproofness.*

The assumption that n be divisible by k appears to be critical; the SAT solver indicates positive results when n is not a multiple of k. However, we do not know short descriptions of these rules, and it is possible (likely?) that impossibility holds for large n and m. Using stronger proportionality axioms, the result holds for all sufficiently large n; see Section 7.4.3.

The proof of this impossibility was found with the help of computers, but it was significantly simplified manually. One convenient first step is to establish the following simple lemma. It uses strategyproofness to extend the applicability of proportionality to certain profiles that are not party-list profiles.

Lemma 7.2. *Let $m = k+1$. Let f be strategyproof and proportional. Suppose that P is a profile in which some singleton ballot $\{c\}$ appears at least $\frac{n}{k}$ times, but in which no other voter approves c. Then $c \in f(P)$.*

Proof. Let P' be the profile defined by

$$P'(i) = \begin{cases} \{c\} & \text{if } P(i) = \{c\}, \\ C \setminus \{c\} & \text{otherwise.} \end{cases}$$

Then P' is a party-list profile, and by proportionality, $c \in f(P')$. Thus, $f(P') \neq C\setminus\{c\}$. Now, step by step, we let each non-$\{c\}$ voter j in P' change back their vote to $P(j)$. By strategyproofness, at each step the output committee cannot be $C \setminus \{c\}$. In particular, at the last step, we have $f(P) \neq C \setminus \{c\}$. Thus, $c \in f(P)$, as required. □

7.4.1. Base case

The first major step in the proof is to establish the impossibility in the case that $k = 3$, $n = 3$, and $m = 4$. The proof of this base case is by contradiction, assuming there exists some f satisfying the axioms. We start by considering the profile $P_1 = (ab, c, d)$, and break some symmetries. (This is a useful strategy to obtain smaller and better-behaved MUSes.) Using proportionality, symmetry-breaking allows us to assume that $f(P_1) = acd$. The proof then goes through seven steps, applying the same reasoning each time. In each step, we use strategyproofness to infer the values of f at certain profiles $P_2, \ldots, P_7$. Finally, we find that strategyproofness implies that $f(P_1) \neq acd$, which contradicts our initial assumption about $f(P_1)$.

Lemma 7.3. *There is no committee rule that satisfies proportionality and strategyproofness for* $k = 3$, $n = 3$, *and* $m = 4$.

Proof. Suppose for a contradiction that such a committee rule f existed. Consider the profile $P_1 = (ab, c, d)$. By proportionality, we have $c \in f(P_1)$ and $d \in f(P_1)$. Thus, we have $f(P_1) \in \{acd, bcd\}$. By relabelling the alternatives, we may assume without loss of generality that $f(P_1) = acd$.

Consider $P_{1.5} = (ab, ac, d)$. By Lemma 7.2, $d \in f(P_{1.5})$. Thus, $f(P_{1.5}) = acd$, or else voter 2 can manipulate towards P_1.

Consider $P_2 = (b, ac, d)$. By proportionality, $f(P_2) \in \{abd, bcd\}$. If we had $f(P_2) = abd$, then voter 1 in $P_{1.5}$ could manipulate towards P_2. Hence $f(P_2) = bcd$.

Consider $P_{2.5} = (b, ac, cd)$. By Lemma 7.2, $b \in f(P_{2.5})$. Thus, $f(P_{2.5}) = bcd$, or else voter 3 can manipulate towards P_2.

Consider $P_3 = (b, a, cd)$. By proportionality, $f(P_3) \in \{abc, abd\}$. If we had $f(P_3) = abc$, then voter 2 in $P_{2.5}$ could manipulate towards P_3. Hence $f(P_3) = abd$.

Consider $P_{3.5} = (b, ad, cd)$. By Lemma 7.2, $b \in f(P_{3.5})$. Thus, $f(P_{3.5}) = abd$, or else voter 2 can manipulate towards P_3.

Consider $P_4 = (b, ad, c)$. By proportionality, $f(P_4) \in \{abc, bcd\}$. If we had $f(P_4) = bcd$, then voter 3 in $P_{3.5}$ could manipulate towards P_4. Hence $f(P_4) = abc$.

Consider $P_{4.5} = (b, ad, ac)$. By Lemma 7.2, $b \in f(P_{4.5})$. Thus, $f(P_{4.5}) = abc$, or else voter 3 can manipulate towards P_4.

Consider $P_5 = (b, d, ac)$. By proportionality, $f(P_5) \in \{abd, bcd\}$. If we had $f(P_5) = abd$, then voter 2 in $P_{4.5}$ could manipulate towards P_5. Hence $f(P_5) = bcd$.

Consider $P_{5.5} = (b, cd, ac)$. By Lemma 7.2, $b \in f(P_{5.5})$. Thus, $f(P_{5.5}) = bcd$, or else voter 2 can manipulate towards P_5.

Consider $P_6 = (b, cd, a)$. By proportionality, $f(P_6) \in \{abc, abd\}$. If we had $f(P_6) = abc$, then voter 3 in $P_{5.5}$ could manipulate towards P_6. Hence $f(P_6) = abd$.

Consider $P_{6.5} = (b, cd, ad)$. By Lemma 7.2, $b \in f(P_{6.5})$. Thus, $f(P_{6.5}) = abd$, or else voter 3 can manipulate towards P_6.

Consider $P_7 = (b, c, ad)$. By proportionality, $f(P_7) \in \{abc, bcd\}$. If we had $f(P_7) = bcd$, then voter 2 in $P_{6.5}$ could manipulate towards P_7. Hence $f(P_7) = abc$.

Finally, consider $P_{7.5} = (ab, c, ad)$. By Lemma 7.2, $c \in f(P_{7.5})$. Thus, $f(P_{7.5}) = abc$, or else voter 1 can manipulate towards P_7. But then voter 3 can manipulate towards $P_1 = (ab, c, d)$, because by our initial assumption, we have $f(P_1) = acd$. Contradiction. □

7.4.2. Induction steps

We now extend the base case to larger parameter values, by proving induction steps. The proofs all take the same form: Assuming the existence of a committee rule satisfying the axioms for large parameter values, we construct a rule for smaller values, and show that the smaller rule inherits the axiomatic properties of the larger rule. This is done, variously, by introducing dummy voters, by introducing dummy alternatives, and by copying voters.

Our first induction step reduces the number of voters. The underlying construction works by copying voters, and using the 'homogeneity' of the axioms of proportionality and strategyproofness. For the latter axiom, we use the fact that in the case $m = k + 1$, the preference extension of approval ballots to committees is *complete*, in that any two committees are comparable.

Lemma 7.4. *Suppose* $k \geqslant 2$ *and* $m = k + 1$, *and let* $q \geqslant 1$ *be an integer. If there exists a proportional and strategyproof committee rule for* $q \cdot k$ *voters, then there also exists such a rule for* k *voters.*

Proof. For convenience, we write profiles as lists. Given a profile P, we write qP for the profile obtained by concatenating q copies of P. Let f_{qk} be the rule for $q \cdot k$ voters. We define the rule f_k for k voters as follows:

$$f_k(P) = f_{qk}(qP) \quad \text{for all profiles } P \in \mathcal{B}^k.$$

Proportionality. Suppose $P \in \mathcal{B}^k$ is a party-list profile in which at least $\frac{n}{k} = \frac{k}{k} = 1$ voters approve $\{c\}$. Then qP is a party-list profile in which at least $q \cdot \frac{n}{k} = \frac{qn}{k} = q$ voters approve $\{c\}$. Since f_{qk} is proportional, $c \in f_{qk}(qP) = f_k(P)$.

Strategyproofness. Suppose for a contradiction that f_k is not strategyproof, so that there is P and an i-variant P' with $f_k(P') \cap P(i) \supsetneq f_k(P) \cap P(i)$. Because $m = k+1$, the committees $f_k(P')$ and $f_k(P)$ must differ in exactly 1 candidate. Since the manipulation was successful, $f_k(P')$ must be obtained by replacing a non-approved candidate in $f_k(P)$ by an approved one, say $f_k(P') = f_k(P) \cup \{c\} \setminus \{d\}$ with $c \in P(i) \not\ni d$. Now consider $f_{qk}(qP)$, and step-by-step let each of the q copies of $P(i)$ in qP manipulate from $P(i)$ to $P'(i)$ obtaining qP' in the last step. Because f_{qk} is strategyproof, at each step of this process f_{qk} cannot have exchanged a non-approved candidate by an approved candidate according to $P(i)$. This contradicts that $f_k(P') = f_k(P) \cup \{c\} \setminus \{d\}$. ☐

Our second induction step is the simplest: We reduce the number of alternatives using dummy candidates that no voter ever approves. This is the only place in the proof where we require the weak efficiency axiom.

Lemma 7.5. *Fix n and k, and let $m \geqslant k$. If there exists a weakly efficient, proportional, and strategyproof committee rule for $m+1$ alternatives, then there also exists such a rule for m alternatives.*

Proof. Let f_{m+1} be the committee rule defined on the candidate set $C_{m+1} = \{c_1, \dots, c_m, c_{m+1}\}$. Note that every profile P over candidate set $C_m = \{c_1, \dots, c_m\}$ is also a profile over candidate set C_{m+1}. We then just define the committee rule f_m for the candidate set C_m by $f_m(P) := f_{m+1}(P)$ for all profiles P over candidate set C_m, where we assume that $|\bigcup_{i\in N} P(i)| \geqslant k$. By weak efficiency, $f_m(P) \subseteq C_m$, so that f_m is a well-defined rule. It is easy to check that f_m is weakly efficient, proportional, and strategyproof. ☐

Our last induction step reduces the committee size from $k+1$ to k. The construction introduces an additional candidate and an additional voter, and appeals to Lemma 7.2 to show that the new candidate is always part of the winning committee. Thus, the larger rule implicitly contains a committee rule for size-k committees.

Lemma 7.6. *Let $k \geqslant 2$. If there exists a proportional and strategyproof committee rule for committee size $k+1$, for $k+1$ voters, and for $k+2$ alternatives, then there also exists such a rule for committee size k, for k voters, and for $k+1$ alternatives.*

Proof. Let f_{k+1} be the committee rule assumed to exist, defined on the candidate set $C_{k+2} = \{c_1, \dots, c_{k+2}\}$. We define the rule f_k for committee size k on candidate set $C_{k+1} = \{c_1, \dots, c_{k+1}\}$ as follows:

$$f_k(A_1, \dots, A_k) = f_{k+1}(A_1, \dots, A_k, \{c_{k+2}\}) \setminus \{c_{k+2}\},$$

for every profile $P = (A_1, \dots, A_k)$ over C_{k+1}. Notice that this is well-defined and returns a committee of size k, since by Lemma 7.2 applied to f_{k+1}, we always have $c_{k+2} \in f_{k+1}(A_1, \dots, A_k, \{c_{k+2}\})$.

Proportionality. Let $P = (A_1, \dots, A_k)$ be a party-list profile over C_{k+1}, in which the ballot $\{c\}$ occurs at least $\frac{n}{k} = \frac{k}{k} = 1$ time. Then $P' = (A_1, \dots, A_k, \{c_{k+2}\})$ is a party-list profile, in which $\{c\}$ occurs at least $\frac{n+1}{k+1} = \frac{k+1}{k+1} = 1$ time; thus, by proportionality of f_{k+1}, we have $c \in f_{k+1}(P') = f_k(P)$.

Strategyproofness. If there is a successful manipulation from P to P' for f_k, then there is a successful manipulation from $(P, \{c_{k+2}\})$ to $(P', \{c_{k+2}\})$ for f_{k+1}, contradiction. □

Finally, we can combine all three induction steps, applying them in order, and the base case, to get our main result.

Proof of the Main Theorem. Let $k \geqslant 3$, let n be divisible by k, and let $m \geqslant k+1$. Suppose for a contradiction that there does exist an approval-based committee rule f which satisfies weak efficiency, proportionality, and strategyproofness for these parameters.

By Lemma 7.5 applied repeatedly to f, there also exists such a rule f' for $k+1$ alternatives. By Lemma 7.4 applied to f', there exists a proportional and strategyproof rule f'' for k voters. By Lemma 7.6 applied to f'', there must exist a proportional and strategyproof rule for committee size 3, for 3 voters, and for 4 alternatives. But this contradicts Proposition 7.3. □

7.4.3. Extension to other electorate sizes

One drawback of Theorem 7.1 is the condition on the number of voters n. For larger values of k, practical elections are unlikely to have a number of voters which is exactly a multiple of k. The impossibility as we have proved it does not rule out that for other values of n, there does exist a proportional and strategyproof rule. Indeed, at least for small parameter values, the SAT solver confirms that this is the case. An important open question is whether, for fixed $k \geqslant 3$, the impossibility holds for all sufficiently large n.

In this section, we give one result to this effect, obtained by strengthening the proportionality axiom. Note that all the axioms we discussed in Section 7.3.2 are based on the intuition that a group of $\frac{n}{k}$ voters should be represented by one committee member. The value "$\frac{n}{k}$" is known as the *Hare quota.* An alternative proposal is the *Droop quota*, according to which every group consisting of strictly more than $\frac{n}{k+1}$ voters should be represented by one committee member. Thus, with Droop quotas, slightly smaller groups already need to be represented. The strengthened axiom is as follows.

Droop Proportionality Suppose P is any profile, and some singleton ballot $\{c\} \in \mathcal{B}$ appears strictly more than $\frac{n}{k+1}$ times in P. Then $c \in f(P)$.

Note that Droop proportionality applies to all profiles and not only party-list profiles. With this stronger proportionality axiom, we can show that for fixed k and *all* sufficiently large n, we have an incompatibility with strategyproofness.

Proposition 7.7. *Let $k \geqslant 3$, let $m \geqslant k+1$, and let $n \geqslant k^2$. Then there is no approval-based committee rule satisfying weak efficiency, strategyproofness, and Droop proportionality.*

Proof. Suppose such a rule f_n exists. By Lemma 7.5 (suitably reproved to apply to the Droop quota), there also is such a rule for $m = k+1$ alternatives, so we may assume that $m = k+1$.

Write $n = q \cdot k + r$ for some $0 \leqslant r < k$ and some $q \geqslant k$. We will show that there exists a committee rule for $q \cdot k$ voters which satisfies proportionality (with respect to the *Hare* quota) and strategyproofness, which contradicts Theorem 7.1.

Fix r arbitrary ballots $B_1, \ldots, B_r$. We define a committee rule f_{qk} on $q \cdot k$ voters, m alternatives, and for committee size k, as follows:

$$f_{qk}(A_1, \ldots, A_{qk}) = f_n(A_1, \ldots, A_{qk}, B_1, \ldots, B_r),$$

for all profiles $P = (A_1, \ldots, A_{qk}) \in \mathcal{B}^{qk}$.

It is clear that f_{qk} inherits strategyproofness from f_n: Any successful manipulation of f_{qk} is also successful for f_n.

We are left to show that f_{qk} satisfies (Hare) proportionality. So suppose that $P = (A_1, \dots, A_{qk}) \in \mathcal{B}^{qk}$ is a party-list profile in which singleton party $\{c\}$ is approved by at least $\frac{qk}{k} = q$ voters. Note that, because $r < k \leqslant q$,

$$\frac{n}{k+1} = \frac{qk+r}{k+1} < \frac{qk+q}{k+1} = \frac{q(k+1)}{k+1} = q,$$

Thus, in the profile $P' = (A_1, \dots, A_{qk}, B_1, \dots, B_r)$, there are strictly more than $\frac{n}{k+1}$ voters who approve $\{c\}$. Thus, by Droop proportionality, $c \in f_n(P') = f_{qk}(P)$. Thus, f_{qk} is (Hare) proportional. □

Remark 7.8. If we want to restrict the Droop proportionality axiom to only apply to party-list profiles, we can instead assume in Proposition 7.7 that $m \geqslant k+2$, and then let $B_1 = \cdots = B_r = \{c_{k+2}\}$, defining the rule f_{qk} only over the first $k+1$ alternatives. Then the final profile P' is a party-list profile.

7.5. Related Work

A short article by Duddy [2014b] also proves an impossibility about approval-based committee rules involving a proportionality axiom. Duddy's result is about *probabilistic* committee rules, which return probability distributions over the set of committees. Because any deterministic committee rule induces a probabilistic one (which puts probability 1 on the deterministic output), Duddy's probabilistic result also has implications for deterministic rules, which we can state as follows.

Theorem 7.9 (Duddy, 2014b)**.** *For $m = 3$ and $k = 2$, no approval-based committee rule f satisfies the following three axioms.*

1. *(Representative.) There exists a profile P in which n voters approve $\{x\}$ and $n+1$ voters approve $\{y, z\}$, but $f(P) \neq \{y, z\}$, for some $n \in \mathbb{N}$ and all distinct $x, y, z \in C$.*
2. *(Pareto-consistent.) If in profile P, the set of voters who approve of x is a strict subset of the set of voters who approve of y, then $f(P) \neq \{x, z\}$, for all distinct $x, y, z \in C$.*
3. *(Strategyproof.) Suppose profiles P and P' are identical, except that voter i approves $\{x, y\}$ in P but $\{x\}$ in P'. If $f(P) \neq \{x, y\}$, then also $f(P') \neq \{x, y\}$.*

How does Duddy's theorem relate to ours? Duddy's strategyproofness is weaker than but very similar to our strategyproofness. Our result does not require an efficiency axiom. Duddy's representative axiom is noticeably different from the proportionality axioms that we have discussed. Logically it is incomparable to our proportionality axiom; in spirit it may be slightly stronger. Note that not even the strongest of the proportionality axioms that we have discussed (i.e., EJR) imply Duddy's representativeness. It is also worth noting that Duddy's result works for smaller values of m and k than our result, suggesting that Duddy's axioms are stronger overall.

In computational social choice, there has been much recent interest in axiomatic questions in committee rules. Working in the context of strict orders, Elkind et al. [2017a] introduced several axioms and studied which committee rules satisfy them. Skowron et al. [2019] axiomatically characterise the class of *committee scoring rules*, and Faliszewski et al. [2016] study the finer structure of this class. For the approval-based setting, Lackner and Skowron [2018] characterise *committee counting rules*, and give characterisations of PAV and of Chamberlin–Courant. They also have a result suggesting that AV is the only consistent committee rule which is strategyproof.

From a computational complexity perspective, there have been several papers studying the complexity of manipulative attacks on multiwinner elections [Meir et al., 2008, Obraztsova et al., 2013, Faliszewski et al., 2017b, Aziz et al., 2015, Baumeister et al., 2015]. Other work has studied the complexity of evaluating various committee rules. Notably, it is NP-complete to find a winning committee for PAV [Aziz et al., 2015, Skowron et al., 2015].

7.6. Conclusions and Future Work

We have proved an impossibility about approval-based committee rules. The versions of the proportionality and strategyproofness axioms we used are very weak. It seems unlikely that, by weakening the axioms used, one can find a committee rule that exhibits satisfying versions of these requirements. A technical question which remains open is whether our impossibility holds for *all* numbers n of voters, no matter whether it is a multiple of k (see Section 7.4.3). It would also be interesting to study irresolute or probabilistic rules.

To circumvent the classic impossibilities of Arrow and Gibbard–Satterthwaite, it has proved very successful to study restricted domains such as single-peaked preferences, which can often give rise to strategyproof voting rules [Moulin, 1988a, Elkind et al., 2017b]. Elkind and Lackner [2015] propose analogues of single-peaked and single-crossing preferences for the case of approval ballots and dichotomous preferences. For example, a profile of approval ballots satisfies the Candidate Interval (CI) condition if there exists an underlying linear ordering of the candidates such that each voter approves an *interval* of candidates [see also Faliszewski et al., 2011]. Restricting the domain to CI profiles in our SAT encoding suggests that an impossibility of the type we have studied cannot be proved for this domain – at least for small values of n, m, and k. Finding a proportional committee rule that is not manipulable on the CI domain would be an exciting avenue for future work.

It would be interesting to obtain impossibilities using other axioms. Recently, Sánchez-Fernández and Fisteus [2019] found some incompatibilities between proportionality and *monotonicity*. Their version of proportionality ('perfect representation'), however, is very strong and possibly undesirable. It would be interesting to see whether such results hold for weaker versions of their axioms.

Thanks to Markus Brill for suggesting subset manipulations and other ideas, and to Martin Lackner and Piotr Skowron for useful discussions. Thanks to Boas Kluiving, Adriaan de Vries, Pepijn Vrijbergen, and Ulle Endriss for pointing out an error in a previous version.

8. Preferences Single-Peaked on Trees

A preference profile is single-peaked on a tree if the candidate set can be equipped with a tree structure such that the preferences of each voter are decreasing from their top candidate along all paths in the tree. We study the complexity of electing a committee under several variants of the Chamberlin–Courant rule when preferences are single-peaked on a tree. We first show that this problem can be solved in polynomial time for the egalitarian version of this problem, for arbitrary trees and scoring functions. For the standard utilitarian version of this problem, we prove that winner determination remains NP-hard for preferences single-peaked on a tree, even for the Borda scoring function. This hardness result holds even when the underlying tree has bounded pathwidth and bounded diameter. However, we provide algorithms whose running time is polynomial in the input size provided that either the number of leaves or the number of internal vertices of the underlying tree is bounded by a constant. To support these parameterised algorithms, we study the computational problem of finding a tree on which a given profile is single-peaked. We develop a structural approach that enables us to compactly represent all trees with respect to which a given profile is single-peaked. We show how to use this representation to efficiently find the best tree for a given profile for use with our winner determination algorithms: Given a profile, we can efficiently find a tree with a minimum number of leaves, or a tree with a minimum number of internal vertices among trees on which the profile is single-peaked. We also obtain positive results for additional optimisation criteria, but obtain NP-hardness for others.

8.1. Introduction

Computational social choice deals with algorithmic aspects of collective decision-making. One of the fundamental questions studied in this area is the complexity of determining the election winner(s) for voting rules: indeed, for a rule to be practically applicable, it has to be the case that we can find the winner of an election in a reasonable amount of time.

Most common rules that are designed to output a single winner admit polynomial-time winner determination algorithms; examples include such diverse rules as Plurality, Borda, Maximin, Copeland, and Bucklin (see, e.g., Arrow et al., 2002, for definitions). However, there are also some intuitively appealing single-winner rules for which winner determination is known to be computationally hard: this is the case, for instance, for Dodgson's rule [Bartholdi, III et al., 1989, Hemaspaandra et al., 1997], Young's rule [Rothe et al., 2003], and Kemeny's rule [Bartholdi, III et al., 1989, Hemaspaandra et al., 2005]. More recently, there has been much interest in the computational complexity of voting rules whose purpose is to elect a representative *committee* of candidates rather than select a single winner. While one can adapt common single-winner rules to this setting (e.g., appoint the candidates with the top k scores, where k is the target committee size, or split the voters into k districts and determine the winner in each district using a single-winner rule), this approach may result in a committee that does not reflect the true preferences of the electorate (see, e.g., Betzler et al., 2013). Therefore, it is preferable to use a voting system that is specifically designed for multiwinner elections.

One such system was proposed by Chamberlin and Courant [1983]. It is usually defined for the case where voter preferences are given as rankings (rather than approvals like in Chapter 7), and we will use rankings throughout this chapter. Given a committee $C' \subseteq C$ of k candidates, the system assumes that each voter i is *represented* by i's most-preferred candidate in C', that is, the member of C' ranked highest in i's preferences. Voter i is assumed to obtain utility from this representation. This utility is increasing in the rank of i's representative in i's preference ranking. For example, i's utility could be obtained as the Borda score of the representative, but any scoring function is possible. There is no constraint on the number of voters that can be represented by a single candidate; the assumption is that the committee will make its decisions by weighted voting, where the weight of each candidate is proportional to the fraction of the electorate that the candidate represents. Chamberlin and Courant's scheme outputs a committee of a fixed given size that maximises the the sum of voters' utilities according to some chosen scoring function (see Section 8.2 for a formal definition).[1] Recently, Betzler et al. [2013] suggested an egalitarian, or maximin, variant, where the quality of a committee is measured by the utility of the worst-off voter rather than total utility.

Unfortunately, the problem of identifying an optimal committee under the Chamberlin–Courant rule is known to be computationally hard, even for fairly simple scoring functions. In particular, Procaccia et al. [2008] show that this is the case for both schemes under r-approval scoring functions, where a voter i obtains utility 1 if i's representative is one of the r highest-ranked candidates, and utility 0 otherwise. Lu and Boutilier [2011] give an NP-hardness proof for the Chamberlin–Courant rule under the Borda scoring function (where the utility of a voter i is the number of candidates i ranks below i's representative). Betzler et al. [2013] extend these hardness results to the egalitarian variant.

Clearly, this is bad news if we want to use the Chamberlin–Courant rule in practice: elections may involve millions of voters and hundreds of candidates, and the election outcome needs to be announced soon after the votes have been cast. On the other hand, simply abandoning these voting rules in favour of easy-to-compute adaptations of single-winner rules is not acceptable if the goal is to select a truly representative committee. Thus, it is natural to try to circumvent the hardness results, either by designing efficient algorithms that compute an *approximately optimal* committee or by identifying reasonable assumptions on the structure of the election that ensure computational tractability. The former approach was pursued by Lu and Boutilier [2011], and by Skowron et al. [2015]. The latter approach was initiated by Betzler et al. [2013] who provide an extensive analysis of the fixed-parameter tractability of the winner determination problem under both utilitarian and egalitarian variants of the Chamberlin–Courant rule. They also investigate the complexity of this problem for *single-peaked electorates*.

An profile is said to be *single-peaked* [Black, 1948] if the set of candidates can be placed on a one-dimensional axis, such that a voter prefers candidates that are close to the voter's top choice on the axis. We can expect a profile to be single-peaked when every voter evaluates the candidates by their position on a numerical issue, such as the income tax rate or minimum wage level, or by their position on the left-right ideological axis. Many voting-related problems that are known to be computationally hard for general preferences become easy when voters' preferences are assumed to be single-peaked. This is the case for the winner determination problem under Dodgson's, Young's and Kemeny's rules [Brandt et al., 2015]. Betzler et al. [2013] show that this is also the case for winner determination of both the utilitarian and the egalitarian version of the Chamberlin–Courant rule.

Our Contribution The goal of this chapter is to explore whether the easiness results of Betzler et al. [2013] for single-peaked electorates can be extended to a more general class of elections.

[1] Monroe [1995] has subsequently proposed a variant of this scheme where the committee is assumed to use non-weighted voting, and, consequently, each member of the committee is required to represent approximately the same number of voters (up to a rounding error).

We focus on a generalisation of single-peaked preferences introduced by Demange [1982], which captures a much broader class of voters' preferences, while still implying the existence of a Condorcet winner. This is the class of preference profiles which are single-peaked *on a tree*. Informally, an election belongs to this class if we can construct a tree whose vertices are candidates in the election, and each voter has a most-preferred candidate and ranks all other candidates according to their perceived distance along this tree from the most-preferred candidate. A profile is single-peaked if and only if it is single-peaked on a path.

We focus on the Chamberlin–Courant rule. We first show that, for the egalitarian variant of this rule, winner determination is easy for an arbitrary scoring function when voters' preferences are single-peaked on a tree. Our proof proceeds by reducing our problem to an easy variant of the Hitting Set problem. For the utilitarian setting, we show that winner determination for the Chamberlin–Courant rule is NP-complete, even for the Borda scoring function. Hardness holds even if preferences are single-peaked on a tree of bounded diameter and bounded pathwidth. However, we present an efficient winner determination algorithm for preferences that are single-peaked on a tree with a *small number of leaves*: the running time of our algorithm is polynomial in the size of the profile, but exponential in the number of leaves (Section 8.5). Formally, the problem is in XP with respect to the number of leaves. Further, we give an algorithm that works for trees with a small number of *internal vertices* (i.e., with a *large* number of leaves) when using the Borda scoring function. This algorithm places the problem in FPT with respect to the committee size and the number of internal vertices.

Now, these parameterised algorithms assume that the tree with respect to which the preferences are single-peaked is given as an input. However, in practice we cannot expect this to be the case: typically, we are only given the voters' preferences and have to construct such a tree (if it exists) ourselves. While the algorithm of Betzler *et al.* faces the same issue (i.e., it needs to know the societal axis), there exist efficient algorithms for determining the societal axis given the voters' preferences [Bartholdi, III and Trick, 1986, Escoffier et al., 2008, Doignon and Falmagne, 1994]. In contrast, for trees the situation is more complicated. Trick [1989] describes a polynomial-time algorithm that decides whether there exists a tree such that a given election is single-peaked with respect to it, and constructs *some* such tree if this is indeed the case. However, Trick's algorithm leaves us a lot of freedom when constructing the tree; as a result, if the election is single-peaked with respect to several different trees, the output of Trick's algorithm will be dependent on the implementation details. In particular, there is no guarantee that an arbitrary implementation will find a tree that caters to the demands of the winner determination algorithms that we present: for example, the algorithm may return a tree with many leaves, while we wish to find one that has as few leaves as possible. Indeed, Trick's algorithm may output a complex tree even when the input profile is single-peaked on a line.

In Sections 8.7 and 8.8, we propose a general framework for finding trees with desired properties, and use it to obtain polynomial-time algorithms for identifying "nice" trees when they exist, for several appealing notions of "niceness". Specifically, we define a digraph that encodes, in a compact fashion, all trees with respect to which a given profile is single-peaked. This digraph enables us to count and/or enumerate all such trees. Moreover, we show that it has many useful structural properties. This can be exploited to efficiently find trees that minimise the number of leaves, or the number of internal vertices, or the degree or diameter or pathwidth among all trees with respect to which a given profile is single-peaked. These recognition algorithms complement our parameterised algorithms for winner determination. In contrast, we show that it is NP-complete to decide whether a profile is single-peaked on a tree which is isomorphic to a given tree.

8.2. Preliminaries

Let C be a finite set of *candidates*, and let $C!$ be the set of strict total orders over C. Let $N = \{1, \ldots, n\}$ be a set of *voters*. A *preference profile* $P : N \to C!$ assigns to each voter a preference order over C. When the profile P is clear from the context, for $i \in N$, we write $\succ_i$ for the preference order $P(i)$, and if $a \succ_i b$, then we say that voter i (strictly) prefers a to b.

Given a profile P, we denote by $\text{pos}(i, c)$ the position of candidate $c \in C$ in the preference order of voter $i \in N$:

$$\text{pos}(i, c) = |\{c' \in C : c' \succ_i c\}| + 1.$$

We write $\text{top}(i)$ for voter i's most-preferred candidate in position 1, we write $\text{second}(i)$ for the candidate in position 2, and $\text{bottom}(i)$ for i's least-preferred candidate in position m. Given a subset of candidates $C' \subseteq C$, we extend this notation and let $\text{top}(i, C')$, $\text{second}(i, C')$, and $\text{bottom}(i, C')$ denote voter i's most-, second-most- and least-preferred candidate in C', respectively, provided that $|C'| \geqslant 3$.

Given a subset $C' \subseteq C$, we write $P|_{C'}$ for the profile obtained from P by restricting the candidate set to C'.

Multiwinner Elections A *scoring function* for given N and C is a mapping $\mu : N \times C \to \mathbb{Z}$ such that $\text{pos}(i, c) < \text{pos}(i, c')$ implies $\mu(i, c) \geqslant \mu(i, c')$. Intuitively, $\mu(i, c)$ indicates how well candidate c represents voter i. A scoring function is said to be *positional* if there exists a vector $\mathbf{s} = (s_1, \ldots, s_m) \in \mathbb{Z}^m$ with $s_1 \geqslant s_2 \geqslant \ldots \geqslant s_m$ such that $\mu(i, c) = s_{\text{pos}(i,c)}$. We will say that the scoring functions are induced by the vector $\mathbf{s}$. We will usually take $\mathbf{s}$ such that $s_1 = 0$ and $s_2, \ldots, s_m \leqslant 0$, where negative values correspond to 'misrepresentation'. This choice will be without loss of generality, as applying a positive affine transformations to $\mathbf{s}$ will not change the output of the voting rules we introduce below. We will refer to the positional scoring function that corresponds to the vector $(0, -1, \ldots, -m + 1)$ as the *Borda scoring function*.

Given a preference profile P, a committee of candidates $C' \subseteq C$, and a scoring function $\mu : N \times C \to \mathbb{Z}$, we take voter i's utility from the committee C' to be $\mu(i, \text{top}(i, C'))$, that is, the score i gives to i's favourite candidate in C'. We also write

$$m_\mu^+(P, C') = \sum_{i \in N} \mu(i, \text{top}(i, C'))$$

for the sum of utilities of all voters (the *utilitarian Chamberlin–Courant score*), and

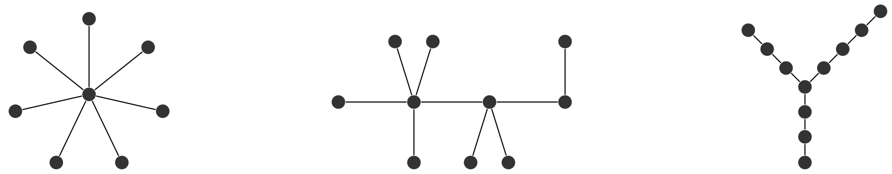
$$m_\mu^{\min}(P, C') = \min_{i \in N} \mu(i, \text{top}(i, C')).$$

for the utility obtained by the worst-off voter (the *egalitarian Chamberlin–Courant score*). Given a committee size $1 \leqslant k \leqslant |C|$, the *utilitarian Chamberlin–Courant rule* elects all committees $C' \subseteq C$ with $|C'| = k$ such that $m_\mu^+(P, C')$ is maximum. The *egalitarian* Chamberlin–Courant rule elects committees $C' \subseteq C$ with $|C'| = k$ such that $m_\mu^{\min}(P, C')$ is maximum. When referring to the Chamberlin–Courant rule, we will mean the utilitarian version by default.

To study the computation of winning committees under these rules, we now formally define the decision problems associated with their optimisation problems.

Definition 8.1. An instance of the UTILITARIAN CC (respectively, EGALITARIAN CC) problem is given by a preference profile P, a committee size $1 \leqslant k \leqslant |C|$, a scoring function $\mu : N \times C \to \mathbb{Z}$, and a bound $B \in \mathbb{Z}$. It is a "yes"-instance if there is a subset of candidates $C' \subseteq C$ with $|C'| = k$ such that $m_\mu^+(P, C') \geqslant B$ (respectively, $m_\mu^{\min}(P, C') \geqslant B$) and a "no"-instance otherwise.[2]

[2]Under our definition it may happen that some candidate in the committee does not represent any voter, i.e.,



(a) Star (b) Caterpillar (c) Subdivision of a star

Figure 8.1.: Examples of different classes of trees

We will sometimes consider the complexity of these problems for specific families of scoring functions. Note that a scoring function is defined for fixed C and N, so the question of asymptotic complexity makes sense for families of scoring functions (parameterised by C and N), but not for individual scoring functions. For instance, the Borda scoring function can be viewed as a family of scoring functions, as it is well-defined for any C and N.

Graphs and Digraphs A *digraph* $D = (C, A)$ is given by a set C of vertices and a set $A \subseteq C \times C$ of pairs, which we call *arcs*. If $(c, d) \in A$, we say that (c, d) is an *outgoing arc* of c. An *acyclic* digraph (a *dag*) is a digraph with no directed cycles. For a vertex $c \in C$, its *out-degree* $d^+(c) = |\{d \in C : (c, d) \in A\}|$ is the number of outgoing arcs of c. A *sink* is a vertex c with $d^+(c) = 0$, i.e., a vertex without outgoing arcs. It is easy to see that every dag has at least one sink. Given a digraph $D = (C, A)$, we write $\mathcal{G}(D)$ for the undirected graph (C, E) where for all $c, d \in C$, we have $\{c, d\} \in E$ if and only if $(c, d) \in A$ or $(d, c) \in A$. Thus, $\mathcal{G}(D)$ is the graph obtained from D when we forget about the orientations of the arcs of D.

Given a digraph $D = (C, A)$ and a set $C' \subseteq C$, we write $D|_{C'}$ for the induced subdigraph. Similarly, for a graph $G = (C, E)$, we write $G|_{C'}$ for the induced subgraph. We say that a set $C' \subseteq C$ is *connected* in a graph G if $G|_{C'}$ is connected.

Classes of trees Recall that a *tree* is a connected graph that has no cycles. A *leaf* of a tree is a vertex of degree 1. Vertices that are not leaves are *internal* vertices. A *path* is a tree that has exactly two leaves. A *star* $K_{1,n}$ is a tree that has exactly one internal vertex and n leaves; the internal vertex is called the *center* of the star. The *diameter* of a tree T is the maximum distance between two vertices of T; e.g., the diameter of a star is 2. A *caterpillar* is a tree in which every vertex is within distance 1 of a central path. A *subdivision of a star* is a tree obtained from a star by replacing its edges by paths. See Figure 8.1 for drawings of some examples.

Pathwidth The *pathwidth* of a tree T is a measure of how close T is to being a path. A *path decomposition* of $T = (C, E)$ is given by a sequence $S_1, \dots, S_r$ of subsets of C (called *bags*) such that

- for each edge $\{c, d\} \in E$, there is a bag S_i with $c, d \in S_i$, and
- for each $c \in C$, the bags containing c form an interval of the sequence, so that if $c \in S_i$ and $c \in S_j$ for $i < j$, then c is also a member of $S_{i+1}, S_{i+2}, \dots, S_{j-1}$.

there exists a $c' \in C'$ such that $c' \neq \text{top}(i, C')$ for all $i \in N$; equivalently, we allow for committees of size $k' < k$. It is assumed that the voting weight of such candidate in the resulting committee will be 0. This definition is also used, by e.g., [Cornaz et al., 2012, Skowron et al., 2015]. In contrast, Betzler et al. [2013] define the Chamberlin–Courant rule by explicitly specifying an assignment of voters to candidates, so that each candidate in C' has at least one voter who is assigned to it. The resulting voting rule is somewhat harder to analyse algorithmically. Note that when $|\{\text{top}(i, C) : i \in N\}| \geqslant k$, the two variants of the Chamberlin–Courant rule coincide.

The *width* of the path decomposition is $\max_{i\in[r]} |S_i| - 1$. The *pathwidth* of T is the minimum width of a path decomposition of T. For more on path- and treewidth, see, e.g., Bodlaender [1994].

Preferences That Are Single-Peaked on a Tree Consider a tree T with vertex set C. A preference profile P is said to be *single-peaked on* T [Demange, 1982] if for every voter $i \in N$, and for every candidate $c \in C$, if another candidate $c' \in C$ lies on the unique path from top(i) to c in T, then top(i) $\succ_i c' \succ_i c$. The profile P is said to be *single-peaked on a tree* if there exists a tree T with vertex set C such that P is single-peaked on T. The profile P is said to be *single-peaked* if P is single-peaked on some tree T that is a path. Let us collect two simple properties here. The proof is straightforward from the definitions.

Proposition 8.2. *Let P be a preference profile and T be a tree on vertex set C. The following are equivalent:*

- *P is single-peaked on T.*
- *For every $C' \subseteq C$ that is connected in T, $P|_{C'}$ is single-peaked on $T|_{C'}$.*
- *For every $i \in N$ and every $c \in C$, the* top-initial segment $\{d \in C : d \succ_i c\}$ *is connected in C'.*

Given a profile P, we denote the set of all trees T such that P is single-peaked on T by $\mathcal{T}(P)$.

8.3. Egalitarian Chamberlin–Courant on Arbitrary Trees

We start by presenting a simple algorithm for finding a winning committee under the egalitarian Chamberlin–Courant rule that works for preferences single-peaked on arbitrary trees. Our algorithm proceeds by finding a committee of minimum size that satisfies a given worst-case utility bound.

First, we show that the winner determination problem in the egalitarian case can be reduced to the following variant of the Hitting Set problem, where the ground set is the vertex set of a tree T, and we need to hit a collection of connected subsets of T.

Definition 8.3. An instance of the Tree Hitting Set problem is given by a tree T on vertex set C, a family $\mathcal{C} = \{C_1, \dots, C_n\}$ of subsets of C such that each C_i is connected in T, and a target cover size $k \in \mathbb{Z}_+$. It is a “yes”-instance if there is a subset of vertices $C' \subseteq C$ with $|C'| \leqslant k$ such that $C' \cap C_i \neq \varnothing$ for $i = 1, \dots, n$, and a “no”-instance otherwise.

Guo and Niedermeier [2006] show that the Tree Hitting Set problem can be solved in polynomial time. Since they consider a dual formulation (in terms of set cover), we present an adaptation of the short argument here.

Theorem 8.4 (Guo and Niedermeier, 2006)**.** Tree Hitting Set *can be solved in polynomial time.*

Proof. Consider a vertex $c \in C$ that is a leaf of T, and let $d \in C$ be the (unique) vertex that c is adjacent to. Suppose that $c \in C_i$ for some i. Then, because C_i is a connected subset of T, we either have $C_i = \{c\}$ or $d \in C_i$.

With this observation, we can now give a simple algorithm that constructs a minimum hitting set: Consider a leaf vertex $c \in C$ adjacent to $d \in C$. If there exists some $C_i \in \mathcal{C}$ with $C_i = \{c\}$, then any hitting set must include c, so add c to the hitting set under construction, remove c from T and remove all copies of $\{c\}$ from $\mathcal{C}$. Otherwise, every set C_i that would be hit by c is also hit by d, so any hitting set including c remains a hitting set when c is replaced by d. Hence, it is safe to delete c from T and from each $C_i \in \mathcal{C}$. Now repeat the process on the smaller instance constructed. Once all vertices have been deleted, return the constructed hitting set, which is minimum by our argument. □

Now we show how to reduce our winner determination problem to the hitting set problem. Suppose we are given an instance of the Egalitarian CC problem, consisting of a profile P, a tree T on which P is single-peaked, a target committee size k, and the bound B. We construct a Tree Hitting Set instance as follows: The ground set is the candidate set C, the tree T is the tree with respect to which voters' preferences are single-peaked, and the target cover size equals the committee size k. For each $i \in N = \{1, \dots, n\}$, construct the set $C_i = \{c \in C : \mu(i, c) \geqslant B\}$. Since μ is monotone, the set C_i is a top-initial segment of i's preference order, i.e., is of the form $\{c \in C : c \succ_i d\}$ for some $d \in C$. By Proposition 8.2, since P is single-peaked on T, each set C_i is connected in T, so we have constructed a legal instance of Tree Hitting Set. Now note that, for every set $C' \in C$,

$$m_\mu^{\max}(P, C') \geqslant B \text{ if and only if } C' \cap C_i = C' \cap \{c \in C : \mu(i, c) \geqslant B\} \neq \varnothing \text{ for all } i.$$

It follows that our reduction is correct.

Using this reduction and the algorithm for Tree Hitting Set, we can solve Egalitarian CC in polynomial time.

Theorem 8.5. *For profiles that are single-peaked on a tree, we can find a winning committee under the egalitarian Chamberlin–Courant rule in polynomial time.*

8.4. Hardness of Utilitarian Chamberlin–Courant on Arbitrary Trees

For preferences single-peaked on a *path*, the utilitarian version of the Chamberlin–Courant rule becomes easy to compute [Betzler et al., 2013], using a dynamic programming algorithm. While we are able to generalise this algorithm to work for some other trees (see Section 8.5), it is not clear how to extend it to arbitrary trees. Indeed, here we show that the utilitarian Chamberlin–Courant rule remains NP-complete for preferences single-peaked on a tree. Hardness holds even for the Borda scoring function, and even for trees that have diameter 4 and pathwidth 2.

Theorem 8.6. Utilitarian CC *is NP-complete, even for the Borda scoring function.*

Proof. We will reduce a restricted version of Exact Cover by 3-Sets (X3C) to Utilitarian CC. Recall that an instance of X3C is given by a ground set $X = \{x_1, \dots, x_p\}$ with $p = 3p'$ for some $p' \in \mathbb{Z}_+$ and a collection $\mathcal{Y} = \{Y_1, \dots, Y_q\}$ of 3-element subsets of X; it is a "yes"-instance if we can pick a subcollection $\mathcal{Y}' \subseteq \mathcal{Y}$ of size p' that covers X, i.e., for each $x_i \in X$ there exists a Y_j in $\mathcal{Y}'$ such that $x_i \in Y_j$. This problem is known to be NP-hard even if each element of X appears in at most three sets in $\mathcal{Y}$ [Garey and Johnson, 1979].

Given an instance $(X, \mathcal{Y})$ of X3C such that $|\{Y_j \in \mathcal{Y} : x_i \in Y_j\}| \leqslant 3$ for each $x_i \in X$, we construct an instance of Utilitarian CC as follows. We let M be a large number ($M = 5pq$ will do).

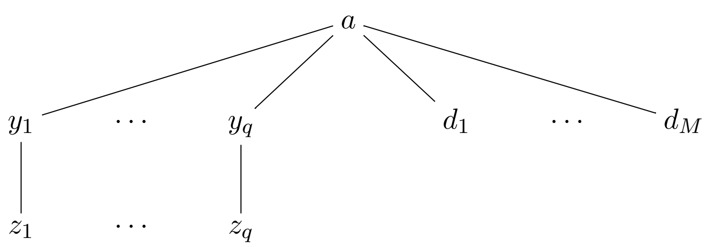
We introduce a candidate a, two candidates y_j and z_j for each set $Y_j \in \mathcal{Y}$, and M dummy candidates. Formally, we set $Y = \{y_1, \dots, y_q\}$, $Y = \{z_1, \dots, z_q\}$, $D = \{d_1, \dots, d_M\}$, and take the candidate set $C = \{a\} \cup Y \cup Z \cup D$.

We now introduce the voters, who will come in three types $N = N_1 \cup N_2 \cup N_3$.

N_1			N_2			N_3		
$5p$	$\cdots$	$5p$	1	$\cdots$	1	M	$\cdots$	M
y_1		y_q	a		a	z_1		z_q
z_1		z_q	$y_{j_{1,1}}$		$y_{j_{p,1}}$	y_1		y_q
a		a	$y_{j_{1,2}}$		$y_{j_{p,2}}$	a		a
$\vdots$		$\vdots$	$y_{j_{1,3}}$		$y_{j_{p,3}}$	$\vdots$		$\vdots$
			d_1		d_1			
			$\vdots$		$\vdots$			
			d_M		d_M			
			$\vdots$		$\vdots$			

- The set N_1 consists of $5p$ identical voters for each $Y_j \in \mathcal{Y}$: they rank y_j first, z_j second, and a third, followed by all other candidates as specified below. (The purpose of these voters will be to populate good committees with representatives y_j of sets $Y_j \in \mathcal{Y}$.)
- The set N_2 consists of 1 voter v_{x_i} for each element $x_i \in X$: v_{x_i} ranks a first, followed by the candidates y_j such that $x_i \in Y_j$ in some order, followed by the dummy candidates $d_1, \ldots, d_M$ in some order, followed by all other candidates as specified below. (The purpose of these voters will be to ensure that every element is covered by one of the sets represented by a y_j in the committee, and to incur a heavy penalty of M if the element is uncovered.)
- The set N_3 is a set of M identical voters for each $Y_j \in \mathcal{Y}$ who all rank z_j first, y_j second, and a third, followed by all other candidates as specified below. (The purpose of these voters is to force any good committee to include *all* the z_j candidates.)

The constructed profile is supposed to be single-peaked on the following tree:



This tree is obtained by starting with a star with center a and leaves $Z \cup D$ and then attaching y_j as a leaf onto z_j for every $j = 1, \ldots, q$. It is easy to see that it has diameter 4 and pathwidth 2. To ensure that the profile is single-peaked on this tree, we need to specify how to order the "all other candidates" in each vote. Inspecting the tree, an arbitrary order of these candidates will do, provided that for each j, we rank y_j above z_j.

This completes the construction of the profile P with voter set N and candidate set C. Next, set the target committee size to be $k = p' + q$ and the target bound to be $B = -(5p)(q - p') - 3p$ (note that by construction, $-M < B$). Intuitively, the "correct committee" we have in mind consists of all z_j candidates (of which there are q), and of a selection of y_j candidates that form an exact cover (of which there should be p' many), should there exist an exact cover. This completes the description of our instance of the Utilitarian CC problem with the Borda scoring function with $\mathbf{s} = (0, -1, -2, \ldots)$. Now let us prove that the reduction is correct.

Suppose we have started with a "yes"-instance of X3C, and let $\mathcal{Y}'$ be a collection of p' many subsets that cover X. Consider the committee $C' = Z \cup \{y_j : Y_j \in \mathcal{Y}'\}$; note that $|C'| = p' + q = k$. The voters in N_3 and $(5p)p'$ voters in N_1 have their most-preferred candidate in C', so they obtain a Borda score of 0. For the remaining $(5p)(q - p')$ voters in N_1, their score under C' is -1, since $z_j \in C'$ for all j. Further, each voter $i \in N_2$ obtains a score of at least -3: Suppose i corresponds to the element x_i. Then i ranks the candidates y_j such that $x_i \in Y_j$ in positions 2, 3, and 4. Since $\mathcal{Y}'$ is a cover of X, at least one of these candidates appears in C', and so i obtains a Borda score of at least -3. We conclude that $m_\mu^+(P, C') \geqslant -(5p)(q - p') - 3p = B$.

Conversely, suppose there exists a committee C' of size $k = p' + q$ with $m_\mu^+(P, C') \geqslant B$. Note first that C' has to contain z_j for each j: otherwise, there are M voters in N_3 with utility at most -1, and then the utilitarian Chamberlin–Courant score of C' is at most $-M < B$. Thus $Z \subseteq C'$. We will now argue that $C' \setminus Z$ is a subset of Y, and that $\mathcal{Y}'' = \{Y_j : y_j \in C' \setminus Z\}$ is an exact cover of X. Suppose that $C' \setminus \{z\}$ contains too few candidates from Y, i.e., at most $p' - 1$ candidates from Y. Then N_1 contains at least $(5p)(q - (p' - 1))$ voters whose score under C' is at most -1, so $m_\mu^+(P, C') \leqslant -(5p)(q - p' + 1) < -(5p)(q - p') - 3p = B$, a contradiction. Thus, we have $C' \setminus Z \subseteq Y$. Now, suppose that $\mathcal{Y}''$ is not an exact cover of X. Let x_i be an element of X that is not covered by $\mathcal{Y}''$, and consider the voter $i \in N_2$ corresponding to x_i. Clearly, none of the candidates ranked in positions $1, \dots, M + 5$ by this voter appear in C'. Thus, the score of this voter under C' is less than $-M$, so the total score of C' is less than $-M < B$, a contradiction. Thus, a "yes"-instance of Utilitarian CC corresponds to a "yes"-instance of X3C. □

8.5. Utilitarian Chamberlin–Courant on Trees with Few Leaves

The above hardness result shows that single-peakedness on trees is not a strong enough assumption to make our multiwinner elections tractable. However, if we place further constraints on the shape of the underlying tree, we may be able to achieve tractability.

In this section, we present an algorithm for utilitarian Chamberlin–Courant whose running time is polynomial for any profile that is single-peaked on a tree with a constant number of leaves. The algorithm proceeds by dynamic programming; it can be seen as a generalisation of the algorithm due to Betzler et al. [2013] for preferences single-peaked on a path, i.e., a tree with two leaves.

Theorem 8.7. *Given a profile P with $|C| = m$ and $|N| = n$ and a tree T with λ leaves such that P is single-peaked on T, we can find a winning committee for the utilitarian Chamberlin–Courant rule in time* $\mathrm{poly}(n, m^\lambda, k^\lambda)$, *where k is the target committee size.*

Proof. We use dynamic programming to find a committee of size k that maximises the utilitarian Chamberlin–Courant score.

We pick an arbitrary vertex r^* to be the root of T. This choice induces a partial order $\succ$ on C: we set $a \succ b$ if a lies on the (unique) path from r^* to b in T. Thus, $r^* \succ a$ for every $a \in C \setminus \{r^*\}$. A set $A \subseteq C$ is said to be an *anti-chain* if no two elements of A are comparable with respect to $\succ$. See Figure 8.2 on the right for an example; if we added the left child of r^* to the set, it would not be an anti-chain anymore. Observe that for every committee $C' \subseteq C$, its set of maximal elements with respect to $\succ$ forms an anti-chain. Note also that if a and b belong to an anti-chain $A \subseteq C$ and c is a leaf of T, then it cannot be the case that both a and b are ancestors of c, and hence $|A| \leqslant \lambda$.

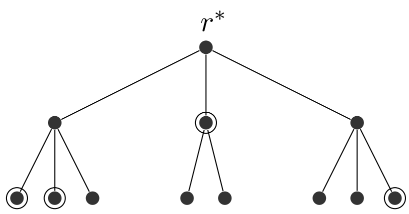


Figure 8.2.: An anti-chain

Given a vertex r, let T_r be the subtree of T rooted at r. The vertex set of T_r is $C_r = \{r\} \cup \{c : r \succ c\}$. Let $N_r = \{i \in N : \text{top}(i) \in C_r\}$ be the set of all voters whose most-preferred candidate belongs to C_r. Let P_r be the profile obtained from P by restricting the candidate set to C_r and the voter set to N_r. For each $r \in C$ and each $\ell = 1, \dots, k$ let

$$M(r, \ell) = \max \{m^+_\mu(P_r, C') : C' \subseteq C_r \text{ with } |C_r| = \ell \text{ and } r \in C'\}$$

be the highest Chamberlin–Courant score obtainable in P_r by a committee from C_r of size at most ℓ, subject to r being selected.

Suppose that we have computed these quantities for all descendants of r; we will now explain how to compute them for r. Let $C' \subseteq C_r$ be an optimal committee of size ℓ including r for P_r, so that $m^+_\mu(P_r, C') = M(r, \ell)$. Let $A = \{r_1, \dots, r_s\}$ be the set of maximal elements of $C' \setminus \{r\}$ with respect to $\succ$ and let $\ell_j = |C' \cap C_{r_j}|$ for $j = 1, \dots, s$; we have $\ell_1 + \dots + \ell_s = \ell - 1$. For each j, each voter in N_{r_j} is better represented by r_j than by any candidate not in C_{r_j} by single-peakedness. Thus, the contribution of voters in N_{r_j} to the total score $M(r, \ell)$ of C' is given by $m^+_\mu(P_{r_j}, C' \cap C_{r_j})$. In fact, this quantity must equal $M(r_j, \ell_j)$, since otherwise we could replace the candidates in $C' \cap C_{r_j}$ by the optimiser of $M(r_j, \ell_j)$ and increase the objective score of C', which would be a contradiction. On the other hand, consider a voter i in $N_r \setminus (N_{r_1} \cup \dots \cup N_{r_s})$. Voter i's most-preferred candidate in C' must be one of $r, r_1, \dots, r_s$: for each $j = 1, \dots, s$, candidate r_j is a better representative for i than any other candidate in C_{r_j} by single-peakedness.

This suggests the following procedure for computing $M(r, \ell)$. Let $\mathcal{T}_r$ be the set of all anti-chains in T_r. An *ℓ-division scheme* for an anti-chain $A = \{r_1, \dots, r_s\} \in \mathcal{T}_r$ is a list $L = (\ell_1, \dots, \ell_s)$ such that $\ell_j \geqslant 1$ for all $j = 1, \dots, s$ and $\ell_1 + \dots + \ell_s = \ell$. We denote by $\mathcal{L}^A_\ell$ the set of all ℓ-division schemes for A. Now, for every anti-chain $A = \{r_1, \dots, r_s\} \in \mathcal{T}_r \setminus \{\{r\}\}$ and every ℓ-division scheme $L = (\ell_1, \dots, \ell_s) \in \mathcal{L}^A_{\ell-1}$, we set $N'_r = N_r \setminus (N_{r_1} \cup \dots \cup N_{r_s})$ and

$$M(A, L) = \sum_{j=1}^{s} M(r_j, \ell_j) + \sum_{i \in N'_r} \mu(i, \text{top}(i, A \cup \{r\})).$$

We then have $M(r, \ell) = \max_{A \in \mathcal{T}_r \setminus \{\{r\}\}, L \in \mathcal{L}^A_{\ell-1}} M(A, L)$, where we maximise over all anti-chains in T_r except $\{r\}$ and over all ways of dividing the $\ell - 1$ slots among the elements of the anti-chain. The base case for this recurrence corresponds to the case when r is a leaf, and is easy to deal with.

The final answer depends on whether the root r^* is part of the optimal Chamberlin–Courant committee. If r^* is selected, then the optimum Chamberlin–Courant committee has objective score $M(r^*, k)$. If r^* is not selected, then we need to maximise over all anti-chains $A = \{r_1, \dots, r_s\} \in \mathcal{T}_{r^*} \setminus \{\{r^*\}\}$ and over all ways of dividing the k slots $L = (\ell_1, \dots, \ell_s) \in \mathcal{L}^A_k$. That is, we set $N'_{r^*} = N_{r^*} \setminus (N_{r_1} \cup \dots \cup N_{r_s})$ and

$$M'(A, L) = \sum_{j=1}^{s} M(r_j, \ell_j) + \sum_{i \in N'_{r^*}} \mu(i, \text{top}(i, A)).$$

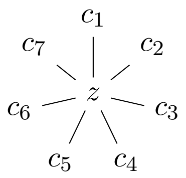
Note that r^* does not appear in the second term. The total Chamberlin–Courant score is given by $\max\{M(r^*, k), \max_{A \in \mathcal{T}_{r^*} \setminus \{\{r^*\}\}, L \in \mathcal{L}^A_k} M'(A, L)\}$.

We have argued that the size of each anti-chain is at most λ. Therefore, to calculate each $M(r, \ell)$, we enumerate at most m^λ anti-chains and at most k^λ divisions. This establishes our bound on the running time. □

Notice that the time bound of our algorithm implies that the problem is in XP with respect to the number λ of leaves in the underlying tree. Whether there is an FPT algorithm for this parameter, or even for the combined parameter (k, λ), is open.

8.6. Utilitarian Chamberlin–Courant on Trees with Few Internal Vertices

Consider the star with center alternative z and leaf alternatives $c_1, \dots, c_7$. Which preference orders are single-peaked on this tree?



Let us think about the possible top alternatives. A ranking could begin with z. After z, we can rank the other alternatives in arbitrary order without violating single-peakedness. But suppose we begin the ranking with a leaf alternative such as c_1. Then z must be the second alternative, because the set of first and second alternative must be connected in the tree. After ranking c_1 and z, we can then order the remaining alternatives arbitrarily without violating single-peakedness. Thus, precisely the orders in which the center vertex is in first or second position are single-peaked on the star.

Proposition 8.8. *A preference profile is single-peaked on a star if and only if there exists a candidate that every voter ranks in first or second position.*

This observation implies that, in some sense, the restriction of being single-peaked on a tree does not give us much information. For example, consider the problem to compute an optimal Kemeny ranking. This problem is NP-complete in general [Bartholdi, III et al., 1989], and we can easily see that it remains hard for preferences single-peaked on a star: reducing from the general problem, we can just add a new alternative that every voter ranks in first position; this new profile is single-peaked on a star.

For some other problems, though, the restriction to stars makes the problem easy. In particular, this is the case for the utilitarian Chamberlin–Courant rule with the Borda scoring function. To see this, note that it will often be a good idea to include the candidate who is the center vertex of the star in the committee. Once we have done so, every voter is already pretty well represented: the Borda score of each voter's representative is either 0 or -1. So we now only need to identify $k-1$ candidates whose inclusion in the committee would bring the score of as many voters as possible up to 0, which amounts to simply selecting $k-1$ candidates with highest plurality scores. Finally, we need to consider the case where the optimum committee does not include the center vertex, but one can check that this can only be the committee consisting of the k candidates with highest plurality scores (see the proof of Theorem 8.9 below). By selecting the better of the two committees produced, we find a winning committee.

The algorithm we have sketched for the Borda scoring function on stars can be generalised to work for trees which have a small number of internal vertices (and thus a large number of leaves). While above we guessed whether the center vertex would be part of the winning committee or not, we now have to guess for *each* internal vertex whether it will be part of the committee.

Theorem 8.9. *Given a profile P with $|C| = m$ and $|N| = n$ and a tree $T \in \mathcal{T}(P)$ with η internal vertices such that P is single-peaked on T, and a target committee size $k \geqslant 1$, we can find a winning committee of size k for P under the Chamberlin–Courant rule with the Borda scoring function in time* $\mathrm{poly}(n, m, (k+1)^\eta)$.

Proof. Given a candidate $c \in C$, let $\mathrm{plu}(c) = |\{i \in N : \mathrm{top}(i) = c\}|$ be the number of voters in P that rank c first. Let C° be the set of internal vertices of T. For each candidate $c \in C^\circ$, let $\mathrm{lvs}(c)$ denote the set of leaf candidates in $C \setminus C^\circ$ that are adjacent to c in T.

Our algorithm proceeds as follows. For each candidate $c \in C^\circ$ it guesses a pair $(b(c), \ell(c))$, where $b(c) \in \{0, 1\}$ and $0 \leqslant \ell(c) \leqslant |\text{lvs}(c)|$. The component $b(c)$ indicates whether c itself is in the committee, and $\ell(c)$ indicates how many candidates in $\text{lvs}(c)$ are in the committee. We require $\sum_{c \in C^\circ}(b(c) + \ell(c)) = k$. Next, the algorithm sets $C' = \{c \in C^\circ : b(c) = 1\}$, and then for each $c \in C^\circ$ it orders the candidates in $\text{lvs}(c)$ in non-increasing order of $\text{plu}(c)$ (breaking ties according to a fixed ordering $\rhd$ over C), and adds the first $\ell(c)$ candidates in this order to C'.

Each guess corresponds to a committee of size k. Guessing can be implemented deterministically: consider all options for the pair $\{(b(c), \ell(c))\}_{c \in C^\circ}$ satisfying $\sum_{c \in C^\circ}(b(c) + \ell(c)) = k$ (there are at most $2^\eta \cdot (k+1)^\eta$ possibilities), compute the Chamberlin–Courant score of the resulting committee for each option, and output the best one.

It remains to argue that this algorithm finds a committee with the maximum Chamberlin–Courant score. To see this, let $\mathcal{S}$ be the set of all size-k committees with the maximum Chamberlin–Courant score, and pick a committee S^* from $\arg\max_{C' \in \mathcal{S}} |C' \cap C^\circ|$. When picking S^*, we may break ties according to $\rhd$, so that there is no set $S \in \arg\max_{C' \in \mathcal{S}} |C' \cap C^\circ|$ such that $S^* \setminus S = \{c\}$, $S \setminus S^* = \{c'\}$ and $c' \rhd c$.

For each $c \in C^\circ$, let $b^*(c) = 1$ if $c \in S^*$ and $b^*(c) = 0$ otherwise, and let $\ell^*(c) = |\text{lvs}(c) \cap S^*|$. Our algorithm will consider the pair $\{(b^*(c), \ell^*(c))\}_{c \in C^\circ}$ at some point, and construct a committee S based on this pair. We will now argue that $S = S^*$. This shows correctness of our algorithm, since it will return a committee with a total score at least as high as that of S.

Clearly, we have $C^\circ \cap S = C^\circ \cap S^*$, so it remains to argue that $\text{lvs}(c) \cap S^* = \text{lvs}(c) \cap S$ for each $c \in C^\circ$. Suppose for the sake of contradiction that this is not the case, i.e., there exists a $c \in C^\circ$ and a pair of candidates $c', c'' \in \text{lvs}(c)$ with $c' \in S \setminus S^*$ and $c'' \in S^* \setminus S$. We distinguish two cases: $c \in S^*$ or $c \notin S^*$.

If $c \in S^*$, consider the committee $S' = (S^* \setminus \{c''\}) \cup \{c'\}$. We claim that S' has the same Chamberlin–Courant score as S^*. Note that when moving from S^* to S',

- the score obtained by the $\text{plu}(c'')$ voters who rank c'' first changes from 0 to -1,
- the score obtained by the $\text{plu}(c')$ voters who rank c' first changes from -1 to 0,
- the score of all other voters is unaffected by the change, since they prefer $c \in S^* \cap S'$ to both c' and c''.

We also have $\text{plu}(c') \geqslant \text{plu}(c'')$ by construction of S, and so the score of S' is at least the score of S^*, and hence they must be equal. But by construction of S, we have $c' \rhd c''$, and this contradicts our choice of S^* from $\arg\max_{C' \in \mathcal{S}} |C' \cap C^\circ|$.

Now, suppose that $c \notin S^*$. Consider the committee $S' = (S^* \setminus \{c''\}) \cup \{c\}$. Again, we claim that S' has the same Chamberlin–Courant score as S^*. Note that when moving from S^* to S',

- we have decreased the score of each of the $\text{plu}(c'')$ voters who rank c'' first by 1 (as all of them rank c second),
- we have increased the score of each of the $\text{plu}(c')$ voters who rank c' first by at least 1 (as all of them rank c second),
- and we do not decrease the score of any other voter (as all of them prefer c to c'').

Again, we have $\text{plu}(c') \geqslant \text{plu}(c'')$ by construction of S, and so the score of S' is at least the score of S^*, and hence they must be equal. Thus, the Chamberlin–Courant score of S' is optimal, and so $S' \in \mathcal{S}$. But $|S' \cap C^\circ| > |S^* \cap C^\circ|$, which contradicts our choice of S^* from $\arg\max_{C' \in \mathcal{S}} |C' \cap C^\circ|$. □

It is clear from our proof that Theorem 8.9 holds for every positional scoring function whose score vector satisfies $s_1 = 0$, $s_2 = -1$, $s_3 \leqslant -2$. Observe also that our algorithm is in FPT with respect to the combined parameter (k, η); in contrast, for general preferences computing the Chamberlin–Courant winners is W[2]-hard with respect to k even under the Borda scoring function [Betzler et al., 2013]. Our previous algorithm for trees with few leaves is in XP with respect to the number of leaves λ, but is not in FPT with respect to λ or even (k, λ).

8.7. The Attachment Digraph

We now move on from our study of multiwinner elections and turn towards the problem of recognising when a given preference profile is single-peaked on a tree. In particular, for each profile P, we will study the collection $\mathcal{T}(P)$ of *all* trees on which P is single-peaked. It turns out that the set $\mathcal{T}(P)$ has a lot of structure and admits a concise representation. In many cases, this will allow us to pick a "nice" tree from $\mathcal{T}(P)$ that satisfies certain additional requirements. For example, to use the algorithm from Section 8.5, we would want to pick the tree from $\mathcal{T}(P)$ with the fewest leaves, and to use the algorithm from Section 8.6, we would want to use the tree with the fewest internal vertices.

Trick [1989] has presented an algorithm that decides whether $\mathcal{T}(P)$ is non-empty. If so, the algorithm produces some tree T with $T \in \mathcal{T}(P)$. While building the tree, the algorithm makes various arbitrary choices. In our approach, we will store all the choices that the algorithm could take; we introduce a data structure which we call the *attachment digraph* of profile P which encapsulates all the choices recorded.

We will start by giving a high-level description of Trick's algorithm; the discussion follows Trick's paper closely. We first take inspiration from algorithms for recognising preferences that are single-peaked on a line. They typically start out by noticing that an alternative that is ranked bottom-most by some voter must be placed at one of the ends of the axis. Trick's algorithm uses the same idea; the analogue for trees is as follows.

Proposition 8.10. *Suppose P is single-peaked on T, and suppose c occurs as a bottom-most alternative, that is,* $\text{bottom}(i) = c$ *for some $i \in N$. Then c is a leaf of T.*

Proof. The set $A \setminus \{c\}$ is a top-initial segment of v_i and, hence, must be connected in T. This can only be the case if c is a leaf of T. □

Suppose we have identified a bottom-ranked alternative c; we deduce that if our profile is single-peaked on any tree T, then c is a leaf of T. Now, being a leaf, c must have exactly one neighbouring vertex b. Which vertex could this be? The following simple observation gives some necessary conditions.

Proposition 8.11. *Suppose P is single-peaked on T, and suppose $c \in C$ is a leaf of T, adjacent to $b \in C$. Let $i \in N$ be a voter. Then either*

(i) $b \succ_i c$, *or*

(ii) $c = \text{top}(i)$ *and* $b = \text{second}(i)$.

Proof. (i) Suppose first that c is not i's top-ranked alternative, and rather $\text{top}(i) = a$. Take the unique path in T from a to c, which passes through b since b is the only neighbour of c. Since i's vote is single-peaked on T, it is single-peaked on this path, and hence i's preference decreases along it from a to c. Since b is visited before c, it follows that $b \succ_i c$.

(ii) Suppose, otherwise, that c is i's top-ranked alternative. Then $\{c, \text{second}(i)\}$ is a top-initial segment of i's vote, which by Proposition 8.2 is a connected set in T, and hence forms an edge. Thus, c is adjacent to $\text{second}(i)$, so $\text{second}(i) = b$ as required. □

Thus, in our search for a neighbour of the leaf c, we can restrict our attention to those alternatives b which for each voter i satisfy either (i) or (ii) in the proposition above. Let us write this down more formally: For each $c \in C$ and $i \in N$, define

$$B(i,c) = \begin{cases} \{c' \in C : c' \succ_i c\} & \text{if } \mathrm{top}(i) \neq c, \\ \{\mathrm{second}(i)\} & \text{if } \mathrm{top}(i) = c. \end{cases}$$

Applying Proposition 8.11 to all voters i gives us the following constraint for our choice of b.

Corollary 8.12. *Suppose a profile is single-peaked on T, and $c \in C$ is a leaf of T. Then c must be adjacent to an element of $B(c) := \bigcap_{i \in N} B(i,c)$.*

We have established that it is necessary for leaf c to be adjacent to some alternative of $B(c)$. It turns out that if the profile is single-peaked on a tree, then for *any* of alternative $b \in B(c)$, there is some tree $T \in \mathcal{T}(P)$ in which c is adjacent to b.

Proposition 8.13. *Let P be a profile in which c occurs bottom-ranked. Suppose that $P|_{C \setminus \{c\}}$ is single-peaked on some tree T_{-c}, and let T be a tree obtained from T_{-c} by attaching c as a leaf adjacent to some element $b \in B(c)$. Then P is single-peaked on T.*

Proof. Let T be a tree obtained as described. We show that P is single-peaked on T. Suppose $C' \subseteq C$ is a top-initial segment of the ranking of some voter i in P. We need to show that C' is connected in T.

- If $c \notin C'$, then C' is connected in T_{-c} because $P|_{C \setminus \{c\}}$ is single-peaked on T_{-c}. Hence C' is also connected in T.
- If $C' = \{c\}$, then C' is trivially connected in T.
- If $c \in C'$ and $C' \neq \{c\}$, then $C' \setminus \{c\}$ is connected in T_{-c} because $P|_{C \setminus \{c\}}$ is single-peaked on T_{-c}. Therefore, to show that C' is connected in T, it suffices to show that c's neighbour b is also an element of C'. Since $b \in B(c) = \bigcap_{i \in N} B(i,c)$, we have that $b \in B(i,c)$. If $\mathrm{top}(i) = c$, then $B(i,c) = \{\mathrm{second}(i)\}$, so $b = \mathrm{second}(i)$. Because C' is a top-initial segment of i with $|C'| \geqslant 2$, we have $b \in C'$, as desired. Otherwise $\mathrm{top}(i) \neq c$, and so $B(i,c) = \{c' : c' \succ_i c\}$, hence $b \succ_i c$. Because C' is a top-initial segment of i including c, we must have $b \in C'$, as desired. □

With these results in place, we can now see how a recognition algorithm could work. Select an alternative c that is ranked bottom-most by some voter, select an arbitrary candidate $b \in B(c)$, add an edge $\{b, c\}$ to the tree under construction, remove c from the profile, and recurse on the remaining candidates. If at any point we find that $B(c) = \varnothing$, then we can conclude from Corollary 8.12 that the profile is not single-peaked on any tree. Algorithm 3 formalises this procedure. To avoid recursion, the algorithm uses the following notation: for every subset $C' \subset C$, for each $c \in C'$, and each $i \in N$, define

$$B(i,C',c) = \begin{cases} \{c' \in C' : c' \succ_i c\} & \text{if } \mathrm{top}(i,C') \neq c, \\ \{\mathrm{second}(i,C')\} & \text{if } \mathrm{top}(i,C') = c. \end{cases}$$

Theorem 8.14. *Algorithm 3 correctly decides whether a profile is single-peaked on a tree.*

Proof. First, note that if Algorithm 3 succeeds and returns a graph T, then T is a tree: first, it is easy to see that T has $|C| - 1$ edges. T is also connected, because every vertex is connected to a vertex in the set C_r at the end of the algorithm.

Algorithm 3 Trick's algorithm to decide whether a profile is single-peaked on a tree

$T \leftarrow (C, \varnothing)$, the empty graph on C
$C_1 \leftarrow C$, $r \leftarrow 1$
while $|C_r| \geqslant 3$ **do**
  $L_r \leftarrow \{\text{bottom}(i, C_r) : i \in N\}$
  for each candidate $c \in L_r$ **do**
    $B(c) \leftarrow \bigcap_{i \in N} B(i, C_r, c)$
    if $B(c) = \varnothing$ **then**
      return *fail : P is not single-peaked on any tree*
    else
      select $b \in B(c)$ arbitrarily
      add an edge between c and b in T
  $C_{i+1} \leftarrow C_r \setminus L_r$
  $r \leftarrow r + 1$
if $|C_r| = 2$ **then**
  add an edge between the two candidates in C' to T
return P is single-peaked on T

We show that the algorithm is correct by induction on $|C|$. If $|C| = 1$ or $|C| = 2$, every profile is single-peaked on the unique tree on C, and Algorithm 3 correctly determines this. If $|C| \geqslant 3$, then the while loop is executed at least once. If in the first iteration, the algorithm claims that the profile is not single-peaked on a tree because $B(c) = \varnothing$ for some $c \in L_1$, then this statement is correct by Corollary 8.12. Otherwise, the behaviour of the algorithm after the first iteration will be identical as if it was run on $P|_{C_2}$ (recall that $C_2 = C \setminus L_1$).

Now, if the algorithm fails in later iterations, by the inductive hypothesis, $P|_{C_2}$ is not single-peaked on a tree. But then P is not single-peaked on a tree either: suppose it was single-peaked on T. Then, by Proposition 8.10, all candidates in L_1 are leaves of T, and therefore $T|_{C_2}$ is still a tree, and so $P|_{C_2}$ is single-peaked on $T|_{C_2}$ (by Proposition 8.2), a contradiction. Thus, in this case, the algorithm run on P correctly determines that P is not single-peaked on a tree.

On the other other hand, if the algorithm run on P terminates and returns a tree T, then the algorithm run on $P|_{C_2}$ would have terminated and returned the tree $T|_{C_2}$. By the inductive hypothesis, $P|_{C_2}$ is single-peaked on $T|_{C_2}$. Hence, by Proposition 8.13, P is single-peaked on T, and so the algorithm is correct. □

Trick's algorithm makes some arbitrary choices when selecting alternatives $b \in B(c)$. Our aim is to understand the set $\mathcal{T}(P)$ of *all* trees that the input profile is single-peaked on, so a natural approach is to record all possible choices that Trick's algorithm could make at each step, and this will encode all possible outputs of the algorithm. We do this by running Algorithm 4, which has the same structure as Algorithm 3. Given a profile which is single-peaked on *some* tree, it constructs and returns a digraph D with vertex set C which contains all possible choices that Trick's algorithm can make. We call D the *attachment digraph* of the input profile.

Example 8.15. The attachment digraphs of the following three profiles are shown in Figure 8.3.

(a) Suppose $C = \{a, b, c, d, e\}$, and let P_1 be the profile with voters $N = \{1, 2\}$ such that $a \succ_1 b \succ_1 c \succ_1 d \succ_1 e$ and $e \succ_2 d \succ_2 c \succ_2 b \succ_2 a$, so that the two votes are the reverse of each other. Running Algorithm 4, we consider the sets $L_1 = \{a, e\}$ and $L_2 = \{b, d\}$.

(b) Suppose $C = \{a, b, c, d, e\}$, and let P_2 be the profile with voters $N = \{1, 2\}$ such that $a \succ_1 b \succ_1 c \succ_1 d \succ_1 e$ and $e \succ_2 b \succ_2 c \succ_2 d \succ_2 a$. Running Algorithm 4, we consider the sets $L_1 = \{a, e\}$ and $L_2 = \{d\}$.

Algorithm 4 Build attachment digraph $D = (C, A)$ of P

$D \leftarrow (C, A)$, $A \leftarrow \varnothing$, so D is the empty digraph on C
$C_1 \leftarrow C$, $r \leftarrow 1$
while $|C_r| \geqslant 3$ **do**
  $L_r \leftarrow \{\text{bottom}(i, C_r) : i \in N\}$
  for each candidate $c \in L_r$ **do**
    $B(c) \leftarrow \bigcap_{i \in N} B(v, C_r, c)$
    if $B(c) = \varnothing$ **then**
      return *fail : P is not single-peaked on any tree*
    else
      for each $b \in B(c)$, add an arc (c, b) to A
  $C_{i+1} \leftarrow C_r \setminus L_r$
  $r \leftarrow r + 1$
if $|C_r| = 2$ **then**
  add an arc between the two candidates in C' to A, arbitrarily directed
return D

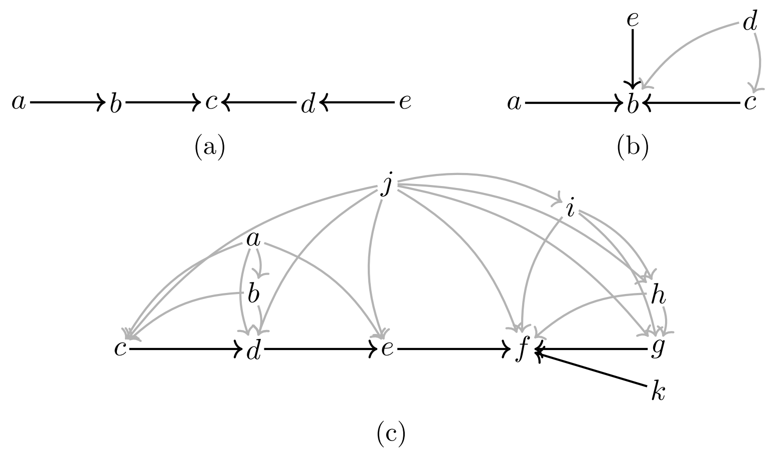


Figure 8.3.: The attachment digraphs of the profiles in Example 8.15. If a vertex has a unique outgoing arc, the arc is drawn in black. If the vertex has at least two outgoing arcs, the arcs are drawn in grey and curved.

(c) Suppose $C = \{a, b, c, d, e, f, g, h, i, j, k\}$, let P_3 be the profile with voters $N = \{1, 2, 3\}$ such that $k \succ_1 f \succ_1 e \succ_1 d \succ_1 g \succ_1 h \succ_1 c \succ_1 i \succ_1 j \succ_1 b \succ_1 a$, and $d \succ_2 c \succ_2 b \succ_2 e \succ_2 a \succ_2 f \succ_2 g \succ_2 h \succ_2 i \succ_2 j \succ_2 k$, and $g \succ_3 f \succ_3 h \succ_3 i \succ_3 e \succ_3 d \succ_3 c \succ_3 b \succ_3 a \succ_3 j \succ_3 k$. Running Algorithm 4, we consider the sets $L_1 = \{a, k\}$, $L_2 = \{b, j\}$, $L_3 = \{c, i\}$, $L_4 = \{d, h\}$, and $L_5 = \{e, g\}$.

Algorithm 4 runs in time $O(|N| \cdot |C|^2)$. In the rest of this section, we will analyse the structure of the attachment digraph, and its relation to the set $\mathcal{T}(P)$ of trees on which P is single-peaked. We start with a few simple properties.

Proposition 8.16. *Let $x \in C$ be a candidate with $x \in L_r$. Then $B(x) \cap L_r = \varnothing$. Hence, for every arc $(x, y) \in A$ with $x \in L_r$ and $y \in L_s$, we have that $s > r$.*

Proof. Assume for a contradiction that $y \in B(x)$ and $y \in L_r$. Since $y \in L_r$, there is some voter $i \in N$ such that $y = \text{bottom}(i, C_r)$. Since $y \in B(x)$, we have that $y \in B(i, C_r, x)$. Because $|C_r| \geqslant 3$ (by the condition of the while loop), by the definition of $B(i, C_r, x)$, this implies that $y \succ_i x$, which contradicts that $y = \text{bottom}(i, C_r)$. So $B(x) \cap L_r = \varnothing$.

For the last statement, note that if $(x, y) \in A$ is an arc, then $y \in B(x)$. Since $B(x) \subseteq C_r = C \setminus (L_1 \cup \cdots \cup L_{r-1})$, we must have $s \geqslant r$. By the previous paragraph, $s = r$ is impossible, and hence $s > r$. □

Proposition 8.17. *Every attachment digraph $D = (C, A)$ is acyclic and has exactly one sink.*

Proof. Suppose that the while loop of Algorithm 4 is executed $f - 1$ times, and consider the sets $L_1, \dots, L_{f-1}$. Set $L_f := C \setminus (L_1 \cup \cdots \cup L_{f-1})$. Then $L_1, \dots, L_f$ is a partition of C.

For acyclicity, note that for each $c \in L_r$, since $B(i, C', c) \subseteq C_r$, we have $B(c) \subseteq C_r$. From this and from Proposition 8.16, if $c \in L_r$ then all outgoing arcs of c point into $L_{i+1} \cup \cdots \cup L_f$. Hence, in the partition $L_1, \dots, L_f$, all arcs point to the right, and there cannot be any cycles in D.

For the number of sinks, note that there is at least one sink in D because D is acyclic. Since for every $c \in C \setminus L_f$ we have $B(c) \neq \varnothing$, at least one outgoing arc of c is added to D. Thus, c is not a sink. Since no $c \in C \setminus L_f$ can be a sink, all sinks are in L_f. The condition of the while loop implies that $|L_f| \leqslant 2$, and so D has at most two sinks. If $|L_f| = 1$, then there is exactly one sink and we are done. If $|L_f| = 2$, then the final if-clause of the algorithm adds an arc between those two vertices, which ensures that only one of them is a sink. □

If we wish to extract a tree $T \in \mathcal{T}(P)$ from the attachment digraph D, Trick's algorithm tells us that we must choose, for each non-sink vertex of D, exactly one outgoing arc, and add this arc as an edge. To formalise this process, we say that a function $f : C \setminus \{t\} \to C$ is an *attachment function* for D if $(c, f(c)) \in A$ is an arc of D for every $c \in C \setminus \{t\}$. Thus, f specifies one outgoing arc for each $c \in C \setminus \{c\}$. Given an attachment function f, we write $T(f)$ for the tree on C with edge set

$$\{\{c, f(c)\} : c \in C \setminus \{t\}\}$$

We now prove that all trees of $\mathcal{T}(P)$ can be obtained in this way.

Theorem 8.18. *Let P be a profile that is single-peaked on some tree, and let D be its attachment digraph. Then $T \in \mathcal{T}(P)$ if and only if $T = T(f)$ for some attachment function f. In other words, P is single-peaked on a tree T if and only if T's edge set consists of exactly one outgoing arc of each non-sink vertex of D.*

Proof. Suppose $T = T(f)$ for some attachment function f. Then T is a possible output of Algorithm 3, for a suitable way of making the selections from $B(c)$. Thus, by Theorem 8.14, the profile P is single-peaked on T.

We prove the converse by induction on $|C|$. If $|C| \leqslant 2$, then P is single-peaked on the unique tree on C, which can be obtained as a $T(f)$. So suppose that $|C| \geqslant 3$, and that $T = (C, E)$ is a tree such that P is single-peaked on T. During the first iteration of Algorithm 4, the algorithm determines the set L_1 of candidates occurring in bottom position, and sets $C_2 = C \setminus L_1$. From Proposition 8.10, the vertices in L_1 are leaves of T. Hence, the induced subgraph $T|_{C_2}$ is also a tree, and thus $P|_{C_2}$ is single-peaked on $T|_{C_2}$. Also, by inspection of Algorithm 4, the attachment digraph of $P|_{C_2}$ is $D|_{C_2}$. By the inductive hypothesis, $T|_{C_2} = T(f')$ for some attachment function f' defined for $D|_{C_2}$. Thus, we can define an attachment function f so that for each $c \in C_2 \setminus \{t\}$ we set $f(c) = f'(c)$, and for each $c \in L_1$ we set $f(c)$ to be the unique neighbour of c in T. By Corollary 8.12, T is obtained from $T|_{C_2}$ by attaching each $c \in L_1$ to an element of $B(c)$, which implies that f is a legal attachment function. Thus, $T = T(f)$, which proves the claim. □

Using this characterisation of the set $\mathcal{T}(P)$, we can conclude the following fact about the cardinality $|\mathcal{T}(P)|$, which must be equal to the number of different attachment functions, noting that $T(f_1) \neq T(f_2)$ whenever $f_1 \neq f_2$.

Corollary 8.19. *The number of trees in $\mathcal{T}(P)$ is equal to the product of the out-degrees of the non-sink vertices of D. Hence we can compute $|\mathcal{T}(P)|$ in polynomial time.*

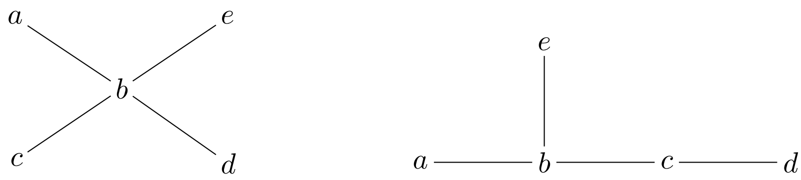


Figure 8.4.: The set $\mathcal{T}(P_2)$ of trees on which P_2 from Example 8.15 are single-peaked consists of these two trees. For the left-hand tree, the attachment function has $f(d) = b$, while for the right-hand tree, it has $f(d) = c$.

For the profiles in Example 8.15, we see that P_1 is single-peaked on a unique tree (a path), that P_2 is single-peaked on exactly 2 trees (shown in Figure 8.4), and that P_3 is single-peaked on exactly 336 different trees.

It turns out that attachment digraphs have a lot of structure beyond the results of Proposition 8.17. A key property, which will allow us to use essentially greedy algorithms, is what we call circumtransitivity.

Definition 8.20. A directed acyclic graph $D = (C, A)$ is *circumtransitive* if its vertices can be partitioned into a set $C^{\to}$ of *forced* vertices and a set $C^{\rightrightarrows} = C \setminus C^{\to}$ of *free* vertices such that

1. every forced vertex $c \in C^{\to}$ has out-degree at most 1, and if $(c, c') \in A$ then also $c' \in C^{\to}$, and
2. every free vertex $c \in C^{\rightrightarrows}$ has out-degree at least 2, and whenever $x, y \in C^{\rightrightarrows}$ and $z \in C$ are such that $(x, y), (y, z) \in A$, then $(x, z) \in A$.

Recall that every directed acylic graph D has at least one sink. If it is also circumtransitive, then the sinks of D must belong to the forced vertices. A circumtransitive digraph consists of an inner part (the *forced part*), and an outer part that is transitively attached to the inner part.

Theorem 8.21. *Every attachment digraph (C, A) is circumtransitive.*

Proof. We will argue that Definition 8.20 is satisfied by taking the partition

$$C^{\to} = \{c : d^+(c) \leqslant 1\}, \qquad C^{\rightrightarrows} = \{c : d^+(c) \geqslant 2\}.$$

Forced: Let $x \in C^{\to}$. If $d^+(x) = 0$, there is nothing to prove, so assume that $d^+(x) = 1$, i.e., $(x, y) \in A$ for some $y \in C$. We will show that $d^+(y) \in \{0, 1\}$ and hence that $y \in C^{\to}$.

If y is a sink, we are done. Otherwise, there exists an arc $(y, z) \in A$ for some $z \in C$. Suppose that $x \in L_r$ and $y \in L_s$ for some $1 \leqslant r < s \leqslant f - 1$. (Such r, s exist because neither x nor y are sinks. We have $r < s$ because $(x, y) \in A$, see Proposition 8.16.) Because y is the only out-neighbour of x, we have that $(x, z) \notin A$ and so $z \notin B(x) = \bigcap_{i \in N} B(i, C_r, x)$. Note that $y \in C_r$ and hence $z \in C_r$ as well. Since $z \notin B(x)$, we have $z \notin B(i, C_r, x)$ for some $i \in N$. On the other hand, since $(x, y) \in A$, we have $y \in B(x)$ and thus $y \in B(i, C_r, x)$ We consider two cases:

(i) $x \neq \mathrm{top}(i, C_r)$. Then $B(i, C_r, x) = \{c \in C_r : c \succ_i x\}$, and thus $y \succ_i x \succ_i z$. Now consider iteration s. If $y = \mathrm{top}(i, C_s)$, then $|B(i, C_s, y)| = 1$ so $|B(y)| = 1$. Hence, y has exactly one out-neighbour, as desired. Otherwise, since $(y, z) \in A$, we have $z \in B(i, C_s, y)$ and so $z \succ_i y$. But then by transitivity $z \succ_i x$, contradicting that $z \notin B(i, C_r, x)$.

(ii) $x = \mathrm{top}(i, C_r)$. Then $B(i, C_r, x) = \{\mathrm{second}(i, C_r)\}$, and thus since $y \in B(i, C_r, x)$ we see that $y = \mathrm{second}(i, C_r)$. Note that $x \notin C_s$ but $y \in C_s$. Therefore, $\mathrm{top}(i, C_s) = y$, and thus $|B(i, C_j, y)| = 1$ so $|B(y)| = 1$. Hence y has exactly one out-neighbour, as desired.

Free: Consider vertices $x, y, z \in C$ with $x, y \in C^{\rightrightarrows}$ and $(x, y), (y, z) \in A$. Since $x, y \in C^{\rightrightarrows}$, they are not sinks, so $x, y \notin L_f$. Take $r < s$ such that $x \in L_r$ and $y \in L_s$. Note that if there was a voter $i \in N$ with $\text{top}(i, C_r) = x$, then $|B(x)| = 1$, contradicting that $d^+(x) > 1$. Thus $\text{top}(i, C_r) \neq x$ for all $i \in N$. Because $(x, y) \in A$, for all $i \in N$ we have $y \in B(i, C_r, x)$ and hence $y \succ_i x$. Similarly, since $(y, z) \in A$ and $d^+(y) > 1$ we have $z \succ_i y$ for all $i \in N$. Hence, by transitivity, $z \succ_i x$ for all $i \in N$. Therefore $z \in \bigcap_{i \in N} B(i, C_r, x) = B(x)$ and so $(x, z) \in A$, as desired. □

Suppose that f is an attachment function for D. Then for each forced vertex $c \in C^{\rightarrow} \setminus \{t\}$, the value of $f(c)$ is uniquely determined, since c has exactly one out-neighbour. Also, it is easy to see that $\mathcal{G}(D|_{C^{\rightarrow}})$ is a tree (it is connected because we can reach the sink t from every forced vertex). It follows that for every $T \in \mathcal{T}(P)$, the tree $\mathcal{G}(D|_{C^{\rightarrow}})$ is a subtree of T.

Finally, we study the free vertices $C^{\rightrightarrows}$ more closely. Part (d) of the following proposition will be particularly useful; it states that every free vertex can be attached to either of two forced vertices which are adjacent in D.

Proposition 8.22. *Suppose $|C| \geqslant 3$. For every free vertex $x \in C^{\rightrightarrows}$ of the attachment digraph $D = (C, A)$,*

(a) there is a forced vertex $y \in C^{\rightarrow}$ with $(x, y) \in A$;

(b) there are at least two forced vertices $y, z \in C^{\rightarrow}$ with $(x, y), (x, z) \in A$;

(c) the set $\{y \in C^{\rightarrow} : (x, y) \in A\}$ induces a subtree in $\mathcal{G}(D)$;

(d) there are two forced vertices $y, z \in C^{\rightarrow}$ with $(x, y), (x, z), (y, z) \in A$.

Proof. (a) The directed acyclic graph D has a unique sink t, and there must be a directed path $x = c_1 \rightarrow c_2 \rightarrow \cdots \rightarrow c_p = t$ from x to t; take such a path of minimum length. If $p = 2$, then $(x, t) \in A$, and so we are done since $t \in C^{\rightarrow}$. So suppose that $p \geqslant 3$. Assume first that c_2 is a free vertex. Then $c_1, c_2 \in C^{\rightrightarrows}$, and $(c_1, c_2), (c_2, c_3) \in A$, and since D is circumtransitive, we have $(c_1, c_3) \in A$. Thus, $c_1 \rightarrow c_3 \rightarrow \cdots \rightarrow c_p$ is a shorter path, contradiction. Hence $c_2 \in C^{\rightarrow}$, and we have proved that x is adjacent to at least one forced vertex.

(b) Suppose the statement is false. Choose r maximum such that there is a free vertex $x \in C^{\rightrightarrows}$ with $x \in L_r$ such that there is a unique forced vertex $y \in C^{\rightarrow}$ with $(x, y) \in A$. Because $x \in C^{\rightrightarrows}$, we have $d^+(x) \geqslant 2$, and so there must be $w \in C^{\rightrightarrows}$ with $(x, w) \in A$. Because $(x, w) \in A$, we have $w \in L_s$ for some $s > r$. Since w is a free vertex, by (a), there must be a forced vertex $y' \in C^{\rightarrow}$ with $(w, y') \in A$. By circumtransitivity, since $(x, w) \in A$ and $(w, y') \in A$, we have that $(x, y') \in A$, and thus $y = y'$. Thus we have $(w, y) \in A$. Hence, w has a unique forced out-neighbour, which contradicts maximality of r.

(c) Suppose $x \in L_r$, and suppose that A contains arcs (x, y) and (x, z) where $y, z \in C^{\rightarrow}$. Since $\mathcal{G}(D|_{C^{\rightarrow}})$ is a tree, there is a unique path Q from y to z in $\mathcal{G}(D|_{C^{\rightarrow}})$; let $C_Q \subseteq C^{\rightarrow}$ be the vertex set of the path Q. Fix any tree $T \in \mathcal{T}(P)$. Then $\mathcal{G}(D|_{C^{\rightarrow}})$ is a subgraph of T, and so Q is a path in T. Pick a vote $i \in N$. Since $y, z \in B(x)$, we have $|B(x)| > 1$ and so $|B(i, C_r, x)| > 1$, and thus we have $y \succ_i x, z \succ_i x$. Consider the top-initial segment $C' = \{c \in C : c \succ_i x\}$. By Proposition 8.2, since P is single-peaked on T, the set C' is connected in T. Since $y, z \in C'$, the path Q must be contained in $T|_{C'}$, and hence $C_Q \subseteq C'$. Thus, $w \succ_i x$ for each $w \in C_Q$, and so $C_Q \subseteq B(i, C_r, x)$. As this holds for every $i \in N$, $C_Q \subseteq B(x)$, and so $C_Q \subseteq \{y \in C^{\rightarrow} : (x, y) \in A\}$. Hence, $\{y \in C^{\rightarrow} : (x, y) \in A\}$ is connected in $\mathcal{G}(D|_{C^{\rightarrow}})$.

(d) The set $\{y \in C^{\rightarrow} : (x, y) \in A\}$ is connected (by (c)) and contains at least two members (by (b)). Hence, by definition of $\mathcal{G}$, it contains some vertices y and z with $(y, z) \in A$. □

This concludes our study of the properties of attachment digraphs.

8.8. Recognition Algorithms: Finding Nice Trees

Suppose we are given a profile P with $\mathcal{T}(P) \neq \varnothing$ and wish to find trees in $\mathcal{T}(P)$ that satisfy additional desiderata. We will now show how the attachment digraph can be used to achieve this. We assume throughout this section that $|C| \geqslant 3$, since otherwise there is a unique tree T on C, and there is no problem of selecting the best tree.

8.8.1. Minimum Number of Internal Vertices

In Section 8.6, we saw an algorithm that could solve Utilitarian CC efficiently for profiles single-peaked on a tree T with few internal vertices, where T was taken as input to the algorithm. We now show how we can find, given a profile, the tree $T \in \mathcal{T}(P)$ that has the fewest internal vertices. Algorithm 5 constructs an attachment function, and tries to make every vertex a leaf, if possible. In particular, every free vertex in the attachment digraph will become a leaf. We begin by showing that the description of Algorithm 5 is well-defined, in the sense that existence statement in the algorithm are correct.

Algorithm 5 Find $T \in \mathcal{T}(P)$ with fewest internal vertices

Let $D = (C, A)$ be the attachment digraph of P
Let $C^{\rightarrow}, C^{\rightrightarrows}$ be the collection of forced and free vertices in D
Let t be the sink vertex of D
$f \leftarrow \varnothing$, an attachment function under construction
for each $c \in C^{\rightarrow} \setminus \{t\}$ **do**
    $f(c) \leftarrow c'$ where c' is the unique $c' \in C$ with $(c, c') \in A$
if $|C^{\rightarrow}| = 2$ **then**
    pick some $y \in C^{\rightarrow}$
    for each $c \in C^{\rightrightarrows}$ **do**
        $f(c) \leftarrow y$
else if $|C^{\rightarrow}| > 2$ **then**
    for each $c \in C^{\rightrightarrows}$ **do**
        find $y \in C^{\rightarrow}$ such that $(c, y) \in A$ and y is internal in $\mathcal{G}(D|_{C^{\rightarrow}})$
        $f(c) \leftarrow y$
return $T^* = T(f)$

Proposition 8.23. *Algorithm 5 returns a tree $T^* \in \mathcal{T}(P)$.*

Proof. This follows from Theorem 8.18 once we can show that the choices of the algorithm are possible. With our running assumption that $|C| \geqslant 3$, it follows that $|C^{\rightarrow}| \geqslant 2$ from Proposition 8.22.

Suppose that $|C^{\rightarrow}| > 2$. By Proposition 8.22, each free vertex $c \in C^{\rightrightarrows}$ has outgoing arcs to two forced vertices which are adjacent in $\mathcal{G}(D|_{C^{\rightarrow}})$. Not both of them can be leaves in the tree $\mathcal{G}(D|_{C^{\rightarrow}})$ since $|C^{\rightarrow}| > 2$, so there is $y \in C^{\rightarrow}$ with $(c, y) \in A$ such that y is internal in $\mathcal{G}(D|_{C^{\rightarrow}})$. Thus, the algorithm is well-defined in this case.

Suppose that $|C^{\rightarrow}| = 2$. By Proposition 8.22, each $c \in C^{\rightrightarrows}$ is adjacent to both vertices in $C^{\rightarrow}$, and thus $(c, y) \in A$ for the choice of $y \in C^{\rightarrow}$ made by the algorithm; thus $T^* \in \mathcal{T}(P)$. □

Next, we show that Algorithm 5 returns an optimal tree.

Proposition 8.24. *Algorithm 5 returns a tree $T^* \in \mathcal{T}(P)$ with the minimum number of internal vertices among trees in $\mathcal{T}(P)$ in polynomial time.*

Proof. For every tree $T \in \mathcal{T}(P)$, we must have that $\mathcal{G}(D|_{C^{\rightarrow}}) \subseteq T$, by Theorem 8.18 and the definition of $C^{\rightarrow}$. Thus, if $c \in C^{\rightarrow}$ is not a leaf in the tree $\mathcal{G}(D|_{C^{\rightarrow}})$, then c cannot be a leaf in T.

Suppose that $|C^{\rightarrow}| > 2$. Note that every free vertex $c \in C^{\rightrightarrows}$ is a leaf in T^* because $f(c) \in C^{\rightarrow}$ for all $c \in C \setminus \{t\}$. Further, every leaf of $\mathcal{G}(D|_{C^{\rightarrow}})$ is also a leaf in T^*. By our initial observation, none of the remaining vertices can be a leaf in any $T \in \mathcal{T}(P)$, so T^* has the maximum number of leaves, and hence a minimum number of internal vertices.

Suppose that $|C^{\rightarrow}| = 2$. Since $|C| \geqslant 3$, we have $C^{\rightrightarrows} \neq \varnothing$. Since the two members of $C^{\rightarrow}$ are adjacent in any $T \in \mathcal{T}(P)$, it can't be that both of them are leaves in T. Hence the number of leaves in $T \in \mathcal{T}(P)$ is at most $|C^{\rightrightarrows}| + 1$. The tree T^* has exactly $|C^{\rightrightarrows}| + 1$ leaves, and hence is optimal. □

8.8.2. Minimum Diameter

It turns out that the tree found by Algorithm 5 is also optimal on another metric: it minimises the diameter.

Proposition 8.25. *Algorithm 5 returns a tree $T^* \in \mathcal{T}(P)$ with minimum diameter among trees in $\mathcal{T}(P)$ in polynomial time.*

Proof. Suppose that $|C^{\rightarrow}| = 2$. Then T^* is a star with center y; no tree on at least three vertices has smaller diameter than a star.

Suppose that $|C^{\rightarrow}| > 2$. In this case the diameter of T^* is equal to the diameter of $\mathcal{G}(D|_{C^{\rightarrow}})$. To see this, consider a longest path $(c_1, \dots, c_k)$ in T^*. If $k = 2$, then T^* is a star, which is a minimum-degree tree when there are $|C| \geqslant 3$ vertices. So suppose that $k \geqslant 3$. On the longest path, only c_1 and c_k can be free vertices, since all free vertices are leaves in T^*. Suppose $c_1 \in C^{\rightrightarrows}$. Then by construction of T^*, $c_2 \in C^{\rightarrow}$, and c_2 is an internal vertex of $\mathcal{G}(D|_{C^{\rightarrow}})$. Hence, c_2 has at least two neighbours in $C^{\rightarrow}$. Thus, we can find a neighbour c_1' of c_2 such that $c_1' \neq c_3$. Then we can replace c_1 by c_1' in the longest path (noting that c_1' cannot appear elsewhere on the path because $\mathcal{G}(D|_{C^{\rightarrow}})$ is a tree). Similarly we can replace c_k by a forced neighbour of c_{k-1} if $c_k \in C^{\rightrightarrows}$. Having replaced all free vertices on the path by forced vertices, we have obtained a longest path in T^* which is completely contained in $\mathcal{G}(D|_{C^{\rightarrow}})$. Hence, the diameter of T^* is equal to the diameter of $\mathcal{G}(D|_{C^{\rightarrow}})$.

Because $\mathcal{G}(D|_{C^{\rightarrow}}) \subseteq T$ for every $T \in \mathcal{T}(P)$, the diameter of any $T \in \mathcal{T}(P)$ must be at least the diameter of $\mathcal{G}(D|_{C^{\rightarrow}})$. Hence T^* has minimum diameter. □

8.8.3. Minimum Number of Leaves

In Section 8.5, we saw an algorithm for Utilitarian CC which is efficient when the input profile is single-peaked on a tree with few leaves. The algorithm assumed that the tree T is given in its input. Here, given a profile P, we show how to find the tree $T^* \in \mathcal{T}(P)$ with the fewest leaves.

Minimising the number of leaves of a tree is equivalent to maximising its number of internal vertices. For this, we first characterise the set of internal vertices of a tree $T(f)$.

Proposition 8.26. *Let f be an attachment function for the attachment digraph D. Then $c \in C \setminus \{t\}$ is an internal vertex of the tree $T(f)$ if and only if $|f^{-1}(c)| \geqslant 1$, i.e., c is in the range of f. The sink vertex t is an internal vertex of $T(f)$ if and only if $|f^{-1}(t)| \geqslant 2$, i.e., there are two distinct $d_1, d_2 \in C$ with $f(d_1) = f(d_2) = c$.*

Proof. A vertex is internal in a tree if and only if it has degree at least two. From the definition of $T(f)$, for $c \in C \setminus \{t\}$, the degree of c is $1 + |f^{-1}(c)|$, and the degree of t is $|f^{-1}(t)|$. The claim follows immediately. □

Algorithm 6 Find $T \in \mathcal{T}(P)$ with fewest leaves

Let $D = (C, A)$ be the attachment digraph of P
$f \leftarrow \varnothing$, an attachment function under construction
Let t be the sink vertex of D, and let $s \in C^{\nrightarrow}$ be a forced vertex with unique outgoing arc $(s, t) \in A$ (this exists by Proposition 8.22)
$f(s) \leftarrow t$
Construct a bipartite graph H with vertex set $L \cup R$ where $A = \{\ell_c : c \in C \setminus \{s, t\}\}$ and $B = \{r_c : c \in C\}$ and edge set $E_H = \{\{\ell_c, r_d\} : (c, d) \in A\}$.
Find a maximum matching $M \subseteq E_H$ in H
for each $c \in C \setminus \{t\}$ **do**
  if ℓ_c is matched in M, i.e. $\{\ell_c, r_d\} \in M$ for some $d \in C$ **then**
    $f(c) \leftarrow d$
  else
    take any $d \in C$ with $(c, d) \in A$
    $f(c) \leftarrow d$
return $T^* = T(f)$

Using this observation, we can prove that Algorithm 6 returns an optimal tree. The algorithm is based on constructing a maximum matching.

Proposition 8.27. *Algorithm 6 returns a tree $T^* \in \mathcal{T}(P)$ with the minimum number of leaves among trees in $\mathcal{T}(P)$ in polynomial time.*

Proof. That the output T^* of the algorithm is a member of $\mathcal{T}(P)$ is clear from Theorem 8.18, since the algorithm constructs an attachment function.

To see optimality, note first that minimising the number of leaves is equivalent to maximising the number of internal vertices. Thus, by Proposition 8.26, we need to find an attachment function f maximising the number of vertices c with $|f^{-1}(c)| \geqslant 1$ if $c \neq t$ or $|f^{-1}(c)| \geqslant 2$ if $c = t$.

We claim that under the attachment function f constructed by Algorithm 6, a vertex $d \in C$ is an internal vertex of $T(f)$ if and only if r_d is matched in the maximum matching M. We start with the if direction.

- Suppose $d = t$. If r_t is matched in M to ℓ_c, then both $c \in f^{-1}(t)$ and $s \in f^{-1}(t)$, where s is the vertex chosen at the very start of the algorithm. By definition of the bipartite graph H, $c \neq s$, and so $|f^{-1}(t)| \geqslant 2$, and hence t is an internal vertex by Proposition 8.26.

- Suppose $d \neq t$. If r_d is matched in M to ℓ_c, then $c \in f^{-1}(d)$, and so d is internal by Proposition 8.26.

For the only if direction, suppose that $d \in C$ is not matched in M. Then in the **for** loop of Algorithm 6, we never set $f(c) \leftarrow d$ for any $c \in C \setminus \{c\}$, because otherwise we could add the edge $\{\ell_c, r_d\}$ to the matching M, contradicting its maximality. Hence, if $d = t$ is not matched, then $f^{-1}(t) = \{s\}$, and so t is not internal. If $d \neq t$ is not matched, then $f^{-1}(d) = \varnothing$, so d is not internal. It follows that the number of internal vertices of $T(f)$ is $|M|$, and our claim is proved.

Now suppose that $T(f)$ is not optimal, and that $T' \in \mathcal{T}(P)$ is a tree with $q > |M|$ internal vertices. By Theorem 8.18, since $T' \in \mathcal{T}(P)$, we have $T' = T(g)$ for some attachment function g. But then we can construct a matching M' in H of size $|M'| = q$, as follows:

- If t is an internal vertex in T', then by Proposition 8.26, we have $|g^{-1}(t)| \geqslant 2$. Select some $c \in g^{-1}(t)$ with $c \neq s$, and add $\{\ell_c, r_d\}$ to M'. If t is not internal in T', do nothing.

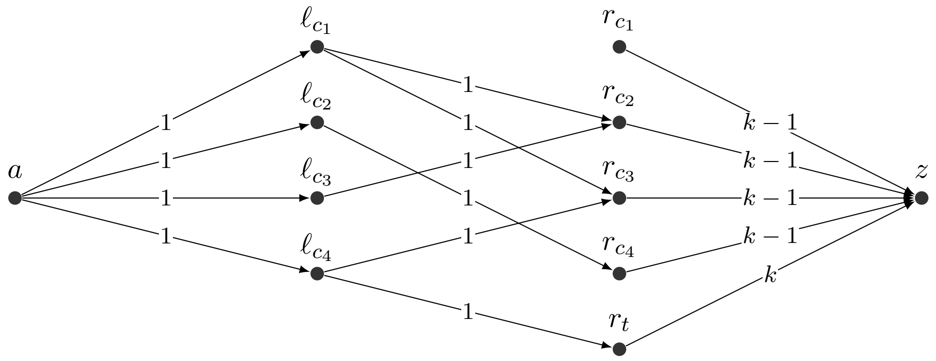


Figure 8.5.: Flow network H constructed by Algorithm 7.

- For each $d \in C \setminus \{c\}$ which is an internal vertex of T', select some $c \in g^{-1}(d)$ (which exists by Proposition 8.26), and add $\{\ell_c, r_d\}$ to M'.

Clearly, we have added q edges to M'. Because g is a function, M' is a matching. Since $|M'| > |M|$, we have a contradiction to maximality of M. □

8.8.4. Minimum Max-Degree

In some situations, it may be desirable to find a tree in which each vertex is connected to only a few other vertices. The following algorithm can be used to do so; it is based on calculating a maximum flow network, an example of which is shown in Figure 8.5.

Algorithm 7 Decide whether there is $T \in \mathcal{T}(P)$ with maximum degree at most k

Let $D = (C, A)$ be the attachment digraph of P
Let t be the sink vertex of D
Let $L = \{\ell_c : c \in C \setminus \{t\}\}$ and $B = \{r_c : c \in C\}$ and construct a flow network H on vertex set $\{a, z\} \cup L \cup R$ with arc set

$$E_H = \{(a, \ell_c) : c \in C \setminus \{t\}\} \cup \{(\ell_c, r_d) : c, d \in C, (c, d) \in A\} \cup \{(r_c, t) : c \in C\},$$

and capacities $\text{cap}(a, \ell_c) = 1$ for all $c \in C \setminus \{t\}$, $\text{cap}(\ell_c, r_d) = 1$ for all $(c, d) \in A$, $\text{cap}(r_c, t) = k-1$ for all $c \in C \setminus \{t\}$, and $\text{cap}(r_t, z) = k$.
Find a maximum flow in H
$f \leftarrow \varnothing$, an attachment function under construction
if the flow transports $|C \setminus \{t\}|$ units of flow **then**
  For each $(c, d) \in A$ such that a unit of flow flows across (ℓ_c, r_d), set $f(c) \leftarrow d$
  return $T^* = T(f)$
else
  return there is no $T^* \in \mathcal{T}(P)$ with maximum degree at most k

Proposition 8.28. *Algorithm 7 returns a tree $T^* \in \mathcal{T}(P)$ with maximum degree at most k if one exists, in polynomial time.*

Proof. Let f be some attachment function. By definition of $T(f)$, for each $c \in C \setminus \{t\}$, the degree of c in $T(f)$ is $1 + |f^{-1}(c)|$, because there is 1 edge in $T(f)$ corresponding to an outgoing arc of c in D, and $|f^{-1}(c)|$ edges in $T(f)$ corresponding to incoming arcs to c in D. Also, the degree of the sink vertex t in $T(f)$ is $|f^{-1}(t)|$. Thus, our task reduces to deciding whether there exists an attachment function f with

$$1 + |f^{-1}(c)| \leqslant k \text{ (i.e., } |f^{-1}(c)| \leqslant k) \text{ for each } c \in C \setminus \{t\} \text{ and } |f^{-1}(t)| \leqslant k. \tag{8.1}$$

Such attachment functions are in one-to-one correspondence with (integral) flows of size $|C \setminus \{t\}|$ in the flow network constructed by Algorithm 7: Suppose f is an attachment function satisfying (8.1); then send one unit of flow from the super-source a along each of its $|C \setminus \{t\}|$ outgoing links. For each $c \in C \setminus \{t\}$, send the incoming flow into ℓ_c towards $r_{f(c)}$. Finally, for each $c \in C$, send the incoming flow into r_c towards the super-sink z; this satisfies the capacity constraints because f satisfies (8.1). Conversely, for any flow of size $|C \setminus \{t\}|$, we can define f by setting $f(c)$ to correspond to the destination of the out-flow from ℓ_c. The resulting f satisfies (8.1) due to the capacity constraints of the links between the r_c and the super-sink z. □

8.8.5. Minimum Pathwidth

Here, we show how to find a tree $T \in \mathcal{T}(P)$ of minimum pathwidth. Our algorithm is based on an algorithm by Scheffler [1990], which computes a minimum-width path decomposition of a given tree in linear time.

We need a preliminary result showing that a tree always admits a minimum-width path decomposition with a certain property: most vertices appear in a bag of the path decomposition which has some 'slack', in the sense that the bag does not have maximum cardinality.

Lemma 8.29. *For every tree $T = (C, E)$, there exists a path decomposition $S_1, \dots, S_r$ of T of minimum width w such that, for every edge $e \in E$, there is $c \in e$ for which there exists a bag S_i with $c \in S_i$ such that $|S_i| \leqslant w$ (note that $\max_i |S_i| = w + 1$).*

Proof. We show how to transform any path decomposition of T into a path decomposition of the same width having the desired property.

Suppose $S_1, \dots, S_r$ is a path decomposition of T with width w. For each edge $\{c, d\} \in E$, we do the following: Because $\{c, d\}$ is an edge, there exists a bag containing both c and d Let $i \in \{1, \dots, r\}$ be minimum such that $c, d \in S_i$.

If $i = 1$, then we can create a new bag $S_0 = S_1 \setminus \{d\}$ and append it to the left of the sequence, and the result is still a path decomposition. In this path decomposition, c appears in bag $|S_0|$, where $|S_0| \leqslant w$ since $|S_0| < |S_1| \leqslant w + 1$.

If $i > 1$, then one of c or d does not appear in S_{i-1}, say $d \notin S_{i-1}$. Then we can create a new bag $S_{i-\frac{1}{2}} = S_i \setminus \{d\}$, and place it in between S_{i-1} and S_i. The resulting sequence is still a path decomposition, in which c appears in $S_{i-\frac{1}{2}}$, with $|S_{i-\frac{1}{2}}| \leqslant w$ since $|S_{i-\frac{1}{2}}| < |S_i| \leqslant w + 1$. □

Clearly, the transformation described in the proof of Lemma 8.29 can be performed in polynomial time. Since one can find some path decomposition in polynomial time, one can find a path decomposition with the property of Lemma 8.29 in polynomial time.

Proposition 8.30. *Algorithm 8 returns a tree $T^* \in \mathcal{T}(P)$ with minimum pathwidth among trees in $\mathcal{T}(P)$ in polynomial time.*

Proof. First, note that the path decomposition constructed by Algorithm 8 is in fact a path decomposition of the output tree $T(f)$: Each free vertex $c \in C^{\rightrightarrows}$ becomes a leaf in $T(f)$, and occurs in only a single bag S_i in the constructed path decomposition. Since c is a leaf, it is only a part of a single edge $\{c, f(c)\}$, and we have $c, f(c) \in S_i$. Also, since c occurs in only a single bag, the set of bags containing c is trivially an interval of the path decomposition sequence.

Next, observe that the path decomposition of $T(f)$ has the same width w as the pathwidth of the forced part $\mathcal{G}(D|_{C^{\nrightarrow}})$, because all new bags have cardinality at most $w + 1$. Now, because $\mathcal{G}(D|_{C^{\nrightarrow}})$ is a subgraph of every $T \in \mathcal{T}(P)$, no $T \in \mathcal{T}(P)$ can have a smaller pathwidth than $\mathcal{G}(D|_{C^{\nrightarrow}})$. Since Algorithm 8 identifies a tree $T \in \mathcal{T}(P)$ with the same pathwidth as $\mathcal{G}(D|_{C^{\nrightarrow}})$, this must be optimal. □

Algorithm 8 Find a tree $T \in \mathcal{T}(P)$ of minimum pathwidth

Let $D = (C, A)$ be the attachment digraph of P
Let $S_1, \dots, S_r$ be a path decomposition of $\mathcal{G}(D|_{C^{\rightarrow}})$ of minimum width w, satisfying the condition of Lemma 8.29
$f \leftarrow \varnothing$, an attachment function under construction
for each $c \in C \setminus \{t\}$ **do**
    if $c \in C^{\rightarrow}$ **then**
        $f(c) \leftarrow d$, for the unique $d \in C$ with $(c, d) \in A$
    else if $c \in C^{\rightrightarrows}$ **then**
        Let $d_1, d_2 \in C^{\rightarrow}$ be two forced vertices such that $(d_1, d_2), (c, d_1), (c, d_2) \in A$
            (these exist by Proposition 8.22)
        Since $\{d_1, d_2\}$ is an edge of $\mathcal{G}(D|_{C^{\rightarrow}})$, by the condition of Lemma 8.29,
            there is a bag S_i with $d_i \in S_i$ and $|S_i| \leqslant w$, for some $i \in \{1, 2\}$
        $f(c) \leftarrow d_i$
        Make a new bag $S_{i+\frac{1}{2}} = S_i \cup \{d_i\}$ and place it to the right of S_i
            in the sequence of the path decomposition
return $T^* = T(f)$

8.8.6. Other Graph Types

Finally, we briefly collect some observations about recognising whether $\mathcal{T}(P)$ contains trees of certain types.

Paths If a profile is single-peaked on a path, then it is simply single-peaked. The literature contains several algorithms for recognising profiles that are single-peaked on a path. The algorithms by Doignon and Falmagne [1994] and Escoffier et al. [2008] can be implemented to run in time $O(mn)$. One could also use some of the algorithms presented above. Algorithm 6 finds a tree $T \in \mathcal{T}(P)$ with a minimum number of leaves; clearly, if $\mathcal{T}(P)$ contains a path, then this will be identified by the algorithm. Alternatively, Algorithm 7 can look for a tree $T \in \mathcal{T}(P)$ with maximum degree $k = 2$; this will succeed if and only if the profile is single-peaked on a path. Both Algorithms 6 and 7 depend on pre-computing the attachment digraph, which takes time $O(m^2 n)$. Thus, this approach is slower than using the linear time algorithms referred to above.

Stars In Proposition 8.8, we observed that a profile is single-peaked on a star graph if and only if there is a candidate $c \in C$ such that every voter ranks c in either first or second condition. This condition can easily be verified in linear time, without needing to compute the attachment digraph. Note that Algorithm 5 (minimising the number of internal vertices) will output a star whenever $\mathcal{T}(P)$ contains a star graph.

Caterpillars Caterpillar graphs are exactly the trees of pathwidth 1 [Proskurowski and Telle, 1999], and so Algorithm 8 can check whether a profile is single-peaked on a caterpillar. One can also search for a caterpillar directly: first compute $\mathcal{G}(D|_{C^{\rightarrow}})$ and check that it is a caterpillar (if not, then no tree in $\mathcal{T}(P)$ can be a caterpillar). If yes, then we can attach every free vertex as a leaf to $\mathcal{G}(D|_{C^{\rightarrow}})$, generating a caterpillar.

Subdivision of a Star A tree is a subdivision of a star if all but one vertex has degree at most 2. We can find a subdivision of a star in $\mathcal{T}(P)$, should one exist, by adapting Algorithm 7: we guess the center of the subdivision of the star, and then appropriately assign upper bounds on the vertex degrees by setting the capacity constrains in the flow network.

8.9. Hardness of Recognising Single-Peakedness on a Specific Tree

The algorithms presented in Section 8.8 enable us to answer a wide range of questions about the set $\mathcal{T}(P)$. The NP-hardness results in this section, however, show that not every such question can be answered efficiently unless P = NP.

Two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are *isomorphic* if there is a bijection $\phi : V_1 \to V_2$ such that for all $u, v \in V_1$, we have that $\{u, v\} \in E_1$ if and only if $\{\phi(u), \phi(v)\} \in E_2$. We consider the following computational problem.

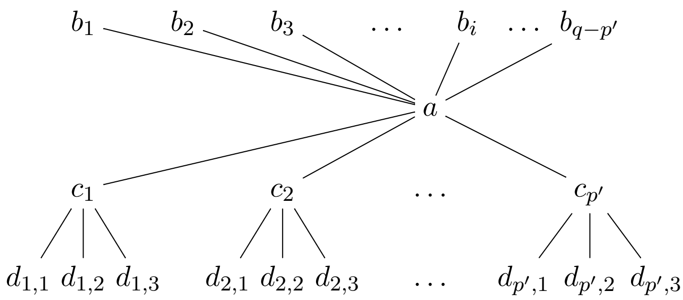
Single-Peaked Tree Labelling	
Instance:	Profile P over C, a tree T_0 on $\lvert C\rvert$ vertices
Question:	Is there a tree $T = (C, E)$ isomorphic to T_0 such that P is single-peaked on T?

In this problem, we are given a "template" unlabelled tree T_0, and need to decide whether we can label the vertices in this template by candidates so as to make the input profile single-peaked on the resulting labelled tree. For example, if T_0 is a path, then the problem is to decide whether the profile P is single-peaked on a path, and in this case the problem is easy to solve. However, the template T_0 occurs in the *input* to the decision problem, and it is not clear how to proceed in checking whether T_0 "fits" into the attachment digraph. In fact, as we now show, this problem is NP-complete.

Theorem 8.31. *The problem* Single-Peaked Tree Labelling *is NP-complete even if* T_0 *is restricted to diameter at most four.*

Proof. The problem is in NP since for a given T and a given isomorphism ϕ, we can easily check that ϕ is an isomorphism and that the profile single-peaked on T.

For the hardness proof, we reduce from X3C. Suppose we are given an X3C-instance with ground set $X = \{x_1, \dots, x_p\}$ with $p = 3p'$ and a collection $\mathcal{Y} = \{Y_1, \dots, Y_q\}$ of 3-element subsets of X. We then construct an instance of Single-Peaked Tree Labelling as follows. First, we construct a tree T_0 with vertex set $C_0 = \{a, b_1, \dots, b_{q-p'}, c_1, \dots, c_{p'}\} \cup \{d_{i,j} : 1 \leqslant i \leqslant p', 1 \leqslant j \leqslant 3\}$, and edge set $E_0 = \{\{a, b_i\} : 1 \leqslant i \leqslant q - p'\} \cup \{\{a, c_i\} : 1 \leqslant i \leqslant p'\} \cup \{\{c_i, d_{i,j}\} : 1 \leqslant i \leqslant p', 1 \leqslant j \leqslant 3\}$. The resulting tree is drawn below; clearly it has diameter 4.



Next, we construct a profile P with $\lvert N\rvert = p + q$ voters on the candidate set $C = \{z, x_1, \dots, x_p, y_1, \dots, y_q\}$. P will contain one vote for each object and one vote for each set. In the following, all indifferences can be broken arbitrarily. For each object x_i, we add a voter v_{x_i}:

$$z \succ \{y_j : Y_j \ni x_i\} \succ x_i \succ \{y_j : Y_j \not\ni x_i\} \succ X \setminus \{x_i\}.$$

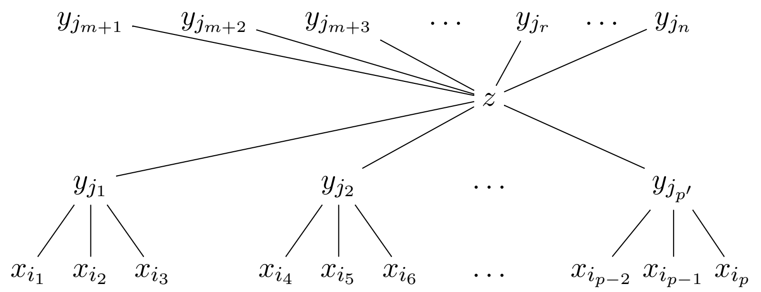
(This voter will force x_i to be attached to z or to a set containing x_i.) For each set Y_j, we add a voter v_{Y_j}:

$$z \succ y_j \succ \{y_j : j = 1, \dots, q\} \setminus \{y_j\} \succ X.$$

(This voter will force an edge from z to y_j.)

This completes the description of the reduction. We now prove that it is correct.

Suppose the X3C-instance is a "yes"-instance, and let $\mathcal{Y}' = \{Y_{j_1}, \dots, Y_{j_{p'}}\}$ be a cover with p' sets. Then we build a labelling isomorphism $\phi : C_0 \to C$ as follows: We put $\phi(a) = z$. For each $Y_j \in \mathcal{Y}'$, we take one of the b_i's and set $\phi(b_i) = y_j$. For each $Y_j \notin \mathcal{Y}'$, we take one of the c_i's and set $\phi(c_i) = y_j$. Finally, for each $x_k \in X$, we find $Y_j \in \mathcal{Y}'$ with $x_i \in Y_j$. Write $b_i = \phi^{-1}(y_j)$, then take one of the $d_{i,j}$'s and set $\phi(d_{i,j}) = x_k$. The resulting labelled tree T is shown below. It is easy to check that the profile P is single-peaked on T.



Conversely, suppose that there is a labelling T of T_0 so that P single-peaked on T. Let $\phi : C_0 \to C$ is a witnessing isomorphism. Note that the vertex labelled z must have degree at least q, because for each $j \in \{1, \dots, q\}$, voter v_{Y_j} can only be single-peaked on T if z and y_j are adjacent in T. There is only one such vertex in T, namely $\phi(a)$, and hence $\phi(a) = z$. The vertex $\phi(a)$ has exactly q neighbours, which then must all be labelled by some y_j. Exactly p' of the q neighbours of $\phi(a)$ have degree 4. Let $\mathcal{Y}' = \{Y_j \in \mathcal{Y} : y_j = \phi(c_i) \text{ for some } 1 \leqslant i \leqslant p'\}$ be the collection of the p' sets occupying the vertices with degree 4. We claim that $\mathcal{Y}'$ is a cover. Let $x_i \in X$; it must be a leaf of T because all internal vertices of T have already been labelled otherwise. Then, because v_{x_i} is single-peaked on T, the set $\{z, x\} \cup \{y_j : Y_j \ni x_i\}$ must be connected in T, so the neighbour of the leaf x_i must be a member of that set. But x_i cannot be a neighbour of z, so z is a neighbour of some y_j where $x \in Y_j$. This implies that y_j is the label of a degree-4 vertex. Hence $Y_j \in \mathcal{Y}'$, and so x_i is covered by $\mathcal{Y}'$. □

By copying the center vertex and adding some peripheral vertices, we can adjust this reduction to show that the problem remains hard even if T_0's maximum degree is three. Notice that the problem is (trivially) fixed-parameter tractable with parameter $k = |C|$ by trying all $k!$ possible labellings of the input tree.

8.10. Conclusions

Without any restrictions on the structure of voters' preferences, winner determination under the Chamberlin–Courant rule is NP-hard. Positive results have been obtained when preferences are assumed to be single-peaked, and we studied whether these results can be extended to preferences that are single-peaked on a tree. For the egalitarian variant of the rule, we showed that winner determination remains polynomial-time solvable for any tree and any scoring function. For the utilitarian setting, we show that winner determination is hard for general preferences single-peaked on a tree, but we find positive results when imposing additional restrictions. One algorithm we present runs in polynomial time when preferences are single-peaked on a tree which has a constant number of leaves, and another runs efficiently on a tree with a small number of internal vertices. It would also be interesting to see whether our easiness results for preferences that are single-peaked on a tree extend to the egalitarian version of Monroe's rule. Betzler et al. [2013] show that this rule becomes easy for preferences single-peaked on a path, but their argument is much more intricate than for egalitarian Chamberlin–Courant.

To make our parameterised winner determination algorithms more applicable, we have also studied the recognition problem in detail. We have designed polynomial-time algorithms for recognising profiles that are single-peaked on special classes of trees. One can interpret some of these algorithms as deciding whether the input profile is close to being single-peaked on a path, in the sense of being single-peaked on a tree which is similar to a path (for example, having small pathwidth). This is an alternative view on a recent literature about almost-structured preferences which has studied various distance measures [Faliszewski et al., 2014, Erdelyi et al., 2017, Bredereck et al., 2016, Cornaz et al., 2012].

9. Preferences Single-Peaked on Circles

We introduce the domain of preferences that are single-peaked on a circle, which is another generalisation of the single-peaked domain. This preference restriction is useful, for example, for scheduling decisions, certain facility location problems, and for one-dimensional decisions in the presence of extremist preferences. We give a fast recognition algorithm of this domain, and show that many popular single- and multiwinner voting rules are polynomial-time computable on this domain. In particular, we prove that Chamberlin–Courant and Proportional Approval Voting can be computed in polynomial time for profiles that are single-peaked on a circle. This result is achieved using special Integer Linear Programming formulations that become totally unimodular whenever the input profile is single-peaked or single-peaked on a circle. In contrast, Kemeny's rule remains hard to evaluate, and several impossibility results from social choice theory can be proved using only profiles in this domain.

9.1. Introduction

A central problem in the study of multi-agent systems is the aggregation of agents' preferences in order to make group decisions. Impossibility theorems and computational hardness result make this problem a hard one to solve. However, a successful line of research going back to Black's [1948] seminal article has managed to circumvent many problems in (computational) social choice for the special case when agents' preferences are *single-peaked*. Under this preference restriction, we assume that agents have preferences over the possible values of a one-dimensional quantity such as the timing of a deadline, a tax rate, a thermostat setting, or the price of a new product. A preference ordering is *single-peaked* if an agent has a certain most-preferred value of the quantity and derives less and less satisfaction from values that are further away from the subjective optimum. Another popular application of this setting is in political elections, where it is often held that candidates can be ordered on a left-to-right spectrum making the voters' preferences single-peaked.

Preference profiles that consist solely of single-peaked preference orderings have attractive properties, both algorithmically and in terms of their social choice behaviour [Elkind et al., 2016, 2017b]. For example, winner determination problems that are computationally hard in the general case tend to be easy when restricted to single-peaked profiles [Brandt et al., 2015, Betzler et al., 2013], and the single-peaked domain guarantees the existence of Condorcet winners as well as transitivity of the majority relation and thus admits a strategyproof voting rule [Moulin, 1988a].

The usefulness of results of this type is limited by the extent to which profiles in practice actually happen to be single-peaked. One way of dealing with this is to consider less restrictive generalisations of single-peakedness. Maybe the structure of the alternative space is not quite one-dimensional, and in this case it might be useful to consider preferences that are single-peaked on a *tree* [Demange, 1982]. As we saw in Chapter 8, this domain is notably larger, yet still retains many desirable properties in social choice terms; however, its algorithmic usefulness is more mixed.

In this chapter, we identify a new preference restriction: being single-peaked on a *circle*. Here we assume that alternatives can be placed on a circle, with agents' preferences again being

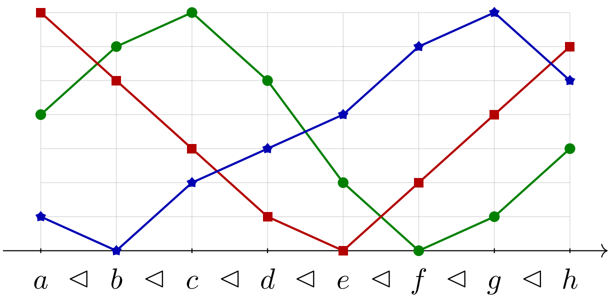


Figure 9.1.: Example of preferences single-peaked on a circle.

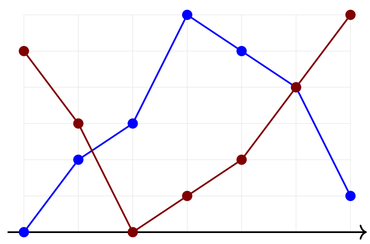
decreasing on both sides of their peaks. See Figure 9.1 for some example shapes that 'preference curves' might have in this setting; higher points are more preferred. Note that the circle wraps around, and so h and a are adjacent. Intuitively, a preference profile is single-peaked on a circle if, for every agent, we can 'cut' the cycle once so that the agent's preferences are single-peaked on the resulting line. Crucially, the location of the cutting point may differ for each agent.

The aim of this chapter is to explore this new preference domain in detail. We will find that this domain is algorithmically useful (it often allows for efficient winner determination), but it performs less convincingly in terms of axiomatic properties (since voting paradoxes still occur and impossibility results can still be proved). Interestingly, this is precisely opposite to how the results turned out for single-peakedness on trees.

	single-peaked	single-peaked on a tree	single-peaked on a circle
axiomatic usefulness	++	+	−−
algorithmic usefulness	++	−	+

Table 9.1.: Rough comparison of the virtues of different domain restrictions, from very high usefulness (++) to very low (−−).

Motivating Examples



There are many practical scenarios where we might expect preferences to be single-peaked on a circle. This is even the case when, on first sight, there seems to be no circle anywhere. Indeed, suppose that alternatives are naturally ordered on a line; we may pretend this line is a circle by joining up its endpoints. Of course, every order that is single-peaked on the line is also single-peaked on the circle. But crucially, the *reverse* of such an order, now single-*caved* on the line, is still single-peaked on the same circle. Thus, our new preference restriction allows us to *combine* single-peaked and single-caved votes (as shown on the right). One interpretation is that this move allows us to accommodate "extremists". For example, while most people have a sweet spot somewhere on the left-right political axis, some people might dislike centrist options and prefer the extremes. When planning a vacation, some might have an optimal climate in mind, while others like it both very cold (skiing) and very hot (beaches), but dislike compromises (England).

Other examples of alternative spaces are more explicitly cyclic. Consider, for example, finding a good time for a daily event (such as a day or night shift, or a meeting, or the timing of backups) where possibilities are arranged in a 24-hour cycle. A similar structure exists when scheduling an international phone call; here, different time zones are arranged along the equator, and lead to cyclic preferences.

But perhaps the most appealing example of preferences that can be expected to be single-peaked on a circle come from problems inspired by facility location. Rather many structures have a boundary that is (roughly) isomorphic to a cycle, including most cities and countries. The problem of deciding where to locate a new airport for a city would be one example, since airports are usually positioned on the boundary. Similarly, where should a company build new office space? To which coastal region should a family move? Where do we want to sit in a football stadium? Another plausible application could be inspired by security concerns, if we consider the placement of border security checkpoints.

Contributions

The main results in this chapter can be summarised as follows:

- We formally define single-peakedness on circles and immediately extend this definition to preferences with ties, and to dichotomous (approval) preferences.

- We show that it is possible to efficiently recognise whether a given preference profile is single-peaked on some circle, and if so return a suitable circle. For the case of preferences without ties, we give a recognition algorithm that runs in linear time, matching the performance in the case for the line.

- While single-peakedness on a line serves as a way to circumvent many impossibility results in social choice, we show that such impossibilities (including the Gibbard–Satterthwaite theorem) can still be proved when preferences are allowed to be single-peaked on a circle.

- We then study the algorithmic properties of our new preference extension. We show that Young's voting rule (and also Young *scores*) can be efficiently computed if preferences are single-peaked on a circle; this algorithm also improves upon the state-of-the-art when it comes to preferences single-peaked on a line. In contrast, we show that Kemeny's rule is NP-hard to compute even in this restricted domain.

- Finally, we show that several multiwinner voting rules are efficiently computable in our restricted case, specifically all that are included in the large class of so-called OWA-based rules. This class includes, e.g., the *Chamberlin–Courant* rule and *Proportional Approval Voting* (PAV). It is noteworthy that some of these algorithmic results have not yet been established even for single-peaked profiles (such as the one for PAV). This general result relies on using total unimodularity and integer programming.

9.2. Definition

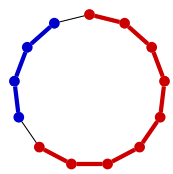
Let A be a finite set of *alternatives* (or *candidates*). A *weak order* (or *preference relation*) is a binary relation $\succcurlyeq$ over A which is complete and transitive. A *linear order* is a weak order that is antisymmetric, and so does not allow preference ties; a *strict* linear order $\succ$ is the irreflexive part of a linear order. A *profile* $P = (v_1, \dots, v_n)$ over A is a list of weak orders over A. The elements of $N = \{1, \dots, n\}$ are called *voters*, and we associate voter $i \in N$ with the order v_i, which we call the *vote* of voter i. For convenience, we write $a \succcurlyeq_i b$ whenever $(a, b) \in v_i$, i.e., when voter i weakly prefers alternative a to alternative b. We also use $\succ_i$ and $\sim_i$ for the strict and indifference parts of $\succcurlyeq_i$. We will always write m for the number of alternatives and n for the number of voters. If v_i is a linear order, we write $\text{top}(v_i)$ for i's most-preferred alternative.

An *axis* $\lhd$ is a strict linear order of the alternatives. We usually think of an axis as describing the underlying one-dimensional structure of the alternative space. A linear order v_i is *single-peaked* with respect to the axis $\lhd$ if for each pair of alternatives $a, b \in A$ with $\text{top}(v_i) \lhd b \lhd a$ or $a \lhd b \lhd \text{top}(v_i)$ it holds that $b \succ_i a$. Let us also give another, equivalent definition. An *interval* $I \subseteq A$ of an axis $\lhd$ is any set such that for all $a, b, c \in A$, if we have $a, c \in I$ and $a \lhd b \lhd c$, then $b \in I$. Then a vote v_i is single-peaked with respect to the axis $\lhd$ if and only if for every $c \in A$, the top-initial segment $\{a \in A : a \succ_i c\}$ is an interval of $\lhd$. This definition in terms of intervals immediately gives a definition of the single-peaked property for *weak* orders as well. There are several possible definitions of single-peakedness for weak orders; the one we use here is often referred to as *possible single-peakedness* [Lackner, 2014], since it is equivalent to saying that there exists a linearisation of the weak order which is single-peaked.

We say that two axes $\lhd$ and $\lhd'$ are *cyclically equivalent* if there is $l \in [m]$ such that we can write $a_1 \lhd a_2 \lhd a_3 \lhd \cdots \lhd a_m$ and $a_l \lhd' a_{l+1} \lhd' \cdots \lhd' a_m \lhd' a_1 \lhd' \cdots \lhd' a_{l-1}$. For an axis $\lhd$, we then define the *circle* $C(\lhd)$ *of* $\lhd$ to be the set of axes cyclically equivalent to $\lhd$. Any set C of axes that can be written as $C = C(\lhd)$ for some $\lhd$ we call a *circle*. For example, $C = \{a \lhd b \lhd c, b \lhd' c \lhd' a, c \lhd'' a \lhd'' b\}$ is a circle. Note that "cutting" a circle C at a point yields an axis $\lhd \in C$. We say that $\lhd$ *starts in* $a \in A$ if $a \lhd b$ for all $b \in A \setminus \{a\}$.

Definition 9.1. Let C be a circle. A vote v_i is *single-peaked on* C if there is an axis $\lhd \in C$ such that v_i is single-peaked with respect to $\lhd$. A preference profile P is *single-peaked on a circle (SPOC)* if there exists a circle C such that every vote $v_i \in P$ is single-peaked on C.

Intuitively, a vote v_i is single-peaked on C if C can be cut so that v_i is single-peaked on the resulting line.



Again let us state another equivalent definition. An *interval* $I \subseteq A$ of a circle C is a set that is an interval of one of the axes $\lhd \in C$ of the circle. Then a vote is single-peaked on a circle C if and only if each top-initial segment $\{a \in A : a \succ_i c\}$ is an interval of C. Note that the complement $A \setminus I$ of an interval I of C is again an interval. Thus, a weak order $\succcurlyeq$ is single-peaked on C if and only if its reverse $\overleftarrow{\succcurlyeq} = \{(b, a) : (a, b) \in \succcurlyeq\}$ is also single-peaked on C.

A vote is *single-caved* if its reverse is single-peaked. It follows, then, that mixtures of single-peaked and single-caved orders (on the same axis) are SPOC. However, not all SPOC profiles have this form; one such example is the profile shown in Figure 9.1, where the circle cannot be cut so as to make every preference curve either single-peaked or single-caved.

9.3. Recognition Algorithms

In this section we design algorithms that decide whether a given profile is single-peaked on some circle, and if so, return a suitable circle C.

A matrix $M = (a_{ij})$ with $a_{ij} \in \{0, 1\}$ has the *consecutive ones property* if the columns of M can be put into a linear order $\lhd$ so that for every row of M, the columns with 1-entries form an interval of $\lhd$. The matrix M has the *circular ones property* if its columns can be arranged in a circle C so that the 1-entries of each row form an interval of C. Given our definitions in terms of intervals above, it is straightforward to translate a profile P of weak orders into an $mn \times m$ matrix M so that P is single-peaked [single-peaked on a circle] if and only if M has the consecutive [circular] ones property [Bartholdi, III and Trick, 1986]: Take one column for each alternative, and one row for every top-initial segment of every voter in P; the row is the incidence vector of the segment. Since it is possible to check in linear time whether a matrix A has the consecutive or circular ones property [Booth and Lueker, 1976], this gives us an $O(m^2 n)$ algorithm to recognise profiles that are single-peaked on a circle.

In the remainder of this section, we design a more explicit algorithm that runs in time $O(mn)$ when the input profile consists of *linear* orders.[1]

Suppose $P = (v_1, \dots, v_n)$ is a profile of linear orders over A, and fix some alternative $z \in A$. We will build another profile $P' = (v_1^u, v_1^l, \dots, v_n^u, v_n^l)$ of $2n$ *weak* orders by *slicing* each vote v_i *at* z into an *upper* part v_i^u and a *lower* part v_i^l. The upper part v_i^u ranks all alternatives a such that $a \succ_i z$ in order of $\succ_i$, and puts all remaining alternatives into a least-preferred indifference class. The lower part v_i^l ranks all alternatives a such that $z \succ_i a$ in *reverse* order of $\succ_i$, and again puts all remaining alternatives into a least-preferred indifference class.

Example 9.2. Slicing the order $a \succ b \succ c \succ z \succ d \succ e \succ f$ at z yields the upper part $a \succ^u b \succ^u c \succ^u z \sim^u d \sim^u e \sim^u f$ and the lower part $f \succ^l e \succ^l d \succ^l z \sim^l a \sim^l b \sim^l c$.

The notion of slicing reduces SPOC to single-peakedness:

Proposition 9.3. *Suppose a profile P' of weak orders is obtained by slicing each vote of a profile P of linear orders at some fixed $z \in A$. Then P is SPOC if and only if the profile P' is single-peaked.*

Proof. Suppose P is SPOC on C, and let $\lhd \in C$ be an axis starting in z. Since z is least-preferred by all voters in P', z is not contained in any top-initial segment of any voter in P'. However, all top-initial segments of votes in P' are intervals of C. Since they do not contain z, they must also be intervals of $\lhd$. Thus, P' is single-peaked with respect to $\lhd$.

Suppose P' is single-peaked with respect to $\lhd$. We show that P is SPOC on $C = C(\lhd)$. Take a top-initial segment S of a vote v_i in P; we prove that S is an interval of C. If $z \notin S$, then S is a top-initial segment of v_i^u in P'. Thus, S is an interval of $\lhd$ and so an interval of C. If however $z \in S$, then the complement $A \setminus S$ is a top-initial segment of v_i^l in P', hence an interval of $\lhd$, and so $A \setminus S$ is an interval of C. But the complement of an interval of a circle is again an interval, and so S is an interval of C. Hence P is SPOC. □

Thus, we can use an algorithm that decides whether a profile of weak orders is single-peaked to decide whether a profile of linear orders is SPOC. Next, note that if we select $z \in A$ to be the alternative that is ranked last by v_1 (say), then the profile P' obtained by slicing P at z contains a linear order (namely the upper part of v_1). Lackner [2014] has given a $O(mn)$ time algorithm that decides whether a profile of weak orders containing at least one linear order is single-peaked. Since P' can be constructed from P in time $O(mn)$, by Proposition 9.3, we obtain the following.

Theorem 9.4. *There is an $O(mn)$ time algorithm that decides whether a profile of linear orders is single-peaked on a circle.*

9.4. Impossibility Theorems

One of the major advantages of the traditional single-peaked domain is the existence of a non-manipulable voting rule on this domain: The well-known median voter procedure sorts voters' most preferred alternatives according to the axis $\lhd$ and then returns the median alternative a. This alternative is, in fact, a (weak) Condorcet winner: for any other alternative b, a (weak) majority of voters prefers a to b. One might hope to be able to extend this procedure to circles, but this turns out to be impossible: the Gibbard–Satterthwaite theorem can be proved using only profiles that are single-peaked on a circle.

A *resolute voting rule* f on SPOC profiles is a function assigning a single winning alternative to every SPOC profile of linear orders. The rule f is *non-dictatorial* if there is no fixed voter i such that f always picks i's top alternative. The profile obtained from P by replacing vote v_i by

[1] Actually, the algorithm works whenever P contains at least one linear order.

v_i' is denoted by (P_{-i}, v_i'). A voting rule f on SPOC profiles is *strategyproof* if $f(P) \succcurlyeq_i f(P_{-i}, v_i')$ for all orders v_i' such that (P_{-i}, v_i') is still SPOC.

Theorem 9.5 (Gibbard–Satterthwaite Theorem for SPOC)**.** *There is no resolute voting rule on SPOC profiles that is non-dictatorial, onto, and satisfies strategyproofness.*

Proof. This follows immediately from the results of Kim and Roush [1980] and Sato [2010], who prove this result for an even more restricted domain consisting only of the $2m$ orders which traverse the circle clockwise and counter-clockwise starting from every possible alternative. □

Note that the SPOC orders used in this proof are 'unbalanced', in that the most- and least-preferred alternatives are adjacent on the circle for every agent. Still, a similar dictatorship result can be proved even using orders that are 'Euclidean' on a circle, where preferences decrease uniformly in both directions from the peak [Schummer and Vohra, 2002]. It can also be shown that, with these Euclidean orders, the *random dictatorship* rule is *group*-strategyproof [Alon et al., 2010b], and there is an intriguing randomised mechanism that is strategyproof and provides a 3/2-approximation to the egalitarian social welfare [Alon et al., 2010a].

Theorem 9.5 is the only impossibility theorem in this thesis that is not proved with the help of SAT solvers. The reason is that the theorem uses two global axioms (non-dictatorial and onto) and this rules out small MUSes.

Another desirable axiomatic property is *participation*, which, intuitively, states that no voter can strictly benefit by abstaining from an election. A celebrated result of Moulin [1988b] shows that this property is incompatible with Condorcet-consistency, which requires the voting rule to select the Condorcet winner if one exists (see Chapter 1). This result can also be proved using only SPOC profiles.

Theorem 9.6 (No-Show Paradox for SPOC)**.** *For $m \geqslant 4$ and $n \geqslant 12$, there is no resolute voting rule on SPOC profiles that is Condorcet-consistent and satisfies participation.*

Proof. For $m = 4$ alternatives, one can check that all preference profiles in the proof of Theorem 1.5 are single-peaked on the circle induced by $a \lhd b \lhd d \lhd c \lhd a$. For $m \geqslant 5$, one can extend the proof by replacing, in every profile of the proof, the alternative a by $m-3$ clones $a_1, \dots, a_{m-3}$ where each agent ranks them as $a_1 \succ \cdots \succ a_{m-3}$. All resulting profiles are single-peaked on the circle induced by $a_1 \lhd \cdots \lhd a_{m-3} \lhd b \lhd d \lhd c \lhd a_1$. The proof of Theorem 1.5 goes through after this change essentially verbatim. □

As described in the next section, further impossibilities about tournament-based rules can be deduced from Lemma 9.7.

9.5. Kemeny's and Young's Rules

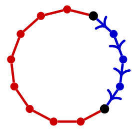
In this section, we will consider the problem of determining an election winner according to two well-known voting rules, Young's rule and Kemeny's rule, that are NP-hard to evaluate in general [Bartholdi, III et al., 1989, Rothe et al., 2003, Hemaspaandra et al., 2005]. We will be interested to see whether these problems can be solved in polynomial time for SPOC profiles. We leave the complexity of Dodgson's rule for SPOC profiles for future work.

Kemeny's rule

Kemeny's rule is a *rank aggregation* rule: Given a profile P over A, its aim is to produce a *consensus* ranking over A. Suppose r is a linear order over A. Its *Kemeny score* is $\sum_{i \in N} |v_i \cap r|$, the number of pairwise agreements of r with P. A *Kemeny ranking* is a linear order r with maximum Kemeny score. While it is NP-hard to find a Kemeny ranking Bartholdi, III et al. [1989], this problem is easy for single-peaked profiles whose transitive majority relation is easily seen to give rise to a Kemeny ranking. For SPOC preferences, the situation is less clear: the Condorcet paradox profile $(x \succ_1 y \succ_1 z, y \succ_2 z \succ_2 x, z \succ_3 x \succ_3 y)$ on 3 alternatives is SPOC, so SPOC does not guarantee a transitive majority relation. In fact, SPOC does not guarantee *anything at all* about the majority relation.

Lemma 9.7 (McGarvey's theorem for SPOC)**.** *All (weighted) majority tournaments are inducible by SPOC profiles.*

Proof. Fix a circle C with $x_1 \lhd x_2 \lhd \cdots \lhd x_m$. For any arc (x_i, x_j) of the target tournament, consider the following two votes which are single-peaked on C: (subscripts modulo m)



$$x_{i+1} \succ \cdots \succ x_{j-1} \succ x_i \succ x_j \succ x_{j+1} \succ \cdots \succ x_{i-1}$$
$$x_{x-1} \succ \cdots \succ x_{j+1} \succ x_i \succ x_j \succ x_{j-1} \succ \cdots \succ x_{i+1}$$

These two votes induce a majority arc $x_i \to x_j$ with weight 2, but all other arcs have weight 0. By combining pairs of such votes, any tournament can be obtained. If odd edge weights are desired, start with an arbitrary single order, and then use pairs as above to adjust the weights as needed. □

Recall that Kemeny scores only depend on the weighted majority relation of a profile. Since the profiles in the proof of McGarvey's theorem above can be produced in polynomial time, the hardness of Kemeny in the general case carries over.

Theorem 9.8. *Finding a Kemeny ranking is NP-hard, even for SPOC preferences.*

Indeed, by the same argument essentially all negative (axiomatic or computational) results in the sphere of voting rules based on (weighted) tournaments (see Brandt et al., 2016a, Fischer et al., 2016) still hold restricted to SPOC preferences.

Young's rule

Given a profile P over A, an alternative $c \in A$ is a *Condorcet winner* if for every $b \in A \setminus \{c\}$, a majority of voters in P strictly prefers c to b. The *Young score* of an alternative $c \in A$ is the minimum number of voters that have to be deleted from P so that c becomes a Condorcet winner. Then, Young's rule selects all alternatives with minimum Young score as winners. It is known that Young winners can be found in polynomial time for single-peaked preferences [Brandt et al., 2015], since in this case Condorcet winners always exists when the number of voters n is odd; and the case with n even is also handled easily.

Because SPOC does not guarantee the existence of a Condorcet winner, a different approach is needed. We will use the interpretation of SPOC in terms of intervals of the underlying circle to give a polynomial-time algorithm that calculates the Young score of every alternative; clearly this algorithm can then be run repeatedly to find a Young winner. Of course, our algorithm also works for preferences single-peaked on a line; while the algorithm of Brandt et al. [2015] returns only a Young winner, our algorithm can find the Young score of *any* alternative. Note that precise definitions of Young scores differ slightly: sometimes it is only required that an alternative be made a *weak* Condorcet winner through voter deletion; our algorithm can be easily adapted for this alternative definition.

Theorem 9.9. *For SPOC profiles, the Young score of an alternative can be computed in $O(mn^2)$ time.*

Proof. We fix an axis $\lhd \in C$ that starts with the alternative a whose Young score we want to compute; let $a \lhd b \lhd \cdots \lhd c$ (b is the candidate right of a, c is the rightmost candidate). We partition voters into two sets: $N_1 = \{i \in N : b \succ_i a\}$ and $N_2 = N \setminus N_1$. Since P is SPOC, for any voter i, the set $I_i := \{d \in A : d \succ_i a\}$ forms an interval of $\lhd$. Voters in N_1 correspond to intervals containing b; voters in N_2 correspond to intervals containing c but not b, and to empty intervals. Figure 9.2 illustrates the situation: For each voter i, an arc indicates the set I_i. The red arcs on the right belong to voters from N_1, and the blue arcs on the left belong to voters from N_2.

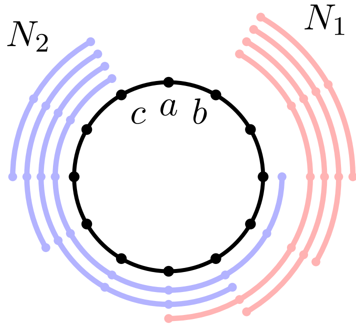


Figure 9.2.: Illlustration of the proof of Theorem 9.9, for a profile with eight voter. For each voter, an arc indicates the set of alternatives preferred to a.

The idea behind our algorithm is that if there are voters i and j with $I_i \subseteq I_j$, then it is at least as profitable (for purposes of making a the Condorcet winner) to remove voter j as to remove voter i. Now note that the intervals I_i of voters in N_1 are nested by set inclusion, and similarly for voters in N_2. Thus, we let N_1^{-r} and N_2^{-s} denote the subsets of N_1 and N_2 obtained by deleting, respectively, the r and s voters from N_1 and N_2 that have the r and s largest (with respect to set inclusion) intervals I_i. Because of the nesting property, *if* there is a way of deleting r and s voters from N_1 and N_2 that makes a the Condorcet winner, then the deletions giving N_1^{-r} and N_2^{-s} also make a the Condorcet winner.

These observations suggest the following simple algorithm: For every pair (r, s) with $0 \leqslant r \leqslant |N_1|$ and $0 \leqslant s \leqslant |N_2|$, we check whether a is the Condorcet winner in $N_1^{-r} \cup N_2^{-s}$. We return a pair (r^*, s^*) with $r^* + s^*$ minimum for which this is the case. Then the Young score of a is $r^* + s^*$. If no such pair exists, the Young score of a is infinite.

To see that this algorithm can be run in $O(mn^2)$ time, we show how to check in $O(m)$ time whether a is the Condorcet winner in $N_1^{-r} \cup N_2^{-s}$. To do so, we precompute for every $x \in A \setminus \{c\}$, $0 \leqslant r \leqslant |N_1|$, and $0 \leqslant s \leqslant |N_2|$ the numbers

$$d_r^1(x) = |\{i \in N_1^{-r} : a \succ_i x\}| - |\{i \in N_1^{-r} : x \succ_i a\}|,$$
$$d_s^2(x) = |\{i \in N_2^{-s} : a \succ_i x\}| - |\{i \in N_2^{-s} : x \succ_i a\}|.$$

Note that a is a Condorcet winner in $N_1^{-r} \cup N_2^{-s}$ if and only if for all $x \in A \setminus \{c\}$ it holds that $d_r^1(x) + d_s^2(x) > 0$. The quantities $d_r^1(x)$ and $d_s^2(x)$ can be precomputed in $O(mn^2)$ time. Verifying whether $d_r^1(x) + d_s^2(x) > 0$ requires constant time and hence $O(m)$ time for every $x \in A \setminus \{c\}$. □

9.6. Multiwinner Elections

Much recent work has studied voting rules that select not a single winner, but a *committee* $W \subseteq A$ of candidates, where $|W| = k$ has some desired size k (see, e.g., a recent survey by

Faliszewski et al., 2017a). Depending on the context, we may wish this committee to have different properties. For example, we may aim for a representative committee in which as many voters as possible have a good representative, or we may aim for a proportional committee in which subgroups of the voters are represented by committee members in proportion to the subgroup size. Many of the commonly-studied multiwinner rules optimise an objective function over the set of all committees. Unsurprisingly, many of them are NP-hard to evaluate. In this section, we show that several popular rules can be evaluated in polynomial time when preferences are single-peaked on a circle.

In Chapter 8, we studied a rule of Chamberlin and Courant [1983] that aims for a committee that represents as many voters as well as possible. It is usually defined for profiles of linear orders. Recall that according to this rule, each voter i is *represented* by i's favourite (highest-ranked) alternative in W; suppose this is $c_i \in W$. Then, we take the 'utility' of voter i to be the Borda score (i.e., position counting from the bottom) of c_i in i's ranking. The Chamberlin–Courant rule selects a committee of size k that maximises the sum of voter utilities. By replacing Borda scores by other scoring vectors, we obtain a whole family of rules. The class of OWA-based rules, as defined below, is a further generalisation of this idea.

Finding a winning committee under the Chamberlin–Courant rule is known to be NP-hard for Borda scores [Lu and Boutilier, 2011]. Betzler et al. [2013] showed that this problem becomes easy when the input profile is single-peaked. Their algorithm starts by running a recognition algorithm for single-peakedness on the input to obtain an underlying axis $\lhd$ on which the profile is single-peaked. Then, they run a dynamic programming algorithm which constructs an optimal committee. Roughly, this dynamic program successively considers left prefixes of the axis $\lhd$, and constructs an optimal committee using only candidates from the prefix. Unfortunately, it is unclear how to extend this approach to preferences single-peaked on a circle, since a circle does not have a left endpoint where we could start the dynamic program.

Thus, we follow a different approach: We design an integer linear programming (ILP) formulation encoding the winner determination problem. We then show that the matrix of coefficients appearing in the constraints of this ILP is *totally unimodular* whenever the input profile is SPOC. A well-known result states that ILPs with totally unimodular constraint matrices are optimally solved by their LP relaxations [Hoffman and Kruskal, 1956], and can thus be solved in polynomial time.

This approach works not only for the Chamberlin–Courant rule, but for a large class of rules introduced by Skowron et al. [2016], called *OWA-based rules* (OWA stands for *ordered weighted average*). Let us describe this class of rules. We will give a definition that works for weak order inputs, and so this class includes rules that work both for linear order profiles, and for approval profiles. Given a preference profile, as a first step the rule converts preferences into numerical scores, using a positional scoring system. Let us describe formally how this is done. Suppose that $\succcurlyeq$ is a weak order over A. Then we can uniquely partition $A = A_1 \cup \cdots \cup A_q$ into disjoint non-empty sets such that $A_1 \succ \cdots \succ A_q$ and such that $a \sim b$ for all $a, b \in A_r$ for $r \in [q]$. The sets A_r are called the *indifference classes* of the weak order $\succcurlyeq$. Now, for an alternative $a \in A$, if $a \in A_r$, the *rank* of a in $\succcurlyeq$ is r, and we write $\text{rank}_{\succcurlyeq}(a) = r$. Thus, the alternatives with rank 1 are the most-preferred alternatives. If we are given a profile P, then we write $\text{rank}_i(a)$ for $i \in N$ and $a \in A$ for the rank of a in voter i's preferences. A *score vector* is a vector $\mathbf{s} \in \mathbb{R}^m$ such that $s_1 \geqslant s_2 \geqslant \cdots \geqslant s_m$. Common examples are $\mathbf{s} = (m-1, m-2, \dots, 0)$ for Borda scores and $\mathbf{s} = (1, 0, \dots, 0)$ for plurality scores. Given such a score vector $\mathbf{s}$, we say that voter $i \in N$ assigns the score $s_{\text{rank}_i(a)}$ to alternative $a \in A$, and we write $\mathbf{s}(i, a) = s_{\text{rank}_i(a)}$. This is the standard definition when preferences are given by linear orders. If a voter submits an approval ballot, and we use plurality scores, then the voter assigns score 1 to all approved alternatives and score 0 to the remaining alternatives. Note that whenever $a \succcurlyeq_i b$ then $\mathbf{s}(i, a) \geqslant \mathbf{s}(i, b)$.

The utility a voter derives from a committee under an OWA-based rule will be a linear combination of the scores assigned to the candidates in the committee, and these values are calculated using an OWA operator. A weight vector $\boldsymbol{\alpha} \in \mathbb{R}^k$ defines an *ordered weighted average* (OWA) operator as follows: Given any vector $\mathbf{x} \in \mathbb{R}^k$, first sort the entries of $\mathbf{x}$ into non-increasing order, so that $x_{\sigma(1)} \geqslant \ldots \geqslant x_{\sigma(k)}$; second, apply the weights: the ordered weighted average of $\mathbf{x}$ with weights $\boldsymbol{\alpha}$ is given by $\boldsymbol{\alpha}(\mathbf{x}) := \sum_{i=1}^k \alpha_i x_{\sigma(i)}$. For example, if $\boldsymbol{\alpha} = (1, 0, \ldots, 0)$, then $\boldsymbol{\alpha}(\mathbf{x}) = x_{\sigma(1)} = \max_{i\in[k]} x_i$, so that this operator returns the maximum of the vector $\mathbf{x}$. Alternatively, if $\boldsymbol{\alpha} = (1, 1, \ldots, 1)$, then $\boldsymbol{\alpha}(\mathbf{x}) = \sum_{i=1}^k x_{\sigma(i)} = \sum_{i=1}^k x_i$, so that this operator gives the sum of the numbers in $\mathbf{x}$.

Given a profile P, a scoring vector $\mathbf{s}$, and an OWA operator $\boldsymbol{\alpha}$, we define the utility of a committee $W = \{c_1, \ldots, c_k\}$ as

$$U(\mathbf{s}, \boldsymbol{\alpha}, W) = \sum_{i\in N} \boldsymbol{\alpha}(\mathbf{s}(i, c_1), \ldots, \mathbf{s}(i, c_k)).$$

Then the OWA-based multiwinner rule based on $\mathbf{s}$ and $\boldsymbol{\alpha}$ outputs a committee W of size k for which $U(\mathbf{s}, \boldsymbol{\alpha}, W)$ is maximum.

For example, choosing $\boldsymbol{\alpha} = (1, 0, \ldots, 0)$ and $\mathbf{s} = (m-1, m-2, \ldots, 0)$ (Borda scores) gives us the Chamberlin–Courant rule, where each voter derives as utility the score of their favourite committee member. Choosing $\boldsymbol{\alpha} = (1, 1, 0, \ldots, 0)$ gives us an analogue of Chamberlin–Courant where voters obtain as utility the sum of the scores of their favourite *two* members of the committee (this rule is sometimes known as 2-Borda). An OWA-based rule applied to profiles with approval votes with $\boldsymbol{\alpha} = (1, \frac{1}{2}, \ldots, \frac{1}{k})$ and plurality scores $\mathbf{s} = (1, 0, 0, \ldots)$ gives us Proportional Approval Voting (PAV). Thus, OWA-based rules generalise both Chamberlin–Courant and PAV.

Our polynomial-time result will only work for *non-increasing* OWA vectors with $\alpha_1 \geqslant \cdots \geqslant \alpha_k$. For example, this excludes the rule where voters are represented by their *least*-favourite committee member, or by their median committee member. While such rules may be sensible in some contexts [Skowron, 2015], this restriction seems mild for most contexts.

Next, let us give an overview about total unimodularity. A matrix $A = (a_{ij})_{ij} \in \mathbb{Z}^{m\times n}$ with $a_{ij} \in \{-1, 0, 1\}$ is called *totally unimodular* if every square submatrix B of A has $\det B \in \{-1, 0, 1\}$. (The rows and columns of B need not occur contiguously in A.) The following results are well-known. Proofs and much more about their theory can be found in the textbook by Schrijver [1998].

Theorem 9.10 (see Schrijver, 1998, Theorem 19.1)**.** *Suppose $A \in \mathbb{Z}^{m\times n}$ is a totally unimodular matrix, $b \in \mathbb{Z}^m$ is an integral vector of right-hand sides, and $c \in \mathbb{Q}^n$ is an objective vector. Then the linear program*

$$\text{max } c^T x \text{ subject to } Ax \leqslant b \tag{P}$$

has an integral optimum solution, which is a vertex of the polyhedron $\{x : Ax \leqslant b\}$. Thus, the integer linear program

$$\text{max } c^T x \text{ subject to } Ax \leqslant b, x \in \mathbb{Z}^n \tag{IP}$$

is solved optimally by its linear programming relaxation (P).

An optimum solution to (IP) can be found in polynomial time [Maurras et al., 1981, Tardos, 1986]. We will now state some elementary results about totally unimodular matrices, which allows us to build new matrices from old.

Proposition 9.11 (see Schrijver, 1998, chapter 19)**.** *If A is totally unimodular, then so is*

1. *its transpose A^T,*
2. *the matrix $[A \mid -A]$ obtained from A by appending the negated columns of A,*

3. *the matrix $[A \mid I]$ where I is the identity matrix,*

4. *any matrix obtained from A through permuting or deleting rows or columns.*

In particular, from (3) and (4) it follows that appending a unit column $(0, \dots, 1, \dots, 0)^T$ will not destroy total unimodularity. Further, using these transformations, we can see that Theorem 9.10 remains true even if we add to (P) constraints giving lower and upper bounds to some variables, if we replace some of the inequality constraints by equality constraints, or change the direction of an inequality.

A binary matrix $A = (a_{ij}) \in \{0,1\}^{m \times n}$ has the *strong consecutive ones property* if the 1-entries of each row form a contiguous block, as in the example on the right. A binary matrix has the *consecutive ones property* if its columns can be permuted so that the resulting matrix has the strong consecutive ones property. The key result that will allow us to connect single-peaked preferences to total unimodularity is as follows:

$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Proposition 9.12 (see Schrijver, 1998, page 279)**.** *Every binary matrix with the consecutive ones property is totally unimodular.*

We remark that by a celebrated result of Seymour [1980], it is possible to decide in polynomial time whether a given matrix is totally unimodular, though we do not use this fact.

We are now ready to prove our main result, that OWA-based rules are easy to compute for SPOC profiles.

Theorem 9.13. *Given a SPOC profile P, and an OWA-based rule specified by a scoring vector $\mathbf{s}$ and a non-increasing OWA operator $\boldsymbol{\alpha}$, a winning committee can be found in polynomial time.*

Proof. We begin by showing the result for single-peaked profiles, and later show how to modify the argument for SPOC profiles.

Let P be a preference profile, and let k be the target committee size. Consider the following integer linear program, whose optimal solutions correspond to winning committees under the OWA-based rule with operator $\boldsymbol{\alpha}$ and scoring vector $\mathbf{s}$. In the program, for each $r = 2, \dots, m$, we write $s'_r = s_r - s_{r-1}$, and we write $s'_1 = s_1$. Thus, for every $r \in [m]$, we have that $s_r = \sum_{p=1}^{r} s'_p$.

$$\text{maximise} \sum_{i \in N} \sum_{\ell \in [k]} \sum_{r \in [m]} \alpha_\ell \cdot s'_r \cdot x_{i,\ell,r} \qquad \text{(OWA-ILP)}$$

$$\text{subject to} \sum_{c \in A} y_c = k \qquad (2)$$

$$\sum_{\ell \in [k]} x_{i,\ell,r} \leqslant \sum_{c \,:\, \mathrm{rank}_i(c) \leqslant r} y_c \qquad \text{for } i \in N,\ r \in [m] \qquad (3)$$

$$x_{i,\ell,r} \in \{0,1\} \qquad \text{for } i \in N,\ \ell \in [k],\ r \in [m]$$

$$y_c \in \{0,1\} \qquad \text{for } c \in A$$

Every feasible solution $((x_{i,\ell,r})_{i,\ell,r}, (y_c)_c)$ to the ILP corresponds to a committee $W = \{c \in A : y_c = 1\}$. Due to the constraint $\sum_{c \in A} y_c = k$, we have that $|W| = k$, so this is a committee of the required size. Next suppose that $S = ((x_{i,\ell,r})_{i,\ell,r}, (y_c)_c)$ is an optimal solution to the ILP. We may assume that under S, all constraints (3) are satisfied with equality, since otherwise we could set additional variables $x_{i,\ell,r}$ to 1 without affecting feasibility, and without lowering the objective value of the solution (because the coefficient $\alpha_\ell \cdot s'_r$ of $x_{i,\ell,r}$ is non-negative). Further, this operation does not change the committee W. Now fix a voter $i \in N$ and a rank $r \in [m]$. Suppose that there are L candidates in W which voter i places in rank r or better, i.e.,

$$L = |W \cap \{c \in A : \text{rank}_i(c) \leqslant r\}|.$$

By our assumption that the constraint (3) is satisfied in S with equality, exactly L of the variables $x_{i,\ell,r}$ for $\ell \in [k]$ are set to 1 in S. By our assumption that the vector $\boldsymbol{\alpha}$ is non-increasing, the coefficients $\alpha_\ell \cdot s'_r$ of $x_{i,\ell,r}$ in the objective function are non-increasing as ℓ goes from 1 to k. Hence, we may assume without loss of generality that in S, we have

$$x_{i,1,r} = \cdots = x_{i,L,r} = 1 \quad \text{and} \quad x_{i,L+1,r} = \cdots x_{i,k,r} = 0.$$

Then it follows that for $i \in N$, $\ell \in [k]$, $r \in [m]$, we have

$$x_{i,\ell,r} = 1 \quad \text{if and only if} \quad W \text{ contains at least } \ell \text{ candidates } c \text{ with } \text{rank}_i(c) \leqslant r.$$

Fix $i \in N$ and $\ell \in [k]$. Write $W = \{c_1, \dots, c_k\}$ so that $c_1 \succcurlyeq_i \cdots \succcurlyeq_i c_k$. Then it follows that, for every $r \in [m]$,

$$x_{i,\ell,r} = 1 \quad \text{if and only if} \quad \text{rank}_i(c_\ell) \leqslant r.$$

(If $x_{i,\ell,r} = 1$, then W contains at least ℓ candidates c with $\text{rank}_i(c) \leqslant r$, and so in particular $c_1, \dots, c_\ell$ must have rank r or better. Conversely, if $\text{rank}_i(c_\ell) \leqslant r$ then $c_1, \dots, c_\ell$ all have rank r or better, so there are at least ℓ candidates in W with rank r or better, and so $x_{i,\ell,r} = 1$.) Hence, the utility of voter i in committee W is

$$\begin{aligned}
\boldsymbol{\alpha}(s_{\text{rank}_i(c_1)}, \dots, s_{\text{rank}_i(c_k)}) &= \sum_{\ell \in [k]} \alpha_\ell \cdot s_{\text{rank}_i(c_\ell)} \\
&= \sum_{\ell \in [k]} \alpha_\ell \cdot \left(\textstyle\sum_{r=1}^{\text{rank}_i(c_\ell)} s'_r \right) \\
&= \sum_{\ell \in [k]} \sum_{r=1}^{\text{rank}_i(c_\ell)} \alpha_\ell \cdot s'_r \\
&= \sum_{\ell \in [k]} \sum_{r \in [m]} \alpha_\ell \cdot s'_r \cdot x_{i,\ell,r}.
\end{aligned}$$

Summing over all $i \in N$, we see that the objective value of solution S to the ILP equals $U(\mathbf{s}, \boldsymbol{\alpha}, W)$, the total utility of the committee W under the OWA-based rule. Thus, the ILP correctly encodes the winner determination problem.

Next suppose that the profile P is single-peaked. Consider the matrix M of coefficients in the constraints of the ILP. Let M' be the submatrix consisting only of the columns corresponding to the variables $(y_c)_{c \in A}$. Then M' has one row consisting of only 1s (corresponding to constraint (2)), and for each $i \in N$ and $r \in [m]$ a row whose 1-entries encode the set $\{c \in A : \text{rank}_i(c) \leqslant r\}$. Note that each of these sets is a top-initial segment of the preference order of i, and hence (see Section 9.2) an interval of the axis on which P is single-peaked. Therefore M' is a consecutive ones matrix (with the columns ordered according to the axis). Thus M' is totally unimodular by Proposition 9.12. Now, the matrix M is obtained from M' by appending columns corresponding to the variables $x_{i,\ell,r}$. Each of these variables occurs in only 1 constraint of type (3) with coefficient ± 1 (the sign depends on how we rearrange constraint (3) to bring all variables to one side). Thus, the column of the variable $x_{i,\ell,r}$ is a (negative) unit column, and so by Proposition 9.11, the matrix remains totally unimodular after appending it. Thus, M is totally unimodular. (Technically, we also need to include the constraints $0 \leqslant x_{i,\ell,r} \leqslant 1$ and $0 \leqslant y_c \leqslant 1$, but these are unit rows which can again be added without destroying total unimodularity.) Thus, by Theorem 9.10, the ILP can be solved in polynomial time.

The above argument for total unimodularity does not go through if P is SPOC but not single-peaked, because then the matrix M' only has the *circular* ones property. However, we can rearrange the ILP in such a way that we can show total unimodularity. This is a standard technique described in a useful survey by Dom [2009, Sec 4.1.4]. Before we begin, let us note the following general fact: suppose we are given a system of constraints

$$f(\mathbf{x}) = 0 \text{ and } g_j(\mathbf{x}) \leqslant 0 \text{ for } j = 1, \dots, J.$$

If in this system we replace one or more of the constraints $g_j(\mathbf{x}) \leqslant 0$ by $g_j(\mathbf{x}) - f(\mathbf{x}) \leqslant 0$, then the set of feasible solutions $\mathbf{x}$ to the system does not change.

Let P be a SPOC preference profile. Using the algorithms from Section 9.3, find a circle C such that P is single-peaked on C, and take some $\lhd \in C$ arbitrarily. For $i \in N$ and $r \in [m]$, write $T_{i,r} = \{c \in A : \text{rank}_i(c) \leqslant r\}$. Then $T_{i,r}$ is a top-initial segment of i's vote. Since P is single-peaked on C, $T_{i,r}$ is an interval of C. Thus, either $T_{i,r}$ or $A \setminus T_{i,r}$ is an interval of $\lhd$. Define the sets

$$\Gamma_1 = \{(i,r) : i \in N, r \in [m] \text{ such that } T_{i,r} \text{ is an interval of } \lhd\},$$
$$\Gamma_2 = \{(i,r) : i \in N, r \in [m] \text{ such that } A \setminus T_{i,r} \text{ is an interval of } \lhd\} \setminus \Gamma_1.$$

Then Γ_1 and Γ_2 form a partition of $N \times [m]$. Now consider the following integer linear program:

$$\text{maximise} \sum_{i \in N} \sum_{\ell \in [k]} \sum_{r \in [m]} \alpha_\ell \cdot s'_r \cdot x_{i,\ell,r} \qquad \text{(OWA-ILP-SPOC)}$$

$$\text{subject to} \sum_{c \in A} y_c - k = 0 \qquad \text{(2')}$$

$$\sum_{\ell \in [k]} x_{i,\ell,r} \leqslant \sum_{c \,:\, \text{rank}_i(c) \leqslant r} y_c \qquad \text{for } i \in N,\ r \in [m] \text{ with } (i,r) \in \Gamma_1 \qquad \text{(3')}$$

$$\sum_{\ell \in [k]} x_{i,\ell,r} \leqslant -\sum_{c \,:\, \text{rank}_i(c) > r} y_c + k \qquad \text{for } i \in N,\ r \in [m] \text{ with } (i,r) \in \Gamma_2 \qquad \text{(3'')}$$

$$x_{i,\ell,r} \in \{0,1\} \qquad \text{for } i \in N,\ \ell \in [k],\ r \in [m]$$

$$y_c \in \{0,1\} \qquad \text{for } c \in A$$

The program (OWA-ILP-SPOC) is very similar to the original program (OWA-ILP). Note that constraint (2') is the same as (2) after rearranging. The constraints (3') are a selection of the constraints (3). Finally, constraints (3") are obtained from constraints (3) after subtracting constraint (2'). Since (2') is an equality constraint, by the earlier mentioned general fact, we see that (OWA-ILP-SPOC) and (OWA-ILP) have the same set of feasible solutions. They also have the same objective function, and therefore (OWA-ILP-SPOC) also correctly encodes the problem of finding a winning committee.

Finally, we can prove that (OWA-ILP-SPOC) is totally unimodular, establishing the result that a winning committee can be found in polynomial time for SPOC profiles. Again take the constraint matrix M and consider the submatrix M' corresponding to the variables $(y_c)_{c \in A}$. If we rearrange the columns of M' according to $\lhd$, then each row of M' consists of either of an interval of $+1$s surrounded by 0s, or of an interval of -1s surrounded by 0s (the latter arising from constraints (3")). Combining Propositions 9.12 and 9.11, we see that M' is totally unimodular. As before, M is obtained from M' by adding columns with a single non-zero entry, so M is also totally unimodular. ☐

We obtain immediately the following two corollaries:

Corollary 9.14. *For SPOC profiles, Chamberlin–Courant can be computed in polynomial time.*

Corollary 9.15. *For SPOC profiles, PAV can be computed in polynomial time.*

An interesting question is whether the method of Theorem 9.13 can be further generalized. For example, does winner determination of OWA-based rules remain easy on single-peaked or SPOC profiles when we drop the unit cost assumption? This would be interesting for participatory budgeting applications. Since totally unimodular matrices can only include coefficients from $\{-1, 0, 1\}$, it seems unlikely that the packing constraint for non-unit costs can be implemented in this approach. However, we are not aware of a hardness result for this problem.

Fluschnik et al. [2019] study a rule that is an analogue of PAV where utilities need not be binary. They show that this generalization of PAV remains hard to compute for single-peaked utilities, thereby establishing a limit on the generalisability of the method of Theorem 9.13.

The ILP formulation (OWA-ILP) from the proof of Theorem 9.13 is of independent interest for computing OWA-based rules; for example, it seems to have proven useful in the empirical study of Faliszewski et al. [2018b]. Indeed, the "algorithm" that we propose for computing OWA-based rules (i.e., solving the program (OWA-ILP) using an ILP solver) is correct for general preferences, and comes with a polynomial-time guarantee in case the algorithm's input is single-peaked. This is in contrast to other winner determination algorithms that exploit preference structure: most such algorithms are specialised, and do not work at all if their input fails to be appropriately structured (this is the case, e.g., for the algorithms of Theorems 8.7, 8.9, and 9.9).

9.7. Discussion and Open Problems

Our results show that restricted preference domains that behave unfavourably in terms of axiomatic properties might still be very useful for algorithmic purposes. Indeed, our algorithms for Young's rule and OWA-based committee selection rules demonstrate that it is possible to move to a larger class than single-peaked preferences while maintaining polynomial-time runtime bounds. Thus, our findings can be seen as a challenge to established algorithmic results based on restricted preferences: to which degree can their application domain be extended without resorting to super-polynomial algorithms? One open problem of this type asks whether Dodgson's rule can be evaluated in polynomial time for SPOC profiles.

Our definition of SPOC is not the only sensible definition. One alternative definition that would fit into the generalised notion of single-peakedness introduced by Nehring and Puppe [2007] is based on *shortest paths*: it requires that for every voter i and for every alternative x, there is a *shortest* path between $\text{top}(i)$ and x along which i's preferences are decreasing. (Without the word "shortest" this is equivalent to SPOC.) The impact of this alternative definition is that every voter's least-preferred alternative needs to be *antipodal* to the voter's peak; this is strictly more restrictive than SPOC. It would be interesting to see whether this smaller domain allows for a wider range of positive results than SPOC.

Another direction for future work would be to extend the SPOC concept to two (and more) dimensions – preferences single-peaked on a *sphere* – but this may be difficult since little is known even about extensions of single-peakedness on a *line* to two or more dimensions (see Sui et al. 2013).

Thanks to Edith Elkind for discussions, and an anonymous AAAI-17 reviewer who pointed out mistakes in an early version.

Part IV.

Allocation of Indivisible Items with Connected Bundles

10. Maximin Fair Share and Envy-Freeness up to One Good

We study the allocation of indivisible private goods under the additional constraint that each bundle needs to be connected in an underlying item graph G. We allow agents to have arbitrary monotonic utility functions over bundles. In particular, we are interested in the existence of fair allocations, in the sense of satisfying the maximin share guarantee (MMS) or being envy-free up to one good (EF1). We show that MMS allocations are guaranteed to exist whenever G is a tree. When the items are arranged in a path, we show that EF1 allocations are guaranteed to exist, provided that either there are at most three agents, or there are any number of agents but they all have identical utility functions. Our existence proofs are based on classical arguments from the divisible cake-cutting setting, and involve discrete analogues of cut-and-choose and of Stromquist's moving-knife protocol. Further, the Su-Simmons technique based on Sperner's lemma can be used to show that on a path, an EF2 allocation exists for any number of agents. Except for the results using Sperner's lemma, all of our procedures can be implemented by efficient algorithms. Our positive results for paths imply the existence of connected EF1 or EF2 allocations whenever G is traceable, i.e., contains a Hamiltonian path. For the case of two agents, we completely characterise the class of graphs G that guarantee the existence of EF1 allocations as the class of graphs whose biconnected components are arranged in a path. This class is strictly larger than the class of traceable graphs; one can check in linear time whether a graph belongs to this class, and if so return an EF1 allocation.

10.1. Introduction

A famous literature considers the problem of *cake-cutting* [Brams and Taylor, 1996, Robertson and Webb, 1998, Procaccia, 2016]. There, a divisible heterogeneous resource (a *cake*, usually formalised as the interval $[0, 1]$) needs to be divided among n agents. Each agent has a valuation function over subsets of the cake, usually formalised as an atomless measure over $[0, 1]$. The aim is to partition the cake into n pieces, and allocate each piece to one agent, in a "fair" way. By fair, we will mean that the allocation is *envy-free*: no agent i thinks that another agent's piece is more valuable than i's own piece. A weaker fairness notion is *proportionality*, which requires that each agent i obtains a piece that i values at least $1/n$ as much as the entire cake $[0, 1]$.

When there are two agents, the classic procedure of cut-and-choose can produce an envy-free division: a knife is moved from left to right, until an agent shouts to indicate that the agent thinks the pieces to either side are equally valuable. The other agent then picks one of the pieces, leaving the remainder for the shouter. As is easy to see, the result is an envy-free allocation. For three or more agents, finding an envy-free division has turned out to be much trickier. An early result by Dubins and Spanier [1961] used Lyapunov's Theorem and measure-theoretic techniques to show, non-constructively, that an envy-free allocation always exists. However, as Stromquist [1980] memorably writes, "their result depends on a liberal definition of a 'piece' of cake, in which the possible pieces form an entire σ-algebra of subsets. A player who only hopes for a modest interval of cake may be presented instead with a countable union of crumbs." In many applications of resource allocation (such as land division, or the allocation of time slots), agents have little use for a severely disconnected piece of cake.

Stromquist [1980] himself offered a solution, and gave a new non-constructive argument (using topology) which proved that there always exists an envy-free division of the cake into *intervals*. Forest Simmons later observed that the proof could be simplified by using Sperner's lemma, and this technique was subsequently presented in a paper by Su [1999]. For the three-agent case, Stromquist [1980] also presented an appealing moving-knife procedure that more directly yields a connected envy-free allocation. For $n \geqslant 4$ agents, no explicit procedures are known to produce a connected envy-free allocation (i.e., an allocation where the cake is cut in exactly $n-1$ places). However, for $n = 4$, several moving-knife procedures exist that only need few cuts; for example, the Brams–Taylor–Zwicker [1997] procedure requires 11 cuts, and a protocol of Barbanel and Brams [2004] requires 5 cuts. (If we only aim for proportionality instead of envy-freeness, then explicit procedures are known for every $n \geqslant 2$.)

In many applications, the resources to be allocated are not infinitely divisible, and we face the problem of allocating *indivisible goods*. Most of the literature on indivisible goods has not assumed any kind of structure on the item space, in contrast to the rich structure of the interval $[0, 1]$ in cake-cutting. Thus, there has been little attention on minimising the number of "cuts" required in an allocation. However, when the items have a spatial or temporal structure, and disconnected bundles are infeasible or undesirable, this consideration is important.

In this chapter, we study the allocation of items that are arranged on a *path* or other structure, and impose the requirement that only *connected* subsets of items may be allocated to the agents. Formally, we assume that the items form the vertex set of a graph G. A bundle of items is connected if it induces a connected subgraph of G.

In general, it is impossible to achieve envy-freeness or proportionality with indivisible items: Consider two agents and a single desirable item; none of the possible partitions is envy-free or proportional. Instead, we can look for approximations. In an influential paper, Budish [2011] introduced weakened versions both of proportionality and of envy-freeness, adapted for the indivisible setting (without connectivity constraints). Both notions have turned out to be extremely productive, and have received much attention in the recent literature.

Budish's version of proportionality is the *maximin share guarantee*, abbreviated *MMS*. Rather than aiming to guarantee each agent a bundle of value at least $1/n$ of the entire set of available items, Budish defines a different guarantee. We imagine that each agent divides the items into n bundles, and will then choose last among the bundles, so that the agent receives the worst of the n bundles in the worst case. If the agent chooses the division into bundles so as to optimise the value of the worst bundle, we refer to the value of that bundle as the agent's MMS value. An MMS allocation gives every agent a bundle that is at least as valuable as the MMS value. For additive valuations in the setting without connectivity constraints, it was unclear whether an MMS allocation is guaranteed to exist. Bouveret and Lemaître [2016] could not find a counterexample with extensive sampling. Procaccia and Wang [2014] later found a counterexample, and a family of more compact examples was found by Kurokawa et al. [2018]. These examples are very intricate in the sense that the additive valuations have many decimal digits. Here, we consider the analogue of the MMS property for the connected setting. The definition is almost the same, except that when calculating the MMS values, we only allow partitions into connected bundles. Therefore, the MMS values are smaller than in the general setting, making the MMS guarantee easier to satisfy. On the other hand, we are restricted to satisfying the MMS guarantee with connected allocations only, which makes it harder. We are able to show that, when G is a path or a tree, then an MMS allocation always exists. The argument can be seen as an adaptation of the last diminisher procedure, which guarantees proportionality in cake cutting. Our argument works for any monotonic valuations, and they need not be additive. In contrast, we find a simple example where no MMS allocation exists when G is a cycle.

Budish also introduced the notion of envy-freeness *up to one good* (EF1). It requires that an agent's envy towards another bundle vanishes if we remove some item from the envied bundle. Caragiannis et al. [2016a] show that, in the setting without connectivity constraints and with additive valuations, the maximum Nash welfare solution satisfies EF1, as does a simple round-robin procedure. The well-known envy-graph algorithm from Lipton et al. [2004] also guarantees EF1. However, none of these procedures respects connectivity constraints.

We attempt to prove that connected EF1 allocations exist when G is a path. Like for MMS, we again take inspiration from the methods that worked for cake-cutting. Thus, we use successively more complicated tools to establish these existence results, and we prove that connected EF1 allocations exist when there are two or three agents. For two agents, there is a discrete analogue of cut-and-choose that satisfies EF1. In that procedure, a knife moves across the path, and an agent shouts when the knife reaches what we call a *lumpy tie*, that is when the bundles to either side of the knife have equal value *up to one item*. For three agents, we design an algorithm mirroring Stromquist's moving-knife procedure which guarantees EF1. For four or more agents, we show that Sperner's lemma can be used to prove that an EF2 allocation exists, via a technique inspired by the Simmons–Su approach, and an appropriately triangulated simplex of connected partitions of the path. We also show that if all agents have the same valuation function over bundles, then an egalitarian-welfare-optimal allocation, after suitably reallocating some items, is EF1.

These existence results require only that agents' valuations are monotonic (they need not be additive). Moreover, the fairness guarantee of our algorithms is slightly stronger than the standard notion of EF1: in the returned allocations, envy can be avoided by removing just an *outer* item – one whose removal leaves the envied bundle connected. Computationally speaking, all our existence results are immediately useful, since an example of an EF1 allocation can be found by iterating through all $O(m^n)$ connected allocations (this stands in contrast to cake-cutting where we cannot iterate through all possibilities). While we know of no faster algorithms to obtain an EF1 allocation in the cases where we appeal to Sperner's lemma, our other procedures can all be implemented efficiently.

In simultaneous and independent work, Oh et al. [2019] designed protocols to find EF1 allocations in the setting without connectivity constraints, aiming for low *query complexity*. They found that adapting cake-cutting protocols to the setting of indivisible items arranged on a path is an especially potent way to achieve low query complexity. This led them to also study a discrete version of the cut-and-choose protocol which achieves connected EF1 allocations for two agents, and they found an alternative proof that an EF1 allocation on a path always exists with identical valuations. They also present a discrete analogue of the Selfridge–Conway procedure which, for three agents with additive valuations, produces an allocation of a path into bundles that have a constant number of connected components. However, they do not study connected allocations on graphs that are not paths, and they do not consider the case of (non-identical) general valuations with more than two agents.

A recurring theme in our algorithms is the specific way that the moving knives from cake-cutting are rendered in the discrete setting. While one might expect knives to be placed over the edges of the path, and 'move' from edge to edge, we find that this movement is too 'fast' to ensure EF1 (see also footnote 4). Instead, our knives alternate between hovering over edges and items. When a knife hovers over an item, we imagine the knife's blade to be 'thick': the knife *covers* the item, and agents then pretend that the covered item does not exist. These intermediate steps are useful, since they can tell us that envy will vanish if we hide an item from a bundle.

What about graphs G other than paths? Our positive results for paths immediately generalise to traceable graphs (those that contain a Hamiltonian path), since we can run the algorithms pretending that the graph only consists of the Hamiltonian path. For the two-agent case, we completely characterise the class of graphs that guarantee the existence of EF1 allocations: Our discrete cut-and-choose protocol can be shown to work on all graphs G that admit a *bipolar numbering*, which exists if and only if the biconnected components (blocks) of G can be arranged in a path. By constructing counterexamples, we prove that no graph failing this condition (for example, a star) guarantees EF1, even for identical, additive, binary valuations. For the case of three or more agents, it is a challenging open problem to characterise the class of graphs guaranteeing EF1 (or even to find an infinite class of non-traceable graphs that guarantees EF1).

10.2. Preliminaries

For each natural number $s \in \mathbb{N}$, write $[s] = \{1, 2, \dots, s\}$.

Let $N = [n]$ be a finite set of *agents* and $G = (V, E)$ be an undirected finite graph. We refer to the vertices in V as *goods* or *items*. A subset I of V is *connected* if it induces a connected subgraph $G[I]$ of G. We write $\mathcal{C}(V) \subseteq 2^V$ for the set of connected subsets of V. We call a set $I \in \mathcal{C}(V)$ a (connected) *bundle*. Each agent $i \in N$ has a *valuation function* $u_i : \mathcal{C}(V) \to \mathbb{R}$ over connected bundles, which we will always assume to be *monotonic*, that is, $X \subseteq Y$ implies $u_i(X) \leqslant u_i(Y)$. We also assume that $u_i(\varnothing) = 0$ for each $i \in N$. Monotonicity implies that items are *goods*; we do not consider bads (or chores) in this chapter. We say that an agent $i \in N$ *weakly prefers* bundle X to bundle Y if $u_i(X) \geqslant u_i(Y)$; also, agent $i \in N$ *strictly prefers* bundle X to bundle Y if $u_i(X) > u_i(Y)$.[1] A (connected) *allocation* is a function $A : N \to \mathcal{C}(V)$ assigning to each agent $i \in N$ a connected bundle $A(i) \in \mathcal{C}(V)$ such that each item occurs in exactly one agent's bundle, i.e., $\bigcup_{i \in N} A(i) = V$ and $A(i) \cap A(j) = \varnothing$ whenever $i \neq j$.

We say that the agents have *identical valuations* when for all $i, j \in N$ and every bundle $I \in \mathcal{C}(V)$, we have $u_i(I) = u_j(I)$. A valuation function u_i is *additive* if $u_i(I) = \sum_{v \in I} u_i(\{v\})$ for each bundle $I \in \mathcal{C}(V)$. Many examples in this chapter will use identical additive valuations, and will take G to be a path. In this case, we use a succinct notation to specify these examples; the meaning of this notation should be clear. For example, we write "2–1–3–1" to denote an instance with four items v_1, v_2, v_3, v_4 arranged on a path, and where $u_i(\{v_1\}) = 2, \dots, u_i(\{v_4\}) = 1$ for each i. For such an instance, an allocation will be written as a tuple, e.g., (2, 1–3–1) denoting an allocation allocating bundles $\{v_1\}$ and $\{v_2, v_3, v_4\}$, noting that with identical valuations it does not usually matter which agent receives which bundle.

Let Π_n denotes the set of all partitions of V into n connected bundles. The *maximin share* of an agent $i \in N$ is

$$\text{MMS}_i := \max_{(P^1, P^2, \dots, P^n) \in \Pi_n} \min_{j \in [n]} u_i(P^j).$$

This is the utility achieved by agent i if i divides the items into n connected pieces so as to maximise the value of the worst piece. An allocation A is an *MMS allocation* if $u_i(A(i)) \geqslant \text{MMS}_i$ for each agent $i \in N$.

An allocation A is *envy-free* if $u_i(A(i)) \geqslant u_i(A(j))$ for every pair $i, j \in N$ of agents, that is, if every agent thinks that their bundle is a best bundle in the allocation. It is well-known that an envy-free allocation may not exist (consider two agents and one good). The main fairness notion that we study is a version of *envy-freeness up to one good* (EF1), a relaxation of envy-freeness introduced by Budish [2011] and popularised by Caragiannis et al. [2016a], adapted to the model with connectivity constraints. This property states that an agent i will not envy another agent j after we remove some single item from j's bundle. Since we only allow connected bundles in

[1] Our arguments only operate based on agents' ordinal preferences over bundles, and the (cardinal) valuation functions are only used for notational convenience. One exception, perhaps, is in Algorithm 9 where we calculate a leximin allocation, but the algorithm can be applied after choosing an arbitrary utility function consistent with the ordinal preferences.

our set-up, we may only remove an item from $A(j)$ if removal of this item leaves the bundle connected. Thus, our formal definition of EF1 is as follows.

Definition 10.1 (EF1: envy-freeness up to one *outer* good)**.** An allocation A satisfies *EF1* if for any pair $i, j \in N$ of agents, either $A(j) = \varnothing$ or there is a good $v \in A(j)$ such that $A(j) \setminus \{v\}$ is connected and $u_i(A(i)) \geqslant u_i(A(j) \setminus \{v\})$.

In the instance 2–1–3–1 for two agents, the allocation (2–1, 3–1) is EF1, since the left agent's envy can be eliminated by removing the item of value 3 from the right-hand bundle. However, the allocation (2, 1–3–1) fails to be EF1 according to our definition, since eliminating either outer good of the right bundle does not prevent envy.[2]

Definition 10.2. A graph G *guarantees EF1* (for a specific number of agents n) if for all possible monotonic valuations for n agents, there exists some connected allocation that is EF1. A graph G guarantees EF1 for n agents and a restricted class of valuations if for all allowed valuations, a connected EF1 allocation exists.

For reasoning about EF1 allocations, let us introduce a few shorthands. Given an allocation A we will say that $i \in N$ *does not envy* $j \in N$ *up to* v if $u_i(A(i)) \geqslant u_i(A(j) \setminus \{v\})$. The *up-to-one valuation* $u_i^- : \mathcal{C}(V) \to \mathbb{R}_{\geqslant 0}$ of agent $i \in N$ is defined, for every $I \in \mathcal{C}(V)$, as

$$u_i^-(I) := \begin{cases} 0 & \text{if } I = \varnothing, \\ \min\{u_i(I \setminus \{v\}) : v \in I \text{ such that } I \setminus \{v\} \text{ is connected}\} & \text{if } I \neq \varnothing. \end{cases} \tag{10.1}$$

Thus, an allocation A satisfies EF1 if and only if $u_i(A(i)) \geqslant u_i^-(A(j))$ for any pair $i, j \in N$ of agents.

Given an ordered sequence of the vertices $P = (v_1, v_2, \dots, v_m)$, and $j, k \in [m]$ with $j \leqslant k$, we denote the subsequence from v_j to v_k by $P(v_j, v_k)$, i.e.,

$$P(v_j, v_k) = (v_j, v_{j+1}, \dots, v_{k-1}, v_k).$$

With a little abuse of notation, we often identify a subsequence $P(v_j, v_k)$ with the bundle of the corresponding vertices. Let us define $L(v_j) = P(v_1, v_{j-1})$ as the subsequence of vertices strictly left of v_j and $R(v_j) = P(v_{j+1}, v_m)$ as the subsequence of vertices strictly right of v_j. When G is a path, in the following we always implicitly assume that its vertices $v_1, v_2, \dots, v_m$ are numbered from left to right according to the order they appear along the path, so that the set of the edges of G is $\{\{v_j, v_{j+1}\} : 1 \leqslant j < m\}$. Each connected bundle in the path clearly corresponds to a subpath or subsequence of the vertices. A *Hamiltonian path* of a graph G is a path that visits all the vertices of the graph exactly once. A graph is *traceable* if it contains a Hamiltonian path.

10.3. MMS Existence

Suppose G is a path, and we are given a profile of agent valuations. Our aim is to find an MMS allocation. To this end, for each agent $i \in N$, let $(P_i^1, \dots, P_i^n) \in \Pi_n$ be a partition of the items such that $\min_{j \in [n]} P_i^j = \text{MMS}_i$. Now, in the allocation A we are going to construct, some agent $i \in N$ is going to obtain the left-most bundle, and in order for A to be an MMS allocation, we

[2]This example shows that our definition is strictly stronger than the standard definition of EF1 without connectivity constraints. In the instance 2–1–3–1, considered without connectivity constraints, the allocation (2, 1–3–1) does satisfy EF1 since in the standard setting we are allowed to remove the middle item (with value 3) of the right bundle.

need that $P_i^1 \subseteq A(i)$. Since we can reallocate extra items, while looking for an MMS allocation, we may search for allocations where $A(i) = P_i^1$ for some agent i.

Now, how can we decide which agent should receive the left-most item? An intuitive choice would be the agent whose MMS guarantee can most easily be satisfied. Thus, we would look for an agent $j \in N$ such that $|P_j^1|$ is minimal, and allocate P_j^1 to that agent. Now we can recurse on the remaining items and the remaining agents.

The reduced instance we have obtained in this way has a convenient property: If we calculate the MMS values of the remaining $n-1$ agents on the remaining items, these MMS values can only have increased. This is because the bundles in the partitions $(P_i^2, \dots, P_i^n)$ for $i \in N \setminus \{j\}$ only contain items that still remain, having chosen $|P_j^1|$ minimal. Thus, by induction (the case $n = 1$ being trivial), if we can obtain an MMS allocation of the remaining instance and combine it with the allocation of P_j^1 to agent j, we have obtained an MMS allocation for our original instance. Thus, we have proved that MMS allocations always exist when G is a path.

Theorem 10.3. *When G is a path, an MMS allocation exists.*

The procedure sketched above can be implemented in polynomial time, because we can use a straightforward dynamic program to compute MMS values when G is a path. The resulting procedure can be seen as a discrete analogue of the last-diminisher procedure, which can be used in cake-cutting to obtain a proportional division. This procedure was discovered by Banach and Knaster [Steinhaus, 1948, Brams and Taylor, 1996].

One can prove that Theorem 10.3 continues to hold when G is a tree or a forest [Bouveret, Cechlárová, Elkind, Igarashi, and Peters, 2017]. In the argument, we root the tree, and then look for an agent whose MMS guarantee can be satisfied by a subtree of minimum height. The correctness proof is similar, and the procedure can be implemented in polynomial time. On the other hand, an MMS allocation is not guaranteed to exist when G is a cycle, and there is a simple counterexample [Bouveret et al., 2017]. Lonc and Truszczynski [2018] showed that there exist approximate MMS allocations when G is a cycle.

10.4. EF1 Existence for Two Agents

In cake-cutting for two agents, the standard way of obtaining an envy-free allocation is the cut-and-choose protocol: Alice divides the cake into two equally-valued pieces, and Bob selects the piece he prefers; the other piece goes to Alice. The same strategy almost works in the indivisible case when items form a path; the problem is that Alice might not be able to divide the items into two exactly-equal pieces. Instead, we ask Alice to divide the items into pieces that are equally valued "up to one good". The formal version is as follows. For a sequence of vertices $P = (v_1, v_2, \dots, v_m)$ and an agent i, we say that v_j is the *lumpy tie* over P for agent i if j is the smallest index such that

$$u_i(L(v_j) \cup \{v_j\}) \geqslant u_i(R(v_j)) \quad \text{and} \quad u_i(R(v_j) \cup \{v_j\}) \geqslant u_i(L(v_j)). \tag{10.2}$$

For example, when i has additive valuations 1–3–2–1–3–1, then the third item (of value 2) is the lumpy tie for i, since $1+3+2 \geqslant 1+3+1$ and $2+1+3+1 \geqslant 1+3$. The lumpy tie always exists: taking j to be the smallest index such that $u_i(L(v_j) \cup \{v_j\}) \geqslant u_i(R(v_j))$ (which exists as the inequality holds for $j = m$ by monotonicity), the first part of (10.2) holds. If $j = 1$, the second part of (10.2) is immediate by monotonicity. If $j > 1$, then since j is minimal, we have $u_i(L(v_j)) = u_i(L(v_{j-1}) \cup \{v_{j-1}\}) < u_i(R(v_{j-1})) = u_i(R(v_j) \cup \{v_j\})$ which is the second part of (10.2).

Using the concept of the lumpy tie, our discrete version of the cut-and-choose protocol is specified as follows.

Discrete cut-and-choose protocol for $n = 2$ agents over a sequence $P = (v_1, v_2, \dots, v_m)$:
Step 1. Alice selects her lumpy tie v_j over $(v_1, v_2, \dots, v_m)$.
Step 2. Bob chooses a weakly preferred bundle among $L(v_j)$ and $R(v_j)$.
Step 3. Alice receives the bundle of all the remaining vertices, including v_j.

Intuitively, the protocol allows Alice to select an item v_j that she will receive for sure, with the advice that the two pieces to either side of v_j should have almost equal value to her. Then, Bob is allowed to choose which side of v_j he wishes to receive. In our example with valuations 1–3–2–1–3–1, Alice selects the lumpy tie of value 2, then Bob chooses the bundle 1–3–1 to the right and receives it, and Alice receives the bundle 1–3–2. The result is EF1. This is true in general, and also if valuations are not identical.

Proposition 10.4. *When G is a path and there are $n = 2$ agents, the discrete cut-and-choose protocol yields an EF1 allocation.*

Proof. Clearly, the protocol returns a connected allocation. The returned allocation satisfies EF1: Bob does not envy Alice up to item v_j, since Bob receives his preferred bundle among $L(v_j)$ and $R(v_j)$. Also, by (10.2), Alice does not envy Bob, since Alice either receives the bundle $L(v_j) \cup \{v_j\}$ which she weakly prefers to Bob's bundle $R(v_j)$, or she receives the bundle $R(v_j) \cup \{v_j\}$, which she weakly prefers to Bob's bundle $L(v_j)$. □

Proposition 10.4 implies that an EF1 allocation always exists on a path. It immediately follows that an EF1 allocation exists for every traceable graph G: simply use the discrete cut-and-choose protocol on a Hamiltonian path of G; the resulting allocation must be connected in G. In fact, the discrete cut-and-choose protocol works on a broader class of graphs: We only need to require that the vertices of the graph can be numbered in a way that the allocation resulting from the discrete cut-and-choose protocol is guaranteed to be connected. Since the protocol always partitions the items into an initial and a terminal segment of the sequence, such a numbering needs to satisfy the following property.

Definition 10.5. A *bipolar numbering* of a graph G is an ordered sequence $(v_1, v_2, \dots, v_m)$ of its vertices such that for each $j \in [n]$, the sets $L(v_j) \cup \{v_j\}$ and $R(v_j) \cup \{v_j\}$ are connected in G.

An equivalent definition (which is the standard one) says that a numbering is bipolar if for every $j \in [n]$, the vertex v_j has a neighbour that appears earlier in the sequence, and a neighbour that appears later in the sequence. Bipolar numberings are used in algorithms for testing planarity and for graph drawing. Every Hamiltonian path induces a bipolar numbering, but there are also non-traceable graphs that admit a bipolar numbering, see Figure 10.1 for examples.

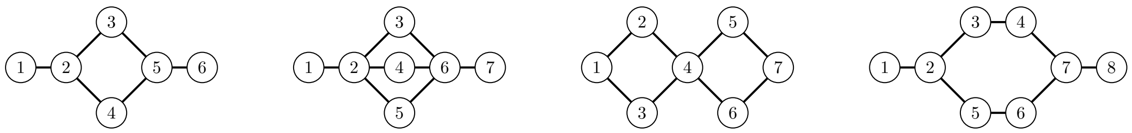


Figure 10.1.: Non-traceable graphs with bipolar numberings.

Proposition 10.6. *When there are $n = 2$ agents, then the discrete cut-and-choose protocol run on a bipolar numbering of G yields an EF1 allocation.*

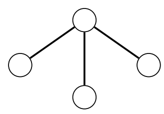
Proof. The discrete cut-and-choose protocol always returns an allocation whose bundles are either initial or terminal segments of the ordered sequence $(v_1, v_2, \dots, v_m)$. By definition of a bipolar numbering, such an allocation is connected. The argument of Proposition 10.4 shows that the allocation satisfies EF1. □

It is clear that the discrete cut-and-choose protocol cannot be extended to graphs other than those admitting a bipolar numbering. However, it could be that a different protocol is able to produce EF1 allocations on other graphs. In the remainder of this section, we prove that this is not the case: for $n = 2$ agents, a connected graph G guarantees the existence of an EF1 allocation if and only if it admits a bipolar numbering. This completely characterises the class of graphs that guarantee EF1 existence in the two-agent case.[3]

For a different number of agents, the class of graphs guaranteeing an EF1 allocation will be different. In particular, the star with three leaves does not guarantee an EF1 allocation for two agents (as it does not have a bipolar numbering, see below), but one can check that this star does guarantee an EF1 allocation for three or more agents.

10.4.1. Characterisation of graphs guaranteeing EF1 for two agents

Based on a known characterisation of graphs admitting a bipolar numbering, we characterise this class in terms of forbidden substructures. We then show that these forbidden structures are also forbidden for EF1: if a graph contains such a structure, we can exhibit an additive valuation profile for which no EF1 allocation exists.



As a simple example, consider the star with three leaves, which is the smallest connected graph that does not have a bipolar numbering. Suppose there are two agents with identical additive valuations that value each item at 1. Any connected allocation must allocate three items to one agent, and a single item to the other agent. No such allocation is EF1, since the agent with the singleton bundle envies the other agent, even up to one good. This star is an example of what we call a *trident*, and forms a forbidden substructure. Informally, the forbidden substructures take one of two forms. We will prove that a graph G fails to admit a bipolar numbering, and fails to guarantee EF1 for two agents, iff either

(a) there exists a vertex s whose removal from G leaves three or more connected components, or

(b) there are subgraphs C, P_1, P_2, P_3 of G such that (i) G is the union of these subgraphs, (ii) the subgraphs P_1, P_2, P_3 are vertex-disjoint, (iii) C has exactly one vertex is common with P_i, $i = 1, 2, 3$, and (iv) no edge joins a vertex from one of these four subgraphs to a different one.

To reason about these structures, it is useful to consider the *block decomposition* of a graph, which will show that graphs that admit a bipolar numbering have an underlying path-like structure. A *decomposition* of a graph $G = (V, E)$ is a family $\{F_1, F_2, \dots, F_t\}$ of edge-disjoint subgraphs of G such that $\bigcup_{i=1}^{t} E(F_i) = E$ where $E(F_i)$ is the set of edges of a subgraph F_i.

Definition 10.7. A vertex is called a *cut* vertex of a graph G if removing it increases the number of connected components of G. A graph G is *biconnected* if G does not have a cut vertex. A *block* of G is a maximal biconnected subgraph of G.

Equivalently, a block of a graph G can be defined as a maximal subgraph of G where each pair of vertices lie on a common cycle [Bondy and Murty, 2008]. Given a connected graph G, we define a bipartite graph $B(G)$ with bipartition $(\mathcal{B}, S)$, where $\mathcal{B}$ is the set of blocks of G and S is the set of cut vertices of a graph G; a block B and a cut vertex v are adjacent in $B(G)$ if and only if B includes v. Since every cycle of a graph is included in some block, the graph $B(G)$ is known to be a tree:

[3] Note that no non-trivial disconnected graph guarantees EF1 for two agents: If G is disconnected, take a connected component C with at least two vertices. Let both agents have additive valuations that value each item in C at 1, and value items outside of C at 0. Then, in a connected allocation, all items in C must go to a single agent, since the other agent needs to receive items from another connected component. This induces envy in the other agent that is not bounded by one good.

Lemma 10.8 (Bondy and Murty, 2008)**.** *Let G be a connected graph. Then*

- *any two blocks of G have at most one cut vertex in common;*
- *the set of blocks forms a decomposition of G; and*
- *the graph $B(G)$ is a tree.*

Thus, for a connected graph G, we call $B(G)$ the *block tree* of G. It turns out that G admits a bipolar numbering if and only if $B(G)$ is a path. For example, the graphs shown in Figure 10.1 all have their blocks arranged in a path (so that $B(G)$ is a path), as shown in Figure 10.2.

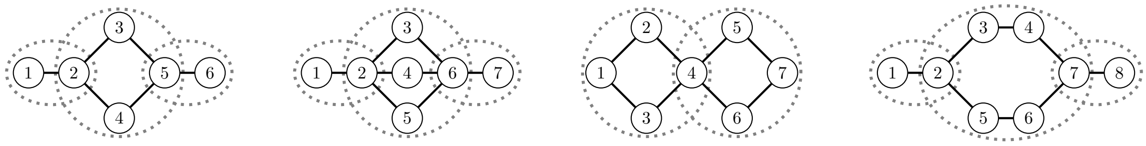


Figure 10.2.: Block decompositions of graphs in Figure 10.1.

Lemma 10.9. *A graph G admits a bipolar numbering if its block tree $B(G)$ is a path.*

Proof. Lempel et al. [1967] proved that G admits a bipolar numbering if there exist $s, t \in V$ such that adding an edge $\{s, t\}$ to G makes the graph biconnected. Suppose $B(G)$ is a path, and let B_1 and B_2 be the leaf blocks at the ends of the path $B(G)$. Take any $s \in B_1$ and $t \in B_2$. If we add the edge $\{s, t\}$ to G, then the graph becomes biconnected. Hence, G admits a bipolar numbering. □

Even and Tarjan [1976] provided a linear-time algorithm based on depth-first search to construct a bipolar numbering for any biconnected graph [see also Tarjan, 1986]. Using an algorithm by Hopcroft and Tarjan [1973] (also based on depth-first search), we can calculate the block tree $B(G)$ of a given graph in linear time. Thus, in linear time, we can compute a bipolar numbering of a graph whose block tree is a path, or establish that no bipolar numbering exists. Clearly, given a bipolar numbering, the discrete cut-and-choose protocol can also be run in linear time.

Next, we show that if $B(G)$ is not a path, then G cannot guarantee EF1. The proof constructs explicit counter-examples, which have a very simple structure. We say that additive valuations u_i are *binary* if $u_i(\{v\}) \in \{0, 1\}$ for every $v \in V$.

Lemma 10.10. *Let G be a connected graph whose block tree $B(G)$ is* not *a path. Then there exist identical, additive, binary valuations over G for two agents such that no connected allocation is EF1.*

Proof. If $B(G)$ is not a path, then $B(G)$ has a *trident*, i.e., a vertex with at least three neighbours, and thus either

(a) there is a cut vertex s adjacent to three blocks B_1, B_2, and B_3; or

(b) there is a block B adjacent to three different cut vertices s_1, s_2, and s_3.

See Figure 10.3 for an illustration. In either case, we will construct identical additive valuations that do not admit a connected EF1 allocation.

In case (a), for each $i = 1, 2, 3$, choose a vertex v_i from $B_i \setminus \{s\}$. Note that we can choose such $v_i \neq s$ due to the maximality of B_i. The two agents have utility 1 for s, v_1, v_2, and v_3, and 0 for the remaining vertices. Now take any connected allocation (I_1, I_2). One of the bundles, say I_1, includes the cut vertex s. Then I_2 can contain at most one of the vertices v_1, v_2, v_3, since I_2 is

(a) A cut vertex adjacent to three blocks (b) A block adjacent to three cut vertices

Figure 10.3.: Tridents.

connected and does not contain s yet any path between distinct v_i and v_j goes trough s. Hence $u_i(I_2) \leqslant 1$. Now, the bundle I_1 contains s and at least two of v_1, v_2, v_3, so $u_i(I_1) \geqslant 3$. Thus, the allocation is not EF1.

In case (b), for each $i = 1, 2, 3$, let B_i be the block sharing the cut vertex s_i with B. Note that each pair of the blocks B_1, B_2, B_3 does not share any cut vertex because $B(G)$ forms a tree. Choose a vertex v_i from $B_i \setminus \{s_i\}$ for each $i = 1, 2, 3$. Again, one can choose $v_i \neq s_i$ due to the maximality of B_i. The two agents have utility 1 for s_1, s_2, s_3, v_1, v_2, and v_3, and 0 for the remaining vertices. Now take any connected allocation (I_1, I_2). One of the bundles, say I_1, contains at least two cut vertices s_i and the other contains at most one cut vertex s_i. Say that $s_1, s_2 \in I_1$. Now, $G \setminus \{s_1, s_2\}$ has three connected components, and since I_2 is connected, it must be contained in of these components. But each component contains at most two vertices with utility 1, so $u_i(I_2) \leqslant 2$. Since there are six vertices with utility 1 in total, $u_i(I_1) \geqslant 4$. Thus, the allocation is not EF1. □

Combining these results, we obtain the promised characterisation.

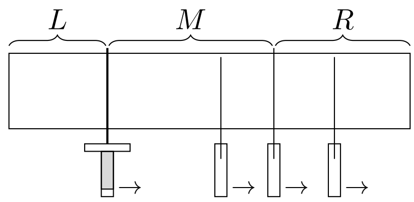
Theorem 10.11. *The following conditions are equivalent for every connected graph G:*

1. *G admits a bipolar numbering.*
2. *G guarantees EF1 for two agents.*
3. *G guarantees EF1 for two agents with identical, additive, binary valuations.*
4. *The block tree $B(G)$ is a path.*

Proof. The implication (1) ⇒ (2) follows from Proposition 10.6 which shows that the discrete cut-and-choose protocol yields a connected EF1 allocation when run on a bipolar numbering. The implication (2) ⇒ (3) is immediate. The implication (3) ⇒ (4) follows from Lemma 10.10 which proves the contrapositive. Finally, (4) ⇒ (1) follows from Lemma 10.9. □

The equivalence (2) ⇔ (3) is noteworthy and perhaps surprising: It is often easier to guarantee fairness when agents' valuations are identical, yet in terms of the graphs that guarantee EF1 for two agents, there is no difference between identical and non-identical valuations. Intriguingly, even for more than two agents, we do not know of a graph which guarantees EF1 for identical valuations, but fails it for non-identical valuations.

10.5. EF1 Existence for Three Agents: A Moving-Knife Protocol



We will now consider the case of three agents. Stromquist [1980] designed a protocol that results in an envy-free contiguous allocation of a divisible cake. In outline, the protocol works as follows. A referee holds a sword over the cake. Each of the three agents holds their own knife over the portion of the cake to the right of the sword. Each agent positions their knife so that the portion to the right of the sword is divided into two pieces they judge to have the same value. Now, initially, the sword is at the left end of the cake and then starts moving at constant speed from left to right, while the agents continuously move their knives to keep dividing the right-hand portion into equally-valued pieces. At some point (when the left-most piece becomes valuable enough), one of the agents shouts "cut", and the cake will be cut twice: once by the sword, and once by the middle one of the three knives. Agents shout "cut" as soon as the left piece is a highest-valued piece among the three. The agent who shouts receives the left piece. The remaining agents each receive a piece containing their knife. One can check that the resulting allocation is envy-free, since the agent receiving the left piece prefers it to the other pieces, and the other agents who are not shouting receive at least half the value of the part of the cake to the right of the sword.

Let G be a path, $P = (v_1, v_2, \ldots, v_m)$. There are several difficulties in translating Stromquist's continuous procedure to the discrete setting for G. First, agents need to divide the piece to the right of the sword in half, and this might not be possible exactly given indivisibilities; but this can be handled using our concept of lumpy ties from Section 10.4. Next, when the sword moves one item to the right, the lumpy ties of the agents may need to jump several items to the right, for example because the new member of the left-most bundle is very valuable. To ensure EF1, we will need to smoothen these jumps, so that the middle piece grows one item at a time. Also, it will be helpful to have the sword move in half-steps: it alternates between being placed between items (so it cuts the edge between the items), and being placed over an item, in which case the sword covers the item and agents ignore that item. Finally, while the sword covers an item, we will only terminate if at least *two* agents shout to indicate that they prefer the left-most piece; this will ensure that there is an agent who is flexible about which of the bundles they are assigned. The algorithm moves in steps, and alternates between moving the sword, and updating the lumpy ties.

In our formal description of the algorithm, we do not use the concepts of swords and knives. Instead, the algorithm maintains three bundles L, M, and R that can be seen as resulting from a certain configuration of these cutting implements. We also need a few auxiliary definitions. Recall that for a subsequence of vertices $P(v_s, v_r) = (v_s, v_{s+1}, \ldots, v_r)$ and an agent i, we say that v_j ($s \leqslant j \leqslant r$) is the *lumpy tie* over $P(v_s, v_r)$ for i if j is the smallest index such that

$$u_i(L(v_j) \cup \{v_j\}) \geqslant u_i(R(v_j)) \quad \text{and} \quad u_i(R(v_j) \cup \{v_j\}) \geqslant u_i(L(v_j)). \tag{10.3}$$

Here, the definitions of $L(v_j)$ and $R(v_j)$ apply to the subsequence $P(v_s, v_r)$. The lumpy tie always exists by the discussion after equation (10.2). Each of the three agents has a lumpy tie over $P(v_s, v_r)$; a key concept for us is the *median lumpy tie* which is the median of the lumpy ties of the three agents, where the median is taken with respect to the ordering of $P(v_s, v_r)$. We say that $i \in N$ is a *left agent* (respectively, a *middle agent* or a *right agent*) over $P(v_s, v_r)$ if the lumpy tie for i appears strictly before (respectively, is equal to, or appears strictly after) the median lumpy tie. Note that by definition of median, there is at most one left agent, at most one right agent, and at least one middle agent.

Suppose that the median lumpy tie over the subsequence $P(v_s, v_r)$ is v_j, and let i be an agent.

Then using the definitions of lumpy tie and left/right agents, we find that

$$\begin{array}{ll} u_i(L(v_j)) \geqslant u_i(R(v_j) \cup \{v_j\}) & \text{if } i \text{ is a left agent, and} \\ u_i(R(v_j)) \geqslant u_i(L(v_j) \cup \{v_j\}) & \text{if } i \text{ is a right agent.} \end{array} \tag{10.4}$$

Given the median lumpy tie v_j over the subsequence $P(v_s, v_r)$, and a two-agent set $S = \{i, k\} \subseteq N$, we define $\mathsf{Lumpy}(S, v_j, P(v_s, v_r))$ to be the allocation of the items in $P(v_s, v_r)$ to S such that

- if i is a left agent and k is a right agent, then i receives $L(v_j)$ and k receives $R(v_j) \cup \{v_j\}$;
- if i is a middle agent, then agent k receives k's preferred bundle among $L(v_j)$ and $R(v_j)$, and agent i receives the other bundle along with v_j.

Using (10.3) and (10.4), we see that $\mathsf{Lumpy}(S, v_j, P(v_s, v_r))$ is an EF1 allocation:

Lemma 10.12 (Median Lumpy Ties Lemma)**.** *Suppose that $S = \{i, k\} \subseteq N$ and v_j is the median lumpy tie over the subsequence $P(v_s, v_r)$. Then $\mathsf{Lumpy}(S, v_j, P(v_s, v_r))$ is an EF1 allocation of the items in $P(v_s, v_r)$ to S. Furthermore, each agent in S receives a bundle weakly better than the two bundles $L(v_j)$ and $R(v_j)$.*

We now present the algorithm. The algorithm alternately moves a left pointer ℓ (in Steps 2 and 3) and a right pointer r (in Step 4). It also maintains bundles L, M, and R during the execution of the algorithm.

Theorem 10.13. *The moving-knife protocol finds an EF1 allocation for three agents and runs in $O(|V|)$ time, when G is traceable.*

Proof. The algorithm is well-defined – there is one place where this is not immediate: If two agents shout in Step 3, the algorithm description claims that there is a shouter who is a middle agent over the subsequence $P(v_{\ell+1}, v_m)$. Suppose for the moment that there is a shouter i who is a *right* agent. Due to (10.4), we have $u_i(R) \geqslant u_i(\{v_{\ell+1}\} \cup M \cup \{v_r\})$. Since i is a shouter, we have $u_i(L) \geqslant u_i(R)$, so $u_i(L) \geqslant u_i(\{v_{\ell+1}\} \cup M \cup \{v_r\})$. But i did not shout in the previous Step 2 (when no-one shouted), so either $u_i(R) > u_i(L)$ or $u_i(\{v_{\ell+1}\} \cup M) > u_i(L)$, and either case is a contradiction. Hence neither of the at least two shouters of Step 3 is a right agent, so at least one shouter is a middle agent, since there is at most one left agent.

The algorithm terminates and returns an allocation, since the bundle L grows throughout the algorithm until eventually, at least two agents will think that L is the best bundle and thus will shout and thereby terminate the algorithm. We will now consider every possible way that the algorithm could have terminated, and show that the resulting allocation is EF1. To follow this proof, it is helpful to look at the figures in the description of the procedure.

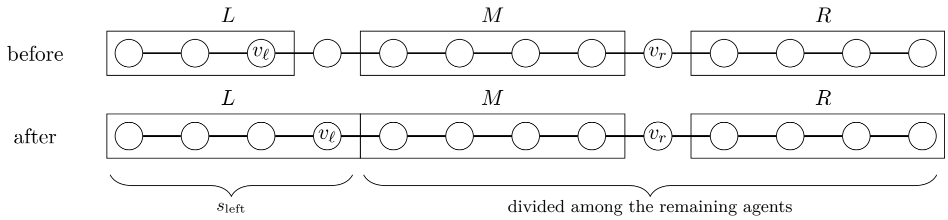
Step 2.

- Agent s_{left} receives L and does not envy the other agents (up to good v_r) since s_{left} is a shouter.
- An agent i who is not a shouter does not envy s_{left} because i prefers either M or R to L, and hence by Lemma 10.12 receives a bundle preferred to L.
  Agent i also does not envy the other agent $j \neq s_{\text{left}}$ up to one good by Lemma 10.12.
- An agent $i \neq s_{\text{left}}$ who is a shouter does not envy s_{left} up to one good: If this is the first time Step 2 was performed, then $L = \{v_1\}$, so i does not envy s_{left} up to v_1. Otherwise, the last step was an iteration of Step 4(c), where by definition of Step 4(c) no-one shouted. Since i did not shout during Step 4(c), and Step 2 did not change the bundles M and R, then i strictly prefers either M or R to the left bundle $L \setminus \{v_\ell\}$ of Step 4(c). By Lemma 10.12, agent i gets a bundle at least as good as M or R. Thus, i does not envy s_{left} up to v_ℓ.
  Also by Lemma 10.12, agent i does not envy the other agent $j \neq s_{\text{left}}$ up to one good.

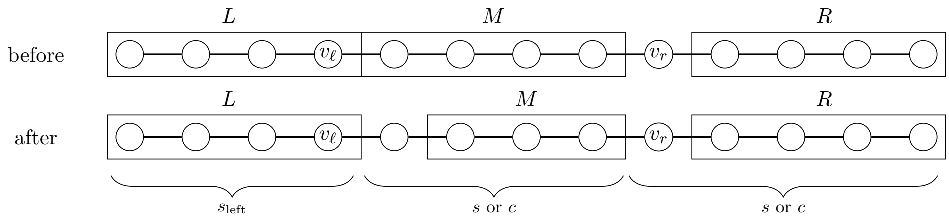
Discrete moving-knife protocol for $n = 3$ agents over a sequence $P = (v_1, v_2, \dots, v_m)$:
An agent $i \in N$ is a *shouter* if L is best among L, M, R, so that $u_i(L) \geqslant u_i(M)$ and $u_i(L) \geqslant u_i(R)$.

Step 1. Initialise $\ell = 0$ and set r so that v_r is the median lumpy tie over the subsequence $P(v_2, v_m)$. Initialise $L = \varnothing$, $M = \{v_2, v_3, \dots, v_{r-1}\}$, and $R = \{v_{r+1}, v_{r+2}, \dots, v_m\}$.

Step 2. Add an additional item to L, i.e., set $\ell = \ell + 1$ and $L = \{v_1, v_2, \dots, v_\ell\}$.
If no agent shouts, go to Step 3. If some agent s_{left} shouts, s_{left} receives the left bundle L. Allocate the remaining items according to $\mathsf{Lumpy}(N \setminus \{s_{\text{left}}\}, v_r, P(v_{\ell+1}, v_m))$.



Step 3. Delete the left-most point of the middle bundle, i.e., set $M = \{v_{\ell+2}, v_{\ell+3}, \dots, v_{r-1}\}$.
If there are zero or one shouters, go to Step 4. If at least two agents shout, we show below that there is a shouter s who is a middle agent over $P(v_{\ell+1}, v_m)$. Then, allocate L to a shouter s_{left} distinct from s. Let the agent c, distinct from s and s_{left}, choose a preferred bundle among $\{v_{\ell+1}\} \cup M$ and $\{v_r\} \cup R$. Agent s receives the other bundle.



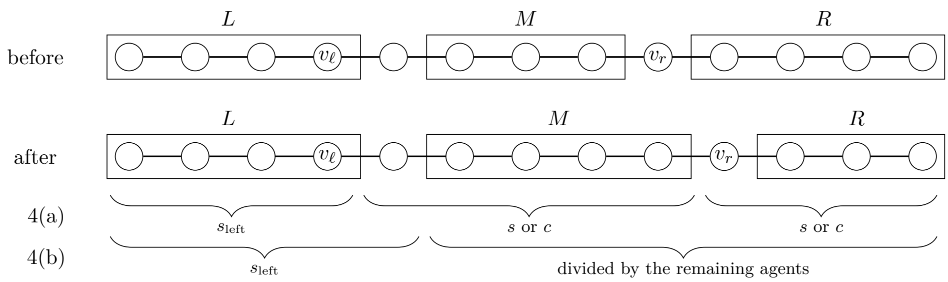
Step 4. If v_r is the median lumpy tie over $P(v_{\ell+2}, v_m)$, go to the following cases (a)–(d). If v_r is not the median lumpy tie over $P(v_{\ell+2}, v_m)$, set $r = r + 1$, $M = \{v_{\ell+2}, v_{\ell+3}, \dots, v_{r-1}\}$, and $R = \{v_{r+1}, v_{r+2}, \dots, v_m\}$; then, consider the following cases (a)–(d).

a) If at least two agents shout, find a shouter s who did not shout at the previous step. If there is a shouter s_{left} who shouted at the previous step, s_{left} receives L; else, give L to an arbitrary shouter s_{left} distinct from s. The agent c distinct from s and s_{left} choose a preferred bundle among $\{v_{\ell+1}\} \cup M$ and $\{v_r\} \cup R$, breaking ties in favour of the former option. Agent s receives the other bundle.

b) If v_r is the median lumpy tie over $P(v_{\ell+2}, v_m)$ and only one agent s_{left} shouts, give $L \cup \{v_{\ell+1}\}$ to s_{left} and allocate the rest according to $\mathsf{Lumpy}(N \setminus \{s_{\text{left}}\}, v_r, P(v_{\ell+2}, v_m))$.

c) If v_r is the median lumpy tie over $P(v_{\ell+2}, v_m)$ but no agent shouts, go to Step 2.

d) Otherwise v_r is not the median lumpy tie over $P(v_{\ell+2}, v_m)$: Repeat Step 4.



Step 3.

- Agent s_{left} receives L and, because s_{left} shouted, does not envy the bundle $\{v_{\ell+1}\} \cup M$ up to good $v_{\ell+1}$, and does not envy the bundle $\{v_r\} \cup R$ up to good v_r.
- Agent c gets his preferred bundle among $\{v_{\ell+1}\} \cup M$ and $\{v_r\} \cup R$, and so does not envy agent s who receives the other bundle. Further, agent c does not envy agent s_{left} since c did not shout at the last Step 2 (where no-one shouted), which, since bundle L did not change in Step 3, means that c prefers either $\{v_{\ell+1}\} \cup M$ or R to L, and hence also prefers his chosen bundle to L.
- Agent s is a middle agent, so the lumpy tie of s over $P(v_{\ell+1}, v_m)$ is v_r, and hence by (10.3),

$$u_s(\{v_r\} \cup R) \geqslant u_s(\{v_{\ell+1}\} \cup M). \tag{10.5}$$

Now, agent s did not shout at the preceding Step 2 (when no-one shouted). However, s *does* shout after deleting $v_{\ell+1}$ from M. Since L and R have not changed, the reason s did not shout at Step 2 was that L is worse than the middle bundle during Step 2, so

$$u_s(\{v_{\ell+1}\} \cup M) > u_s(L). \tag{10.6}$$

Combining (10.5) and (10.6), we also have

$$u_s(\{v_r\} \cup R) > u_s(L).$$

Since s receives either $\{v_{\ell+1}\} \cup M$ or $\{v_r\} \cup R$, agent s does not envy agent s_{left} receiving L. Finally, from (10.5), agent s weakly prefers $\{v_r\} \cup R$ to $\{v_{\ell+1}\} \cup M$. Thus, if c picks $\{v_{\ell+1}\} \cup M$, then s does not envy c. On the other hand, if c picks the bundle $\{v_r\} \cup R$, then s does not envy c up to good v_r: we have $u_s(L) \geqslant u_s(R)$ since s shouts, and so by (10.6), also

$$u_s(\{v_{\ell+1}\} \cup M) > u_s(R).$$

Step 4(a). We first prove that if i is a shouter who did not shout in the previous step, then

$$u_i(\{v_r\} \cup R) > u_i(L) \geqslant u_i(M). \tag{10.7}$$

In the previous step (which was either Step 3 or Step 4), the middle bundle was $M \setminus \{v_{r-1}\}$ and the right bundle was $\{v_r\} \cup R$. (While Step 4 allows for the possibility that the middle and right bundles are not changed in Step 4, this is not the case if we enter Step 4(a): if the bundles are unchanged and two agents shout, these agents already shouted in Step 3, contradicting that we did not terminate then.) Since i did not shout with the middle and right bundles of the previous step, we have

$$u_i(M \setminus \{v_{r-1}\}) > u_i(L) \quad \text{or} \quad u_i(\{v_r\} \cup R) > u_i(L).$$

Since i is a shouter, $u_i(L) \geqslant u_i(M)$, so that the first case is impossible by monotonicity. Hence $u_i(\{v_r\} \cup R) > u_i(L)$, showing (10.7), when combined with $u_i(L) \geqslant u_i(M)$.

- Agent s_{left} receives L and does not envy other agents up to one good like in Step 3.
- Agent c gets his preferred bundle among $\{v_{\ell+1}\} \cup M$ and $\{v_r\} \cup R$, and so does not envy agent s who receives the other bundle. Agent c also does not envy s_{left}: If c is not a shouter, then c does not envy s_{left} because c prefers either M or R to L, and hence prefers his picked piece to L. If c is a shouter, then all three agents are shouters, and by choice of c, this means that c was not a shouter at the previous step, when there was at most one shouter. By (10.7), $u_c(\{v_r\} \cup R) > u_c(L)$, and hence

$$\max\{u_c(\{v_{\ell+1}\} \cup M), u_c(\{v_r\} \cup R)\} \geqslant u_c(L),$$

so that c does not envy s_{left}.

- Agent s does not envy others up to one good:
  - Suppose agent c strictly prefers $\{v_r\} \cup R$ to $\{v_{\ell+1}\} \cup M$. Then agent c's lumpy tie over $P(v_{\ell+1}, v_m)$ appears at or after v_r by definition of the lumpy tie. As we argued before, the bundles M and R were changed in the execution of Step 4, and r was increased by 1. Thus, v_r appears strictly after the median lumpy tie over $P(v_{\ell+1}, v_m)$. Thus, c is the right agent over $P(v_{\ell+1}, v_m)$. Hence s is either a left or middle agent over $P(v_{\ell+1}, v_m)$ since there is at most one right agent. Using (10.3) or (10.4), this implies

$$u_s(\{v_{\ell+1}\} \cup M) \geqslant u_s(\{v_r\} \cup R), \tag{10.8}$$

    so that s does not envy c.

    By definition of s, agent s did not shout in the previous step. By (10.7), $u_s(\{v_r\} \cup R) \geqslant u_s(L)$, so together with (10.8), we have $u_s(\{v_{\ell+1}\} \cup M) \geqslant u_s(L)$, so s does not envy s_{left}.
  - Suppose c weakly prefers $\{v_{\ell+1}\} \cup M$ to $\{v_r\} \cup R$. Then s receives the bundle $\{v_r\} \cup R$ (since c breaks ties in favour of $\{v_{\ell+1}\} \cup M$). By choice of s, agent s did not shout at the last step. So by (10.7), we have $u_s(\{v_r\} \cup R) > u_s(L)$ so that s does not envy s_{left}, and also by (10.7), we have $u_s(\{v_r\} \cup R) > u_s(M)$ so that s does not envy c up to item $v_{\ell+1}$.

Step 4(b).

- Agent s_{left} gets $L \cup \{v_{\ell+1}\}$ and does not envy the other agents (up to good v_r) as s_{left} shouts.
- Any agent $i \neq s_{\text{left}}$ is not a shouter, and thus prefers either M or R to L. Hence by Lemma 10.12 receives a bundle preferred to L, and so does not envy s_{left} up to item $v_{\ell+1}$.
  Agent i also does not envy the other agent $j \neq s_{\text{left}}$ up to one good by Lemma 10.12.

Thus, the allocation returned by any of the steps satisfies EF1.

Our algorithm can be implemented in $O(m)$ time: Each of steps 2, 3, and 4 will be executed at most m times (since ℓ and r can only be incremented m times). Each step execution only needs constant time: In each step, we need to check which agents shout, and this can be done in a constant number of queries to agents' valuations; also, in Step 4 we need to calculate the lumpy ties of the agents, but this can be done in amortised constant time, since during the execution of the algorithm, the position of each agent's lumpy tie can only move to the right. Finally, when enough agents shout, we can clearly compute and return the final allocation in $O(m)$ time. □

10.6. EF2 Existence for Any Number of Agents

For two or three agents, we have seen algorithms that are guaranteed to find an EF1 allocation on a path (and on traceable graphs). Both algorithms were adaptations of procedures that identify envy-free divisions in the cake-cutting problem. For the case of four or more agents, we face a problem: there are no known procedures that find connected envy-free division in cake-cutting if the number of agents is larger than three. However, in the divisible setting, a non-constructive existence result is known: Su [1999] proved, using Sperner's lemma, that for any number of agents, a connected envy-free division of a cake always exists. One might try to use this result as a black box to obtain a fair allocation for the indivisible problem on a path: Translate an indivisible instance with additive valuations into a divisible cake (where each item corresponds to a region of the cake), obtain an envy-free division of the cake, and round it to get an allocation of the items. Suksompong [2017] followed this approach and showed that the result is an allocation where any agent i's envy $u_i(A(j)) - u_i(A(i))$ is at most $2u_{\max}$, where $u_{\max}$ is the maximum valuation for a single item.

In this section, rather than using Su's [1999] result as a black box, we directly apply Sperner's lemma to the indivisible problem. This allows us to obtain a stronger fairness guarantee: We show that on paths (and on traceable graphs), there always exists an EF2 allocation.[4] An allocation is EF2 if any agent's envy can be avoided by removing up to two items from the envied bundle. Again, we only allow removal of items if this operation leaves a connected bundle. For example, on a path, if agent i envies the bundle of agent j, then i does not envy that bundle once we remove its two endpoints. The formal definition for general graphs is as follows.

Definition 10.14 (EF2: envy-freeness up to two outer goods)**.** An allocation A satisfies EF2 if for any pair $i, j \in N$ of agents, either $|A(j)| \leqslant 1$, or there are two goods $u, v \in A(j)$ such that $A(j) \setminus \{u, v\}$ is connected and $u_i(A(i)) \geqslant u_i(A(j) \setminus \{u, v\})$.

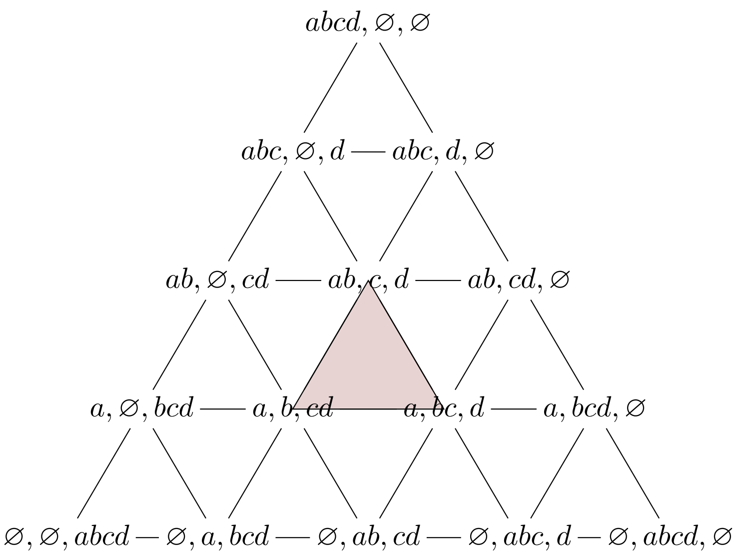


Figure 10.4.: Connected partitions form a subdivided simplex

Let us first give a high-level illustration with three agents of how Sperner's lemma can be used to find low-envy allocations. Given a path, say $P = (a, b, c, d)$, the family of connected partitions of P can naturally be arranged as the vertices of a subdivided simplex, as in Figure 10.4 on the right. For each of these partitions, each agent i *labels* the corresponding vertex by the index of a bundle from that partition that i most-prefers. For example, the top vertex will be labelled as "index 1" by all agents, since they all most-prefer the left-most bundle in $(abcd, \varnothing, \varnothing)$. Now, Sperner's lemma will imply that at least one of the simplices (say the shaded one) is "fully-labelled", which means that the first agent most-prefers the left-most bundle at one vertex, the second agent most-prefers the middle bundle at another vertex, and the third agent most-prefers the right-most bundle at the last vertex. Notice that the partitions at the corner points of the shaded simplex are all "similar" to each other (they can be obtained from each other by moving only 1 item). Hence, we can "round" the corner-partitions into a common allocation A^*, say by picking one of the corner partitions arbitrarily and then allocating bundles to agents according

[4]To see that EF2 is a stronger property than bounding envy up to $2u_{\max}$, consider a path of four items and two agents with additive valuations 1–10–2–2. The allocation $(1, 10\text{–}2\text{–}2)$ is not EF2, but the first agent has an envy of $13 < 20 = 2u_{\max}$.

to the labels. The resulting allocation has the property that any agents' envy can be eliminated by moving at most one good.[5]

The argument sketched above does not yield an EF1 nor even an EF2 allocation. Intuitively, the problem is that the connected partitions at the corners of the fully-labelled simplex are "too far apart", so that no matter how we round the corner partitions into a common allocation A^*, some agents' bundles will have changed too much, and so we cannot prevent envy even up to one or two goods. In the following, we present a solution to this problem, by considering a finer subdivision: we introduce $n-1$ knives which move in half-steps (rather than full steps), and which might 'cover' an item so that it appears in none of the bundles. The result is that the partial partitions in the corners of the fully-labelled simplex are closer together, and can be successfully rounded into an EF2 allocation A^*.

In our approach, we use a specific triangulation (Kuhn's triangulation). This triangulation has the needed property that the partitions at the corners of sub-simplices are close together, and adjacent partitions can be obtained from each other in a natural way. While this type of triangulation has also been used in cake-cutting [e.g., Deng et al., 2012], there it was only used to speed up algorithms (compared to the barycentric subdivision used by Su [1999]), not to obtain better fairness properties.

10.6.1. Sperner's lemma

We start by formally introducing Sperner's lemma. Let $\text{conv}(\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_k)$ denote the convex hull of k vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_k$. An *$n$-simplex* is an n-dimensional polytope which is the convex hull of its $n+1$ *main vertices.* A *k-face* of the n-simplex is the k-simplex formed by the span of any subset of $k+1$ main vertices. A *triangulation* T of a simplex S is a collection of sub-n-simplices whose union is S with the property that the intersection of any two of them is either the empty set, or a face common to both. Each of the sub-simplices $S^* \in T$ is called an *elementary* simplex of the triangulation T. We denote by $V(T)$ the set of vertices of the triangulation T, which is the union of vertices of the elementary simplices of T.

Let T be some fixed triangulation of an $(n-1)$-simplex $S = \text{conv}(\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_n)$. A *labeling function* is a function $L : V(T) \to [n]$ that assigns a number in $[n]$ (called a *colour*) to each vertex of the triangulation T. A labeling function L is called *proper* if

- For each main vertex $\boldsymbol{v}_i$ of the simplex, L assigns colour i to $\boldsymbol{v}_i$: $L(\boldsymbol{v}_i) = i$; and
- $L(\boldsymbol{v}) \neq i$ for any vertex $\boldsymbol{v} \in V(T)$ belonging to the $(n-2)$-face of S not containing $\boldsymbol{v}_i$.

Sperner's lemma states that if L is a proper labeling function, then there exists an elementary simplex of T whose vertices all have different labels.

We will consider a generalised version of Sperner's lemma, proved, for example, by Bapat [1989]. In this version, there are n labeling functions $L_1, \dots, L_n$, and we are looking for an elementary simplex which is *fully-labelled* for some way of assigning labeling functions to vertices, where we must use each labeling function exactly once. The formal definition is as follows.

Definition 10.15 (Fully-labelled simplex)**.** Let T be a triangulation of an $(n-1)$-simplex, and let $L_1, \dots, L_n$, be labeling functions. An elementary simplex S^* of T is *fully-labelled* if we can write $S^* = \text{conv}(\boldsymbol{v}_1^*, \boldsymbol{v}_2^*, \dots, \boldsymbol{v}_n^*)$ such that there exists a permutation $\phi : [n] \to [n]$ with

$$L_i(\boldsymbol{v}_i^*) = \phi(i) \quad \text{for each } i \in [n].$$

[5] One can generalise this argument to show that on paths, there exists an allocation A satisfying a weak form of EF1: for any $i, j \in [n]$, we have $u_i(I_i \cup \{g_i\}) \geqslant u_i(I_j \setminus \{g_j\})$ for some items g_i, g_j such that $I_i \cup \{g_i\}$ and $I_j \setminus \{g_j\}$ are connected. For additive valuations, this implies that envy is bounded by $u_i(g_i) + u_i(g_j) \leqslant 2u_{\max}$, which is Suksompong's [2017] result.

Our generalised version of Sperner's lemma guarantees the existence of a fully-labelled simplex.

Lemma 10.16 (Generalised Sperner's Lemma)**.** *Let T be a triangulation of an $(n-1)$-simplex S, and let $L_1, \dots, L_n$ be proper labeling functions. Then there is a fully-labelled simplex S^* of T.*

10.6.2. Existence of EF2 allocations

Suppose that our graph G is a path $P = (1, 2, \dots, m)$, where the items are named by integers. We assume that $m \geqslant n$, so that there are at least as many items as agents (when $m < n$ it is easy to find EF1 allocations). Our aim is to cut the path P into n intervals (bundles) $I_*^1, I_*^2, \dots, I_*^n$. Throughout the argument, we will use superscripts to denote indices of bundles; index 1 corresponds to the left-most bundle and index n corresponds to the right-most bundle.

Construction of the triangulation. Consider the $(n-1)$-simplex[6]

$$S_m = \{\, \boldsymbol{x} \in \mathbb{R}^{n-1} : \tfrac{1}{2} \leqslant x^1 \leqslant x^2 \leqslant \ldots \leqslant x^{n-1} \leqslant m + \tfrac{1}{2} \,\}. \tag{10.9}$$

We construct a triangulation T_{half} of S_m whose vertices $V(T_{\text{half}})$ are the points $\boldsymbol{x} \in S_m$ such that each x^j is either integral or half-integral, namely,

$$V(T_{\text{half}}) = \{\boldsymbol{x} \in S_m : x^j \in \{\tfrac{1}{2}, 1, \tfrac{3}{2}, 2, \tfrac{5}{2} \dots, m, m + \tfrac{1}{2}\} \text{ for all } j \in [n]\}.$$

For reasons that will become clear shortly, we call a vector $\boldsymbol{x} \in V(T_{\text{half}})$ a *knife position*.

Using Kuhn's triangulation [Kuhn, 1960, see also Scarf, 1982, Deng et al., 2012], we can construct T_{half} so that each elementary simplex $S' \in T_{\text{half}}$ can be written as $S' = \text{conv}(\boldsymbol{x}_1, \boldsymbol{x}_2, \dots \boldsymbol{x}_n)$ such that there exists a permutation $\pi : [n] \to [n]$ with

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \tfrac{1}{2}\mathbf{e}^{\pi(i)} \quad \text{for each } i \in [n-1], \tag{10.10}$$

where $\mathbf{e}^j = (0, \dots, 1, \dots, 0)$ is the j-th unit vector. We give an interpretation of (10.10) shortly.

Each vertex $\boldsymbol{x} = (x^1, x^2, \dots, x^{n-1}) \in V(T_{\text{half}})$ of the triangulation T_{half} corresponds to a partial partition $A(\boldsymbol{x}) = (I^1(\boldsymbol{x}), I^2(\boldsymbol{x}), \dots, I^n(\boldsymbol{x}))$ of P where

$$I^j(\boldsymbol{x}) := \{y \in \{1, 2, \dots, m\} : x^{j-1} < y < x^j\},$$

writing $x^0 = \frac{1}{2}$ and $x^n = m + \frac{1}{2}$ for convenience. Note the strict inequalities in the definition of $I^j(\boldsymbol{x})$. Intuitively, $\boldsymbol{x}$ specifies the location of $n-1$ knives that cut P into n pieces. If x^j is integral, that is $x^j \in \{1, \dots, m\}$, then the j-th knife 'covers' the item x^j, which is then part of neither $I^j(\boldsymbol{x})$ nor $I^{j+1}(\boldsymbol{x})$. This is why $A(\boldsymbol{x})$ is a *partial* partition. Since there are only $n-1$ knives but $m \geqslant n$ items, not all items are covered, so at least one bundle is non-empty.

Property (10.10) means that, if we visit the knife positions $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots \boldsymbol{x}_n$ at the corners of an elementary simplex in the listed order, then at each step exactly one of the knives moves by half a step, and each knife moves only at one of the steps.

Construction of the labeling functions. We now construct, for each agent $i \in [n]$, a labeling function $L_i : V(T_{\text{half}}) \to [n]$. The function L_i takes as input a vertex $\boldsymbol{x}$ of the triangulation T_{half} (interpreted as the partial partition $A(\boldsymbol{x})$), and returns a colour in $[n]$. The colour will specify the index of a bundle in $A(\boldsymbol{x})$ that agent i likes most. Formally,

$$L_i(\boldsymbol{x}) \in \{j \in [n] : u_i(I^j(\boldsymbol{x})) \geqslant u_i(I^k(\boldsymbol{x})) \text{ for all } k \in [n]\}.$$

[6] The simplex S_m is affinely equivalent to the standard $(n-1)$-simplex $\Delta_{n-1} = \{(l_1, \dots, l_n) \geqslant 0 : \sum l_i = 1\}$ via $x_i = m \cdot (l_1 + l_2 + \dots + l_i) + \frac{1}{2}$. In these coordinates, l_i is the length of the ith piece (times $1/m$).

If there are several most-preferred bundles in $A(\boldsymbol{x})$, ties can be broken arbitrarily. However, we insist that the index $L_i(\boldsymbol{x})$ always corresponds to a non-empty bundle (this can be ensured since $A(\boldsymbol{x})$ always contains a non-empty bundle, and u_i is monotonic).

The labeling functions L_i are proper. For each $j \in [m]$, the main vertex $\boldsymbol{v}_j$ of the simplex S_m has the form $\boldsymbol{v}_j = (\frac{1}{2}, \dots, \frac{1}{2}, m+\frac{1}{2}, \dots, m+\frac{1}{2})$, where the first $j-1$ entries are $\frac{1}{2}$ and the rest are $m+\frac{1}{2}$. This vertex corresponds to a partition $A(\boldsymbol{v}_j)$ where $I^j(\boldsymbol{v}_j)$ contains all the items, hence is most-preferred (since u_i is monotonic and by our tie-breaking), and so $L_i(\boldsymbol{v}_j) = j$. Further, any vertex $\boldsymbol{x}$ belonging to the $(n-2)$-face of S_m not containing $\boldsymbol{v}_j$ satisfies $x^{j-1} = x^j$, and thus corresponds to a partition $A(\boldsymbol{x})$ where $I^j(\boldsymbol{x})$ is empty, hence is *not* selected, and so $L_i(\boldsymbol{x}) \neq j$.

By the generalised version of Sperner's lemma (Lemma 10.16), there exists an elementary simplex $S^* = \operatorname{conv}(\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n)$ of the triangulation T_{half} which is fully-labelled, so that, for some permutation $\phi : [n] \to [n]$, we have $L_i(\boldsymbol{x}_i) = \phi(i)$ for all $i \in [n]$.

Translation into partial partitions. The fully-labelled elementary simplex S^* corresponds to a sequence $(A_1, A_2, \dots, A_n)$ of partial partitions of P, which we call the *Sperner sequence*, where $A_i = (I_i^1, \dots, I_i^n) := A(\boldsymbol{x}_i)$ for each $i \in [n]$. An example of a Sperner sequence is shown in Figure 10.5, which also illustrates other concepts that we introduce shortly. From the labeling, for each agent $i \in [n]$, since $L_i(\boldsymbol{x}_i) = \phi(i)$, the bundle with index $\phi(i)$ in the partition A_i is a best bundle for i:

$$u_i(I_i^{\phi(i)}) \geqslant u_i(I_i^j) \quad \text{for each } j \in [n]. \tag{10.11}$$

Now, for each $j \in [n]$, we define the *basic bundle* $B^j := I_1^j \cap \dots \cap I_n^j$ to be the bundle of items that appear in the j-th bundle of every partition in the Sperner sequence. The set of basic bundles is a partial partition. Let us analyse the items between basic bundles.

From (10.10), each of the $n-1$ knives moves exactly once, by half a step, while passing through the Sperner sequence $(A_1, A_2, \dots, A_n)$. Thus, the numbers $x_1^j, \dots, x_n^j$ take on two different values, one of which is integral and the other half-integral. We write y^j for the integral value (so $y^j = x_i^j$ for some $i \in [n]$), and call y^j a *boundary item*. The j-th knife covers the item y^j in some, but not all, of the partial partitions in the Sperner sequence. Now, there are two cases:

(a) $x_1^j = \dots = x_i^j = y^j - \frac{1}{2}$ and then $x_{i+1}^j = \dots = x_n^j = y^j$ for some $i \in [n]$, so that y^j never occurs in the j-th bundle in the Sperner sequence but sometimes occurs in the $j+1$st bundle, or

(b) $x_1^j = \dots = x_i^j = y^j$ and then $x_{i+1}^j = \dots = x_n^j = y^j + \frac{1}{2}$ for some $i \in [n]$, so that y^j sometimes occurs in the j-th bundle in the Sperner sequence but never occurs in the $j+1$st bundle.

Since y^j is sometimes covered by a knife, it is not part of any basic bundle. However, we have that

$$B^j \subseteq I_i^j \subseteq \{y^{j-1}\} \cup B^j \cup \{y^j\} \quad \text{for every } i, j \in [n]. \tag{10.12}$$

Rounding into a complete partition. We now construct a complete partition $(I_*^1, I_*^2, \dots, I_*^n)$ of the path P. We define each bundle as follows:

$$I_*^j := I_1^j \cup \dots \cup I_n^j \quad \text{for each } j \in [n].$$

Thus, the bundle I_*^j contains the basic bundle B^j, plus all of the boundary items y^{j-1} or y^j that occur in the j-th bundle at some point of the Sperner sequence. Precisely, for each boundary item y^j, $j \in [n-1]$, the item y^j is placed in bundle I_*^{j+1} in case (a) above, and it is placed in bundle I_*^j in case (b). So the resulting partition is well-defined: every item is allocated to exactly one bundle.

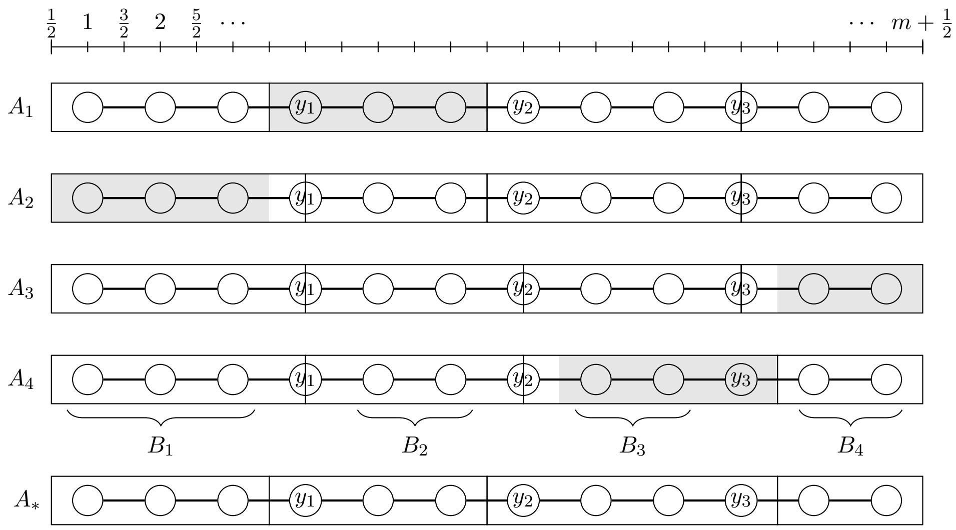


Figure 10.5.: Example of the Sperner sequence $A_1, \ldots, A_4$ for $n = 4$, as well as the derived partition A_*. Vertical lines indicate the positions x_i^1, x_i^2, x_i^3 of the knives, $i = 1, \ldots, 4$. Shaded in gray, for $i = 1, \ldots, 4$, is the bundle $I_i^{\phi(i)}$ selected by agent i as their favourite bundle in A_i; here $\phi(1) = 2, \phi(2) = 1, \phi(3) = 4, \phi(4) = 3$.

An EF2 allocation. We first show that the partition $(I_*^1, I_*^2, \ldots, I_*^n)$ is such that agents' expectations about the value of the bundles I_*^j are approximately correct (namely, correct up to two items):

$$u_i(I_*^j) \geqslant u_i(I_i^j) \geqslant u_i(B^j) \quad \text{for every agent } i \in [n] \text{ and every } j \in [n]. \tag{10.13}$$

This follows by monotonicity of u_i, since $I_*^j = I_1^j \cup \cdots \cup I_n^j \supseteq I_i^j \supseteq B^j$ by (10.12).

Now, based on the partition, we can define an allocation A_* by $A_*(i) = I_*^{\phi(i)}$ for each agent $i \in [n]$. Thus, each agent i receives the bundle in the complete partition corresponding to i's most-preferred index $\phi(i)$. We prove that A_* satisfies EF2: For any pair $i, j \in [n]$ of agents, we have

$$\begin{aligned} u_i(A_*(i)) = u_i(I_*^{\phi(i)}) &\geqslant u_i(I_i^{\phi(i)}) && \text{by (10.13)} \\ &\geqslant u_i(I_i^{\phi(j)}) && \text{by (10.11)} \\ &\geqslant u_i(B^{\phi(j)}) && \text{by (10.13)} \\ &= u_i(A_*(j) \setminus \{y^{j-1}, y^j\}). && \text{by (10.12)} \end{aligned}$$

Hence, we have proved the main result of this section:

Theorem 10.17. *On a path, for any number of agents with monotone valuation functions, a connected EF2 allocation exists.*

10.7. EF1 Existence for Identical Valuations

A special case of the fair division problem is the case of *identical valuations*, where all agents have the same valuation for the goods: for all agents $i, j \in N$ and every bundle $I \in \mathcal{C}(V)$, we have $u_i(I) = u_j(I)$. We then write $u(I)$ for the common valuation of bundle I. The case of identical valuations often allows for more positive results and an easier analysis. Indeed, we can prove that, for identical valuations and *any* number of agents, an EF1 allocation connected on a path is guaranteed to exist. We further show that such an allocation can be found in polynomial time.

Our argument, though intuitive, is not as straightforward as one might think. For example, one might guess that in the restricted case of identical valuations, egalitarian allocations are EF1. However, the leximin-optimal connected allocation may fail EF1: Consider a path with five items and additive valuations 1–3–1–1–1 shared by three agents. The unique leximin allocation is (1, 3, 1–1–1), which induces envy even up to one good. The same allocation also uniquely maximises Nash welfare, so the Nash optimum also does not guarantee EF1. This is in contrast to the results of Biswas and Barman [2018] who consider allocations of items into bundles that satisfy matroid constraints (rather than our connectivity constraints), and find that the Nash optimum satisfies EF1 under matroid constraints and the assumption of identical valuations.

Maximising an egalitarian objective seemed promising because it ensures that no-one is too badly off, and therefore has not much reason to envy others. The problem is that some bundles might be too desirable. To fix this, we could try to reallocate items so that no bundle is too valuable. This is exactly the strategy of our algorithm: It starts with a leximin allocation, and then moves items from high-value bundles to lower-value bundles, until the result is EF1. In more detail, the algorithm identifies one agent i who is worst-off in the leximin allocation, and then adjusts the allocation so that i does not envy any other bundle up to one good. The algorithm does this by going through all bundles in the allocation, outside-in, and if i envies a bundle I^j even up to one good, it moves one item from I^j inwards (in i's direction), see Figure 10.6. As we will show, a key invariant preserved by the algorithm is that the value of I^i never increases, and i remains worst-off. Thus, since i does not envy others up to one good, the allocation at the end is EF1.

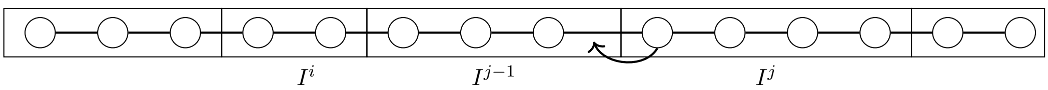


Figure 10.6.: If i envies j even up to one good, Algorithm 9 takes an item out of bundle I^j and moves it in i's direction.

Formally, a *leximin allocation* is an allocation which maximises the lowest utility of an agent; subject to that it maximises the second-lowest utility, and so on. In particular, if the highest achievable minimum utility is u_L, then the leximin allocation is such that every agent has utility at least u_L, and the number of agents with utility exactly u_L is minimum.

Algorithm 9 LEXIMIN-TO-EF1

Input: a path $P = (v_1, v_2, \dots, v_m)$, and identical valuations
Output: an EF1 connected allocation of P
Let $A = (I^1, \dots, I^n)$ be a leximin allocation of P
Fix an agent i with minimum utility in A, i.e., $u(I^i) \leqslant u(I^j)$ for all $j \in [n]$
for $j = 1, \dots, i-1$ **do**
  if i envies I^j even up to one good, i.e., $u(I^i) < u^-(I^j)$ **then**
    repeatedly delete the right-most item of I^j and add it to I^{j+1} until $u(I^i) \geqslant u^-(I^j)$
for $j = n, \dots, i+1$ **do**
  if i envies I^j even up to one good, i.e., $u(I^i) < u^-(I^j)$ **then**
    repeatedly delete the left-most item of I^j and add it to I^{j-1} until $u(I^i) \geqslant u^-(I^j)$
return A

Theorem 10.18. *For identical valuations on a path, Algorithm 9 finds an EF1 allocation.*

Proof. For an allocation $A = (I^1, \dots, I^n)$, write $u_L(A) := \min_{j \in N} u(I^j)$ for the minimum utility obtained in A, and write $L(A) := \{j \in [n] : u(I^j) = u_L(A)\}$ for the set of agents (*losers*) who obtain this utility. For the leximin allocation A_{leximin} obtained at the start of the algorithm, write $u_L^* := u_L(A_{\text{leximin}})$ and $L^* := L(A_{\text{leximin}})$. Note that by leximin-optimality, for every allocation A we must have $u_L(A) \leqslant u_L^*$, and if $u_L(A) = u_L^*$ then $|L(A)| \geqslant |L^*|$. Let $i \in L^*$ be the agent fixed at the start of the algorithm, and recall the definition of u^- from (10.1).

Claim 1. Throughout the algorithm, $u_L(A) = u_L^*$ and $L(A) = L^*$.

The claim is true before we start the for-loops. Suppose the claim holds up until some iteration of the first for-loop, and we now move an item from I^j to I^{j+1}, obtaining the new bundles I^j_{new} and I^{j+1}_{new} in the new allocation A_{new}. Then $u(I^j_{\text{new}}) \geqslant u^-(I^j) > u(I^i) = u_L^*$, where the strict inequality holds by the if- and until-clauses. Since no agent other than j has become worse-off in A_{new}, it follows that $u_L(A_{\text{new}}) \geqslant u_L(A) = u_L^*$. As noted, by optimality of u_L^*, we have $u_L(A_{\text{new}}) \leqslant u_L^*$. Hence $u_L(A_{\text{new}}) = u_L^*$. Thus, by optimality of L^*, we have $|L(A_{\text{new}})| \geqslant |L^*|$. Because agent j has not become a loser (since $u(I^j_{\text{new}}) > u_L^*$ as shown before) and no other agent has become a loser, we have $L(A_{\text{new}}) \subseteq L(A) = L^*$. Thus $L(A_{\text{new}}) = L^*$, as required. The second for-loop is handled similarly.

Claim 2. After both for-loops terminate, agent i does not envy any agent up to one good.

For any $j \neq i$, agent i does not envy j up to one good immediately after the relevant loop has handled j, and at no later stage of the algorithm does I^j change.

It follows that the allocation A returned by the algorithm is EF1: By Claim 1, we have $i \in L(A)$, so that $u(I^j) \geqslant u(I^i)$ for all $j \in [n]$. By Claim 2, agent i does not envy any other agent up to one good, so that $u(I^i) \geqslant u^-(I^k)$ for all $k \in [n]$. Hence, for all $j, k \in [n]$, we have $u(I^j) \geqslant u^-(I^k)$, that is, no agent envies another agent up to one good. □

Algorithm 9 can be implemented to run in polynomial time, because with identical valuations, one can use dynamic programming to find a leximin allocation in time $O(m^2n^2)$, and the remainder of Algorithm 9 takes time $O(mn)$, since each item is moved at most n times. A slight speed-up can be achieved by observing that the proof of Theorem 10.18 only needed that the initial allocation optimises the egalitarian welfare u_L and minimises the cardinality of the set L of losers. Such an allocation can be found by dynamic programming in time $O(m^2n)$.

The reallocation stage of our algorithm bears some similarity to an argument by Suksompong [2017, Theorem 2] which shows that a $u_{\max}$-equitable allocation exists. In a very recent paper, Oh et al. [2019, Lemma C.2] proved independently, using an inductive argument, that EF1 allocations on a path exist for identical valuations. Their procedure can also be implemented in polynomial time.

10.8. Conclusion

We have studied the existence of EF1 allocations under connectivity constraints imposed by an undirected graph. We have shown that for two or three agents, an EF1 allocation exists if the graph is traceable. For any number of agents, we also proved that traceable graphs guarantee the existence of an EF2 allocation, and they guarantee the existence of an EF1 allocation with identical valuations. In the published version of this chapter, we show that our Sperner approach can be adapted to prove that EF1 allocations exist for four agents [Bilò et al., 2018].

There are several questions left open. Most obviously, we do not know whether EF1 allocations on a path exist for five or more agents. Extensive computer sampling did not find an example where no EF1 allocation exists. One can also ask whether different topological restrictions (e.g., cycles), or restricted preference domains (e.g., binary utilities) can allow us to obtain EF1 existence guarantees for $n \geqslant 5$.

While many of our procedures admit efficient implementations for finding fair allocations, for our results based on Sperner's lemma we do not know of algorithms better than a naive search through all connected allocations. For divisible cake-cutting, Deng et al. [2012] proved that it is

PPAD-complete to find an envy-free allocation. What is the complexity of finding an EF1 or EF2 allocation in our setting of items arranged on a path? Moving away from paths, it would be interesting to study the complexity of deciding, given a graph and (say) additive valuations of the agents, whether there exists a connected EF1 allocation.

While we were able to characterise the class of graphs guaranteeing EF1 in the two-agent case, we have no characterisation for three or more agents. For three agents, there are non-traceable graphs that guarantee EF1, such as the star with three leaves. Understanding such examples, and designing EF1 procedures for them, is an interesting research direction.

In this chapter, we have only considered *goods*, with monotonic valuations. The setting where some or all items are undesirable (so-called *chores*) is also of interest [Bogomolnaia et al., 2016, 2017, Meunier and Zerbib, 2018, Segal-Halevi, 2018, Aziz et al., 2019c]. In the model with connectivity constraints, Aziz et al. [2019c] showed that on a path, a connected *Prop1* allocation always exists (a weaker requirement than EF1). Whether EF1 connected allocations exist in this more general domain is an intriguing question. Recently, for cake-cutting, Segal-Halevi [2018] noted that Su's approach using Sperner's lemma is not applicable to establish the existence of an envy-free connected allocation, when agents consider some parts of the cake undesirable. However, the existence of such allocations can be proved using other methods [Segal-Halevi, 2018, Meunier and Zerbib, 2018], and these may be translateable to the indivisible setting.

11. Pareto-Optimality and Computational Complexity

We study the problem of finding a connected allocation of indivisible items that is Pareto-optimal. We focus on additive valuations. While it is easy to find an efficient allocation when the underlying graph is a path or a star, the problem is NP-hard for many other graph topologies, even for trees of bounded pathwidth and bounded diameter. We show that on a path, there are instances where no Pareto-optimal allocation satisfies envy-freeness up to one good.

11.1. Introduction

In mechanism design, Pareto-optimality is a basic desideratum: if we select an outcome that is Pareto-dominated by another, users will justifiably complain. In simple settings, it is computationally trivial to find a Pareto-optimum (e.g., via serial dictatorship). Thus, it is usually sought to be satisfied together with other criteria (like fairness or welfare maximisation). However, in more complicated settings, even Pareto-optimality may be elusive.

In this chapter, we study Pareto-optimality for item allocation into connected bundles: Given agents' preferences over (connected) bundles, we wish to find an allocation that is *Pareto-optimal*, that is, a connected allocation such that there is no other *connected* allocation which makes some agent strictly better off while making no agent worse off. Now, in the standard setting without connectivity constraints and with additive valuations, it is straightforward to find Pareto-optima: For example, we can allocate each item to a person who has the highest valuation for it (maximising utilitarian social welfare in the process), or we can run a serial dictatorship. Neither of these approaches respects connectivity constraints. In fact, we show that it is NP-hard to construct a Pareto-optimal allocation under connectivity constraints, unless G is extremely simple. We will also study the combination of Pareto-optimality with the fairness axioms (MMS and EF1) that we studied in the previous chapter.

Related Work. The relation between efficiency and fairness with connected pieces has been studied for divisible items. Aumann and Dombb [2015] studied the utilitarian social welfare of fair allocations under connectivity constraints. The papers by Bei et al. [2012] and Aumann et al. [2013] considered the computational complexity of finding an allocation with connected pieces maximising utilitarian social welfare. Bei et al. [2012] showed that utilitarian social welfare is inapproximable when requiring that the allocation satisfy proportionality; however, without the proportionality requirement, Aumann et al. [2013] proved that there is a polynomial-time constant-factor approximation algorithm for finding an allocation maximising utilitarian social welfare. The algorithm by Aumann et al. [2013] works also for indivisible items and so applies to our setting when G is a path. A paper by Conitzer et al. [2004] considers combinatorial auctions; translated to our setting, their results imply that one can find a Pareto-optimal connected allocation in polynomial time, when G is a graph of bounded treewidth and agents have *unit demand*: each agent specifies a connected demanded bundle such that agents have positive utility if and only if they obtain a superset of the demanded bundle.

	general	complete	tree	path
PO	NP-hard*	poly-time	NP-hard*	poly-time
PO & MMS	NP-hard*		NP-hard*	NP-hard*
PO & EF1	NP-hard	poly-time†	NP-hard	NP-hard

Table 11.1.: Overview of our complexity results. Hardness results marked * hold under Turing reductions. The result † refers to a pseudo-polynomial algorithm by Barman et al. [2018]. Our hardness results hold even for additive and binary valuations. In this chapter, we only prove the hardness results of the first row, leaving the rest to the published version [Igarashi and Peters, 2019].

With no connectivity constraints, Aziz et al. [2016] studied the computational complexity of finding Pareto-improvements of a given allocation when agents have additive preferences. Technically, our hardness proofs use similar techniques to hardness proofs obtained by Aziz et al. [2013] in the context of hedonic games.

11.2. Preliminaries

Recall that a valuation function $u_i : \mathcal{C}(V) \to \mathbb{R}$ is *additive* if $u_i(X) = \sum_{v \in X} u_i(\{v\})$ for each $X \in \mathcal{C}(V)$. We write $u_i(v) = u_i(\{v\})$ for short. An additive valuation function is *binary* if $u_i(v) \in \{0, 1\}$ for all $v \in V$. If an agent i has a binary valuation function, we say that i *approves* item v if $u_i(v) = 1$.

Given a connected allocation $\pi : N \to \mathcal{C}(V)$ and a subset N' of agents, we denote by $\pi|_{N'}$ the allocation restricted to N'.

Given an allocation π, another allocation π' is a *Pareto-improvement* of π if $u_i(\pi'(i)) \geqslant u_i(\pi(i))$ for all $i \in N$ and $u_j(\pi'(j)) > u_j(\pi(j))$ for some $j \in N$. We say that a connected allocation π is *Pareto-optimal* (or *Pareto-efficient*, or *PO* for short) if there is no *connected* allocation that is a Pareto-improvement of π. The *utilitarian social welfare* of an allocation π is $\sum_{i \in N} u_i(\pi(i))$. It is easy to see that a connected allocation which maximises utilitarian social welfare among connected allocations is Pareto-optimal.

Some graph-theoretic terminology: Given a graph $G = (V, E)$ and a subset $X \subseteq V$ of vertices, we denote by $G \setminus X$ the subgraph of G induced by $V \setminus X$. The *diameter* of G is the maximum distance between any pair of vertices.

11.3. Finding Some Pareto-Optimal Allocation

We start by considering the problem of producing *some* Pareto-optimal allocation, without imposing any additional constraints on the quality of this allocation. When there are no connectivity requirements (equivalently, when G is a complete graph) and valuations are additive, this problem is trivial: Simply allocate each item v separately to an agent i who has a highest valuation $u_i(v)$ for v. The resulting allocation maximises utilitarian social welfare and is thus Pareto-optimal. When G is not complete, this procedure can produce disconnected bundles. We could try to give all items to a single agent (if the graph G is connected), but the result need not be Pareto-optimal if that agent has zero value for some items. Is it still possible to find a Pareto-optimal allocation for specific graph topologies in polynomial time?

11.3.1. Paths and Stars

For very simple graphs and additive valuations, the answer is positive. Our first algorithm works when G is a path. The algorithm identifies an agent i with a non-zero valuation for the item at the left end of the path G, and then allocates all items to i, except for any items at the right end of the path which i values at 0. We then recursively call the same algorithm to decide on how to allocate the remaining items.

Theorem 11.1. *When G is a path, and with additive valuations, a Pareto-optimal allocation can be found in polynomial time.*

Proof. The path G is given by $V = \{v_1, v_2, \ldots, v_m\}$ where $\{v_j, v_{j+1}\} \in E$ for each $j \in [m-1]$. For a subset V' of V, we denote by $\min V'$ the vertex of minimum index in V'.

We design a recursive algorithm $\mathcal{A}$ that takes as input a subset N' of agents, a subpath $G' = (V', E')$ of G, and a valuation profile $(u_i)_{i \in N'}$, and returns a Pareto-optimal allocation of the items in V' to the agents in N'. Without loss of generality, we may assume that there is an agent who likes the left-most vertex of G', i.e., $u_i(\min V') > 0$ for some $i \in N'$, since otherwise we can arbitrarily allocate that item later without affecting Pareto-optimality.

If $|N'| = 1$, then the algorithm allocates all items V' to the single agent. Suppose that $|N'| > 1$. The algorithm first finds an agent i who has positive value for $\min V'$; it then allocates to i a minimal connected bundle $V_i \subseteq V'$ containing all items in V' to which i assigns positive utility (so that $u_i(V_i) = u_i(V')$). To decide on the allocation of the remaining items, we apply $\mathcal{A}$ to the reduced instance $(N' \setminus \{i\}, G' \setminus V_i, (u_{i'})_{i' \in N' \setminus \{i\}})$.

We will prove by induction on $|N'|$ that the allocation produced by $\mathcal{A}$ is Pareto-optimal. This is clearly true when $|N'| = 1$. Suppose that $\mathcal{A}$ returns a Pareto-optimal allocation when $|N'| = k-1$; we will prove it for $|N'| = k$. Let π be the allocation returned by $\mathcal{A}$, where $\mathcal{A}$ chose agent i and allocated the bundle V_i before making a recursive call. Assume for a contradiction that there is a Pareto-improvement π' of π. Thus, in particular, $u_i(\pi'(i)) \geqslant u_i(\pi(i))$. By the algorithm's choice of the bundle V_i, we must have $V_i \subseteq \pi'(i)$ and $u_i(\pi'(i)) = u_i(\pi(i))$. Thus, there is an agent j' different from i who receives strictly higher value in π' than in π.

Now, since $G \setminus \pi'(i)$ is a subgraph of $G \setminus V_i$, the allocation $\pi'|_{N' \setminus \{i\}}$ is an allocation for the instance $I' = (N' \setminus \{i\}, G' \setminus V_i, (u_{i'})_{i' \in N' \setminus \{i\}})$. Also, we have

- $u_j(\pi'(j)) \geqslant u_j(\pi(j))$ for all agents $j \in N' \setminus \{i\}$; and
- $u_{j'}(\pi'(j')) > u_{j'}(\pi(j'))$ for some $j' \in N' \setminus \{i\}$.

Thus, $\pi'|_{N' \setminus \{i\}}$ is a Pareto-improvement of the allocation $\pi|_{N' \setminus \{i\}}$. But $\pi|_{N' \setminus \{i\}}$ is the allocation returned by $\mathcal{A}$ for the instance I', contradicting the inductive hypothesis that $\mathcal{A}$ returns Pareto-optimal allocations for $|N'| = k-1$. □

Another graph type for which we can find a Pareto-optimum is a star. In fact, we can efficiently calculate an allocation maximising utilitarian social welfare. Note that when G is a star, at most one agent can receive two or more items. This allows us to translate welfare maximisation into a bipartite matching instance.

Theorem 11.2. *When G is a star, and valuations are additive, a Pareto-optimal allocation can be found in polynomial time.*

Proof. We give an algorithm to find an allocation that maximises the utilitarian social welfare. Let G be a star with center vertex c and $m-1$ leaves. We start by guessing an agent $i \in N$ who receives the item c. By connectedness, every other agent can receive at most one (leaf) item. To allocate the leaf items, we construct a weighted bipartite graph H_i with bipartition $(N', V \setminus \{c\})$ where N' consists of agents $j \in N \setminus \{i\}$ together with $m-1$ copies $i_1, i_2, \ldots, i_{m-1}$ of agent i.

(These copies represent 'slots' in i's bundle.) Each edge of form $\{j, v\}$ for $j \in N \setminus \{i\}$ has weight $u_j(v)$ and each edge of form $\{i_k, v\}$ has weight $u_i(v)$.

Observe that each connected allocation in which i obtains c can be identified with a matching in H_i: Every leaf object is either matched with the agent receiving it, or is matched with some copy i_k of i if the object is part of i's bundle. Note that utilitarian social welfare of this allocation equals the total weight of the matching. Since one can find a maximum-weight matching in a bipartite graph in polynomial time (see, e.g., Korte and Vygen, 2006), we can find an allocation of maximum utilitarian social welfare efficiently. □

We have shown that finding a Pareto-optimum is easy for paths and for stars. An interesting open problem is whether the problem is also easy for caterpillars, a class of graphs containing both paths and stars. One might be able to combine the approaches of Theorems 11.1 and 11.2 to handle them, but the details are difficult. Note that caterpillars are precisely the graphs of pathwidth 1; we discuss a negative result about graphs of pathwidth 2 below. Another open problem is whether finding a Pareto-optimum is easy when G is a cycle.

11.3.2. Hardness Results

For more general classes of graphs, the news is less positive. We show that, unless P = NP, there is no polynomial-time algorithm which produces a Pareto-optimal allocation when G is an arbitrary tree, even for binary valuations. Notably, this result implies that it is NP-hard to find allocations maximising any type of social welfare (utilitarian, leximin, Nash) when G is a tree, since such allocations are also Pareto-optimal.

To obtain our hardness result, we first consider a more general problem which is easier to analyse, namely the case where G is a forest. Since a Pareto-optimum always exists, we cannot employ the standard approach of showing that a decision problem is NP-hard via many-one reductions. Instead, we show (by a Turing reduction) that a polynomial-time algorithm producing a connected Pareto-optimal allocation could be used to solve an NP-complete problem in polynomial time.

Theorem 11.3. *Unless* P = NP*, there is no polynomial-time algorithm which finds a Pareto-optimal connected allocation when G is a union of vertex-disjoint paths of size 3, even if valuations are binary and additive.*

Proof. We give a Turing reduction from Exact-3-Cover (X3C). Recall that an instance of X3C is given by a set of elements $X = \{x_1, x_2, \dots, x_{3r}\}$ and a family $\mathcal{S} = \{S_1, \dots, S_s\}$ of three-element subsets of X; it is a 'yes'-instance if and only if there is an *exact cover* $\mathcal{S}' \subseteq \mathcal{S}$ with $|\mathcal{S}'| = r$ and $\bigcup_{S \in \mathcal{S}'} S = X$. For a set $S \in \mathcal{S}$, order the three elements of S in some canonical way (e.g., alphabetically) and write S^1, S^2, S^3 for the elements in that order.

Given an instance $(X, \mathcal{S})$ of X3C, for each $S \in \mathcal{S}$, construct a path P_S on three vertices $v_{S,1}, v_{S,2}, v_{S,3}$. Let $G = \bigcup_{S \in \mathcal{S}} P_S$. For each element $x \in X$, we introduce an agent i_x with $u_{i_x}(v_{S,j}) = 1$ iff $S^j = x$, and $u_{i_x}(v_{S,j}) = 0$ otherwise. Thus, agent i_x approves all instances of x. We also introduce $s - r$ dummy agents $d_1, \dots, d_{s-r}$ who approve every item, so $u_{d_k}(v_{S,j}) = 1$ for all j, k, S. Note that for each agent i_x, a highest-value connected bundle has value 1, while for a dummy agent d_k, a highest-value connected bundle has value 3.

Suppose we had an algorithm $\mathcal{A}$ which finds a Pareto-optimal allocation. We show how to use $\mathcal{A}$ to solve X3C. Run $\mathcal{A}$ on the allocation problem constructed above to obtain a Pareto-optimal allocation π. We claim that the X3C instance $(X, \mathcal{S})$ has a solution iff

$$\begin{aligned} u_{i_x}(\pi(i_x)) &= 1 \text{ for all } x \in X \text{ and} \\ u_{d_k}(\pi(d_k)) &= 3 \text{ for all } k \in [s-r]. \end{aligned} \tag{11.1}$$

Since (11.1) is easy to check, this equivalence implies that $\mathcal{A}$ can be used to solve X3C, and hence our problem is NP-hard.

Suppose π satisfies (11.1). We construct a solution to the X3C instance. For each $k \in [s-r]$, since $u_{d_k}(\pi(d_k)) = 3$, we must have $\pi(d_k) = P_S$ for some $S \in \mathcal{S}$. Let $\mathcal{S}' = \{S \in \mathcal{S} : \pi(d_k) \neq P_S \text{ for all } k \in [s-r]\}$. Then $\mathcal{S}'$ is a solution: Clearly $|\mathcal{S}'| = r$; further, for every $x \in X$, we have that $\pi(i_x) \in P_S$ for some $S \in \mathcal{S}$, and since $u_{i_x}(\pi(i_x)) = 1$ by (11.1), this implies that $x \in S$. Thus, $\bigcup_{S \in \mathcal{S}'} S = X$. Hence, $\mathcal{S}'$ is a solution to the X3C instance $(X, \mathcal{S})$.

Conversely, suppose there is a solution $\mathcal{S}'$ to the instance of X3C, but suppose for a contradiction that π fails condition (11.1). Define the following allocation π^*: For each $x \in X$, identify a set $S \in \mathcal{S}'$ and an index $j \in [3]$ such that $S^j = x$ and set $\pi^*(i_x) = \{v_{S,j}\}$; next, write $\mathcal{S} \setminus \mathcal{S}' = \{S'_1, \dots, S'_{s-r}\}$ and set $\pi^*(d_k) = \{v_{S'_k,1}, v_{S'_k,2}, v_{S'_k,3}\}$ for each $k \in [s-r]$. Then π^* satisfies (11.1). Since π fails (11.1), the allocation π^* Pareto-dominates π, contradicting that π is Pareto-optimal. Hence, π satisfies (11.1), as desired. □

Building on this reduction, we find that it is also hard to find a Pareto-efficient allocation if G is a tree (rather than a forest).

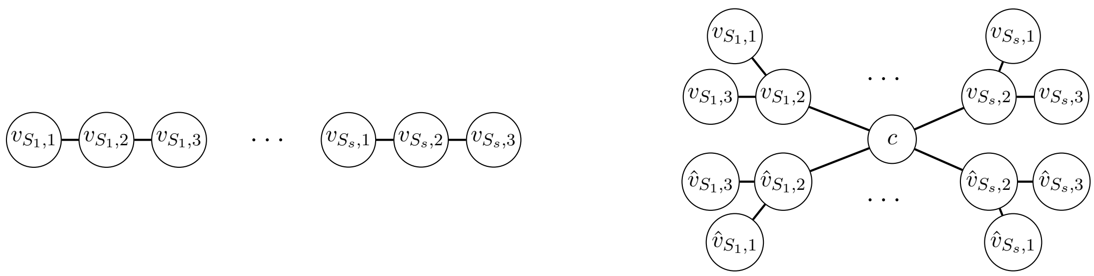


Figure 11.1.: Graphs constructed in the proofs of Theorem 11.3 (left) and Theorem 11.4 (right).

Theorem 11.4. *Unless* P = NP*, there is no polynomial-time algorithm which finds a Pareto-optimal connected allocation when G is a tree, even if valuations are binary and additive.*

Proof. To extend the reduction in the proof of Theorem 11.3 to trees, we first 'double' the reduction, by making a copy of each object and a copy of each agent with the same preference as the original agent. Specifically, given an instance $(X, \mathcal{S})$ of X3C, we create the same instance as in the proof of Theorem 11.3; that is, we make a path $P_S = (v_{S,1}, v_{S,2}, v_{S,3})$ for each $S \in \mathcal{S}$, and construct agent i_x for each $x \in X$ and dummy agents $d_1, d_2, \dots, d_{s-r}$ with the same binary valuations.

In addition, we make a path $\hat{P}_S$ of copies $\hat{v}_{S,1}, \hat{v}_{S,2}, \hat{v}_{S,3}$ of each $S \in \mathcal{S}$. We then make a copy $\hat{i}_x$ of each agent i_x $(x \in X)$ together with copies $\hat{d}_1, \hat{d}_2, \dots, \hat{d}_{s-r}$ of the dummy agents. We also introduce a new item c which serves as the center of a tree; specifically, we attach the center to the middle vertex $v_{S,2}$ of the path P_S, and the middle vertex $\hat{v}_{S,2}$ of the path $\hat{P}_S$, for each $S \in \mathcal{S}$. The resulting graph G is a tree consisting of $2r + 2|\mathcal{S}|$ paths of length 3, each attached to the vertex c by their middle vertex. See Figure 11.1.

No agent has positive value for the center item c. Copied agents only value copied objects and have the same valuations as the corresponding original agents, and non-copied agents only value non-copied objects. Formally, for each element $x \in X$, each $k \in [s-r]$, and each item v, agents' binary valuations are given as follows:

- $u_{i_x}(v) = 1$ iff $v = v_{S,j}$ and $S^j = x$;
- $u_{d_k}(v) = 1$ iff $v = v_{S,j}$ for some S, j;
- $u_{\hat{i}_x}(v) = 1$ iff $v = \hat{v}_{S,j}$ and $S^j = x$;

- $u_{\hat{d}_k}(v) = 1$ iff $v = \hat{v}_{S,j}$ for some S, j.

Write $N_o = \{ i_x : x \in X \} \cup \{d_1, d_2, \dots, d_{s-r}\}$ for the set of original agents, and $V_o = \bigcup_{S \in \mathcal{S}} \{v_{S,1}, v_{S,2}, v_{S,3}\}$ for the set of original items.

Suppose we had an algorithm $\mathcal{A}$ which finds a Pareto-optimal allocation. We show how to use $\mathcal{A}$ to solve X3C. Run $\mathcal{A}$ on the allocation problem constructed above to obtain a Pareto-optimum π. We may suppose without loss of generality that $c \notin \pi(i)$ for any $i \in N_o$, since otherwise we can swap the roles of the originals and the copies. We may further assume that each original agent $i \in N_o$ only receives original items, i.e., $\pi(i) \subseteq V_o$, since we can move any other items from $\pi(i)$ into other bundles without making anyone worse off. Hence, since $c \notin \pi(i)$, we see that $\pi(i) \subseteq P_S$ for some $S \in \mathcal{S}$ because $\pi(i)$ is connected in G. This shows that $u_{i_x}(\pi(i_x)) \leqslant 1$ for all $x \in X$ and $u_{d_k}(\pi(d_k)) \leqslant 3$ for all $k \in [s - r]$. We prove that the X3C instance has a solution iff

$$\begin{aligned} u_{i_x}(\pi(i_x)) &= 1 \text{ for all } x \in X \text{ and} \\ u_{d_k}(\pi(d_k)) &= 3 \text{ for all } k \in [s - r]. \end{aligned} \tag{11.2}$$

Since (11.2) is easy to check, this equivalence implies that $\mathcal{A}$ can be used to solve X3C, and hence our problem is NP-hard. If (11.2) holds, then the argument in the proof of Theorem 11.3 applies and shows that the X3C instance has a solution.

Conversely, suppose there is a solution $\mathcal{S}' \subseteq \mathcal{S}$ to the X3C instance. Then, as in the proof of Theorem 11.3, there is an allocation $\pi^* : N_o \to \mathcal{C}(V_o)$ of the original items to the original agents such that $u_{i_x}(\pi^*(i_x)) = 1$ for all $x \in X$ and $u_{d_k}(\pi^*(d_k)) = 3$ for all $k \in [s - r]$. Extend π^* to all agents by defining $\pi^*(\hat{\jmath}) = \pi(\hat{\jmath}) \cap (V \setminus V_o)$ for every copied agent $\hat{\jmath}$. It is easy to check that π^* is a connected allocation. For each copied agent $\hat{\jmath}$, we have $u_{\hat{\jmath}}(\pi^*(\hat{\jmath})) = u_{\hat{\jmath}}(\pi(\hat{\jmath}))$, since $\hat{\jmath}$ has a valuation of 0 for every item in V_o. Also, for each original agent $i \in N_o$, we have $u_i(\pi^*(i)) \geqslant u_i(\pi(i))$, since i obtains an optimal bundle under π^*. It follows that if π fails (11.2), then π^* is a Pareto-improvement of π, contradicting that π is Pareto-optimal. So π satisfies (11.2). □

Note that the graph constructed in the above proof has pathwidth 2 and diameter 4, so hardness holds even for trees of bounded pathwidth and bounded diameter.

11.4. Pareto-Optimality and EF1 on Paths

In Section 11.3, we were aiming to find *some* Pareto-optimum, and obtained a positive result for the important case where G is a path. Now we aim for an efficient allocation which is also EF1.

When there are no connectivity requirements, it is known that efficiency and fairness are compatible: Caragiannis et al. [2016a] showed that an allocation maximising the *Nash product* of agents' valuations is both Pareto-optimal and EF1. While it is NP-hard to compute the Nash solution, Barman et al. [2018] designed a (pseudo-)polynomial-time algorithm which finds an allocation satisfying these two properties.

In our model, unfortunately, EF1 is incompatible with Pareto-optimality, even when G is a path. The following examples only require binary additive valuations and only two or three agents.

Example 11.5. Consider an instance with two agents a, b and a path with five items $v_1, \dots, v_5$, and binary additive valuations as shown below.

	v_1 –	v_2 –	v_3 –	v_4 –	v_5
a:	1	0	1	1	0
b:	0	1	1	0	1

Write an allocation π as a pair $(\pi(a), \pi(b))$. Then the allocation

- $(\{v_1\}, \{v_2, v_3, v_4, v_5\})$ is not EF1,
- $(\{v_1, v_2\}, \{v_3, v_4, v_5\})$ is Pareto-dominated by $(\{v_1\}, \{v_2, v_3, v_4, v_5\})$,
- $(\{v_1, v_2, v_3\}, \{v_4, v_5\})$ is Pareto-dominated by $(\{v_1, v_2, v_3, v_4\}, \{v_5\})$,
- $(\{v_1, v_2, v_3, v_4\}, \{v_5\})$ is not EF1.

The other allocations also fail Pareto-optimality or EF1, by symmetry. □

The following alternative example shows that Pareto-optimality and EF1 conflict in another restricted setting, where each agent's approval set is an interval.

Example 11.6. Consider an instance with three agents a_1, a_2, and b, and a path with eleven items $v_1, \ldots, v_{11}$, and binary additive valuations as shown below.

	v_1 –	v_2 –	v_3 –	v_4 –	v_5 –	v_6 –	v_7 –	v_8 –	v_9 –	v_{10} –	v_{11}
a_1, a_2:	1	1	1	1	1	1	1	1	1	1	1
b:	0	0	0	1	1	0	0	0	0	0	0

Suppose π is a Pareto-optimal EF1 allocation. Then, for each $i = 1, 2$, because b does not envy a_i up to one good, we have $\{v_4, v_5\} \not\subseteq \pi(a_i)$. Thus, for each $i = 1, 2$, we have either $\pi(a_i) \subseteq \{v_1, \ldots, v_4\}$ (and we say a_i is in group L) or $\pi(a_i) \subseteq \{v_5, \ldots, v_{11}\}$ (and a_i is in group R). Now, a_1 and a_2 are not both in group L, since then there would be a Pareto-improvement by giving the six items $\{v_6, \ldots, v_{11}\}$ to a_1. Also, a_1 and a_2 are not both in group R, since then one of them (say a_1) would receive at most 3 approved items, and there would be a Pareto-improvement by giving items $\{v_1, v_2, v_3\}$ to a_1 and $\{v_6, \ldots, v_{11}\}$ to a_2. Hence, without loss of generality, a_1 is in group L and a_2 is in group R. Since π is Pareto-optimal, we have $\pi(b) \subseteq \{v_4, v_5\}$; if b were to obtain any other items, then we can reallocate these items to a_1 and a_2 to obtain a Pareto-improvement. Thus, a_1 obtains at most four approved items (as $\pi(a_1) \subseteq \{v_1, \ldots, v_4\}$), but a_2 receives at least six approved items (as $\{v_6, \ldots, v_{11}\} \subseteq \pi(a_2)$), so π is not EF1, a contradiction. □

Given that we do not have an existence guarantee, a natural question is whether it is easy to decide whether a given instance admits a Pareto-optimal allocation satisfying EF1. Using the ideas of the reduction of Theorem 11.3 and inserting the above examples as gadgets, one can prove that it is NP-hard to decide this question. See the conference paper for a proof [Igarashi and Peters, 2019]. The obvious complexity upper bound is Σ_2^p; an open problem is whether the problem is complete for this class. A related result of de Keijzer et al. [2009] shows that without connectivity constraints and with additive valuations, it is Σ_2^p-complete to decide whether a Pareto optimal and envy-free allocation exists; see also Bouveret and Lang [2008].

Note that in Examples 11.5 and 11.6, there are two different types of agents' valuations. Invoking Theorem 10.18 from the previous chapter, we can show that a Pareto-optimal EF1 allocation exists on paths for agents with additive valuations that are *identical*, i.e., $u_i(X) = u_i(X)$ for all bundles $X \in \mathcal{C}(V)$ and all $i, j \in N$.

Proposition 11.7. *When G is a path and agents have identical additive valuations, a connected allocation that is Pareto-optimal and satisfies EF1 exists and can be found efficiently.*

Proof. When agents have identical additive valuations, every allocation π has the same utilitarian social welfare $\sum_{i \in N} u_i(\pi(i)) = \sum_{v \in V} u_1(v)$. Hence, every allocation maximises social welfare and is thus Pareto-optimal. Now, by Theorem 10.18, if G is a path then a connected EF1 allocation exists, which, by the above reasoning, is also Pareto-optimal. This allocation can be found efficiently since Theorem 10.18 comes with an efficient algorithm. □

For identical valuations that are not additive, Pareto-optimality and EF1 are again incompatible on a path.

Example 11.8. There are four items a, b, c, d arranged on a path, and two agents with the following identical valuations:

X	$u(X)$	X	$u(X)$	X	$u(X)$
$\{a\}$	2	$\{a, b\}$	2	$\{a, b, c\}$	3
$\{b\}$	2	$\{b, c\}$	3	$\{b, c, d\}$	4
$\{c\}$	2	$\{c, d\}$	3	$\{a, b, c, d\}$	4
$\{d\}$	1				

These valuations are subadditive. Then the allocation

- $\{\{a, b, c, d\}, \varnothing\}$ is not EF1,
- $\{\{a, b, c\}, \{d\}\}$ is not EF1,
- $\{\{a, b\}, \{c, d\}\}$ is Pareto-dominated by $\{\{a\}, \{b, c, d\}\}$,
- $\{\{a\}, \{b, c, d\}\}$ is not EF1. □

11.5. Pareto-Optimality and MMS on Paths

In the previous section, we saw that deciding the existence of an allocation that is Pareto-efficient and satisfies EF1 is computationally hard, even for a path, and saw examples where no such allocation exists. Part of the reason is that envy-freeness notions and Pareto-optimality are not natural companions: it is easy to construct envy-free allocations, which, after a Pareto-improvement, are not envy-free anymore.

An alternative notion of fairness avoids this problem: Pareto-improving upon an MMS allocation preserves the MMS property, because MMS only specifies a lower bound on agents' utilities. In Section 10.3, we saw that if G is a tree, then an MMS allocation always exists (and can be found efficiently). Hence, if G is a tree, there is an allocation that is both Pareto-optimal and MMS: take an MMS allocation, and repeatedly find Pareto-improvements until reaching a Pareto-optimum, which must still satisfy the MMS property.

While existence is guaranteed, it is unclear whether we can find an allocation satisfying both properties in polynomial time. Certainly, by the negative result of Theorem 11.4, this is not possible when G is an arbitrary tree. What about the case when G is a path? In the conference paper, we show that, unless P = NP, there is no polynomial-time algorithm which finds a Pareto-optimal MMS allocation when G is a path, even if valuations are binary and additive [Igarashi and Peters, 2019]. The proof also implies that we cannot in polynomial time find a Pareto-optimal allocation that is α-MMS, for fixed $\alpha > 0$, where an allocation π is said to be α-MMS if $u_i(\pi(i)) \geqslant \alpha \cdot \text{MMS}_i$ for all $i \in N$.

11.6. Conclusion

In this chapter, we have studied the computational complexity of finding Pareto-efficient outcomes, in the natural setting where we need to allocate indivisible items into connected bundles. We showed that although finding a Pareto-optimal allocation is easy for some topologies, this does not extend to general trees. Further, we proved that when imposing additional fairness requirements, finding a Pareto-optimum becomes NP-hard even when the underlying item graph is a path. We have also seen that a Pareto-optimal EF1 allocation may not exist with the contiguity requirement while such an allocation always exists when these requirements are ignored.

Thanks to Erel Segal-Halevi, Warut Suksompong, and Rohit Vaish for useful discussions.

12. Strategyproofness and EF1

In this brief final chapter, we consider the existence of strategyproof allocation rules for items arranged on a path, and find that no such rule exists that also guarantees EF1. We connect this result with similar results from cake-cutting.

12.1. Introduction

In the previous chapters, we have mostly been interested in the *existence* of high-quality allocations of items. For practical uses, there remains the problem of *selecting* among those allocations that satisfy our desired axioms such as EF1, that is, the problem of designing a mechanism that, given a valuation profile, returns an allocation. The cut-and-choose procedure for $n = 2$ agents and the Stromquist-inspired moving knife procedure for $n = 3$ agents are such mechanisms. Once we have a mechanism, it makes sense to not just study intra-profile axioms such as EF1 (which only concern the quality of the output), but to also consider inter-profile axioms that connect the outputs at different profiles. The inter-profile condition we have studied most often in this thesis is strategyproofness, which can easily be defined for our setting. Unfortunately, the cut-and-choose procedure as well as the Stromquist procedure are both manipulable. For example, suppose that $n = 2$, and there are five items arranged on a path. Alice has valuation 1–1–1–1–1, so her lumpy tie is the middle item. Bob has valuation 1–1–0–0–0, so he picks the bundle to the left of Alice's lumpy tie, and Alice obtains the rest, namely three items. However, if Alice reports her valuation as 1–1–0–0–0 (the same as Bob), then her lumpy tie would be the second item; Bob still chooses the left bundle, and now Alice obtains the right four items. Thus, Alice has obtained a more valuable bundle by misreporting her valuations.

In the setting without connectivity constraints, Amanatidis et al. [2017] characterised all strategyproof allocation mechanisms when there are $n = 2$ agents with additive valuations. They then proved that no mechanism in their class guarantees EF1 [Amanatidis et al., 2017, Sec. 4.2] for $m \geqslant 5$ items, and hence no strategyproof EF1 mechanism exists. Now suppose we had a mechanism that satisfies connectivity constraints and is also strategyproof and EF1. The exact same mechanism could be used in the model without connectivity constraints and continues to be strategyproof and EF1, contradicting the result of Amanatidis et al. [2017]. Thus, for example, there does not exist a strategyproof mechanism that allocates 5 or more items arranged on a path to players while guaranteeing EF1. This proof uses rather heavy machinery, requiring a full characterisation of all strategyproof mechanisms. Here, we give a simple and direct argument.

12.2. A Simple Impossibility

Our result will work even on the restricted domain of additive valuations that are binary and where each agent approves items that form an interval of the path G. For $n = 2$ and $m = 5$, there are only $16^2 = 256$ valuation profiles with this restricted domain. For each valuation profile, there are only 12 possible connected allocations (that are *complete* in the sense of allocating all items to the agents). It is easy to write down a propositional formula encoding a mechanism that is strategyproof and EF1, and this formula is unsatisfiable and admits an MUS referring to only five profiles.

Theorem 12.1. *Suppose there are $n = 2$ agents, and $m \geqslant 5$ items arranged on a graph G that is a path. Then there exists no strategyproof mechanism which always returns a complete EF1 allocation connected on G. Impossibility holds even for binary additive valuations, where every agent approves an interval of items.*

Proof. We start by considering the case of $m = 5$ items with $V = \{a, b, c, d, e\}$ arranged on a path in the printed order. Suppose there exists a mechanism that is strategyproof, returns a complete and connected allocation, and guarantees EF1.

Consider the following profile P_1:

$$u_1 : 1\text{–}1\text{–}0\text{–}0\text{–}0 \qquad u_2 : 1\text{–}1\text{–}0\text{–}0\text{–}0$$

By completeness and EF1, each agent needs to obtain at least one of items a and b. Thus, by completeness and connectedness, one agent obtains bundle $\{a\}$ and the other $\{b, c, d, e\}$. Without loss of generality, suppose that agent 1 obtains $\{a\}$.

Consider the following profile P_2:

$$u_1 : 1\text{–}1\text{–}0\text{–}0\text{–}0 \qquad u_2 : 0\text{–}1\text{–}1\text{–}1\text{–}1$$

In this profile, agent 2 needs to obtain items b, c, d, and e, since otherwise agent 2 can manipulate and report valuations 1–1–0–0–0 which brings us to profile P_1 where agent 2 obtains all these items. On the other hand, agent 2 cannot also obtain item a by EF1. Hence, agent 1 obtains bundle $\{a\}$.

Consider the following profile P_3:

$$u_1 : 1\text{–}1\text{–}1\text{–}1\text{–}0 \qquad u_2 : 0\text{–}1\text{–}1\text{–}1\text{–}1$$

In this profile, by EF1, agent 1 either obtains bundle $\{a, b\}$ or $\{a, b, c\}$. Thus, in profile P_2, agent 1 has an incentive to manipulate towards P_3, contradicting strategyproofness.

To extend the argument to the case $m \geqslant 5$, we can append additional items on the right-hand end of the path and stipulate that neither agent values these items. □

12.3. Importing Impossibilities from Cake-Cutting

There is another way to obtain a result along the lines of Theorem 12.1, by reduction from divisible cake-cutting. Fix some $\varepsilon > 0$, and suppose we had a mechanism for allocating a path of M items among n agents while being strategyproof and EF1. Then we can use this mechanism as a mechanism for cake-cutting: Given continuous agent valuations over the interval $[0, 1]$, approximate these by additive valuations over the path of M items and run the mechanism on this instance. For sufficiently large M, the resulting mechanism for cake-cutting will be ε-strategyproof (in the sense that a misreport can increase utility by at most ε) and ε-envy-free (in the sense that envy is bounded by ε). However, the literature on cake-cutting contains impossibilities about strategyproofness and envy-freeness [Bei et al., 2017, Theorem 1, Bei et al., 2018, Theorem 3], and the proofs also establish impossibility for the ε-versions of these properties. Hence, for large enough M, no strategyproof EF1 mechanism for the indivisible setting can exist. By using the result of Bei et al. [2017], we can obtain an impossibility also for $n > 2$ and for a weakening of the requirement that allocations be complete, but requiring non-binary additive valuations. By using the result of Bei et al. [2018], we can obtain an impossibility for $n = 2$ and even for binary additive valuations where each agent approves an interval of items beginning with the left-most item.

Bibliography

S. Airiau, H. Aziz, I. Caragiannis, J. Kruger, J. Lang, and D. Peters. Portioning using ordinal preferences: Fairness and efficiency. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 11–17, 2019. [→ p. xiii]

N. Alon, M. Feldman, A. D. Procaccia, and M. Tennenholtz. Strategyproof approximation of the minimax on networks. *Mathematics of Operations Research*, 35(3):513–526, 2010a. Extended version: arXiv:0907.2049. [→ p. 39, 122]

N. Alon, M. Feldman, A. D. Procaccia, and M. Tennenholtz. Walking in circles. *Discrete Mathematics*, 310(23):3432–3435, 2010b. [→ p. 122]

G. Amanatidis, G. Birmpas, G. Christodoulou, and E. Markakis. Truthful allocation mechanisms without payments: Characterization and implications on fairness. In *Proceedings of the 18th ACM Conference on Economics and Computation (ACM-EC)*, pages 545–562, 2017. [→ p. 167]

K. Arrow, A. Sen, and K. Suzumura, editors. *Handbook of Social Choice and Welfare, Volume 1*. Elsevier, 2002. [→ p. 89]

Y. Aumann and Y. Dombb. The efficiency of fair division with connected pieces. *ACM Transactions on Economics and Computation*, 3:23:1–23:16, 2015. [→ p. 157]

Y. Aumann, Y. Dombb, and A. Hassidim. Computing socially-efficient cake divisions. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 343–350, 2013. [→ p. 157]

M. Ayadi, N. Ben Amor, J. Lang, and D. Peters. Single transferable vote: Incomplete knowledge and communication issues. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1288–1296, 2019. [→ p. xvi]

H. Aziz and P. Stursberg. A generalization of probabilistic serial to randomized social choice. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI)*, pages 559–565, 2014. [→ p. 63, 64]

H. Aziz, F. Brandt, and P. Harrenstein. Pareto optimality in coalition formation. *Games and Economic Behavior*, 82:562–581, 2013. [→ p. 158]

H. Aziz, F. Brandl, and F. Brandt. Universal Pareto dominance and welfare for plausible utility functions. In *Proceedings of the 15th ACM Conference on Economics and Computation (ACM-EC)*, pages 331–332, 2014. [→ p. 57]

H. Aziz, S. Gaspers, J. Gudmundsson, S. Mackenzie, N. Mattei, and T. Walsh. Computational aspects of multi-winner approval voting. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 107–115, 2015. [→ p. 87]

H. Aziz, P. Biró, J. Lang, J. Lesca, and J. Monnot. Optimal reallocation under additive and ordinal preferences. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 402–410, 2016. [→ p. 158]

H. Aziz, M. Brill, V. Conitzer, E. Elkind, R. Freeman, and T. Walsh. Justified representation in approval-based committee voting. *Social Choice and Welfare*, 48(2):461–485, 2017. [→ p. 76, 80, 81]

H. Aziz, B. E. Lee, and N. Talmon. Proportionally representative participatory budgeting: Axioms and algorithms. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 23–31, 2018a. [→ p. 39, 77]

H. Aziz, P. Luo, and C. Rizkallah. Rank maximal equal contribution: a probabilistic social choice function. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, pages 910–916, 2018b. [→ p. 63]

H. Aziz, A. Bogomolnaia, and H. Moulin. Fair mixing: the case of dichotomous preferences. In *Proceedings of the 20th ACM Conference on Economics and Computation (ACM-EC)*, pages 753–781, 2019a. [→ p. 39, 56, 57, 59, 63]

H. Aziz, F. Brandl, F. Brandt, P. Harrenstein, M. Olsen, and D. Peters. Fractional hedonic games. *ACM Transactions on Economics and Computation*, 7(2), 2019b. [→ p. xv]

H. Aziz, I. Caragiannis, and A. Igarashi. Fair allocation of combinations of indivisible goods and chores. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 53–59, 2019c. [→ p. 155]

G. Bachmeier, F. Brandt, C. Geist, P. Harrenstein, K. Kardel, D. Peters, and H. G. Seedig. k-majority digraphs and the hardness of voting with a constant number of voters. *Journal of Computer and System Sciences*, 105:130–157, 2019. [→ p. xvi]

M. Balinski and H. P. Young. *Fair Representation: Meeting the Ideal of One Man, One Vote*. Yale University Press, 1982. (2nd Edition, Brookings Institution Press, 2001). [→ p. 75]

R. B. Bapat. A constructive proof of a permutation-based generalization of Sperner's lemma. *Mathematical Programming*, 44(1):113–120, 1989. [→ p. 149]

J. B. Barbanel and S. J. Brams. Cake division with minimal cuts: envy-free procedures for three persons, four persons, and beyond. *Mathematical Social Sciences*, 48(3):251–269, 2004. [→ p. 134]

S. Barberà. Majority and positional voting in a probabilistic framework. *Review of Economic Studies*, 46(2):379–389, 1979. [→ p. 70]

S. Barberà and M. Jackson. A characterization of strategy-proof social choice functions for economies with pure public goods. *Social Choice and Welfare*, 11(3):241–252, 1994. [→ p. 38, 39]

S. Barberà, F. Gul, and E. Stacchetti. Generalized median voter schemes and committees. *Journal of Economic Theory*, 61(2):262–289, 1993. [→ p. 39]

S. Barberà, J. Masso, and A. Neme. Voting under constraints. *Journal of Economic Theory*, 76 (2):298–321, 1997. [→ p. 39]

S. Barman, S. K. Krishnamurthy, and R. Vaish. Finding fair and efficient allocations. In *Proceedings of the 19th ACM Conference on Economics and Computation (ACM-EC)*, pages 557–574, 2018. [→ p. 158, 162]

J. Bartholdi, III and M. Trick. Stable matching with preferences derived from a psychological model. *Operations Research Letters*, 5(4):165–169, 1986. [→ p. 91, 120]

J. Bartholdi, III, C. A. Tovey, and M. A. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 6(3):157–165, 1989. [→ p. 89, 99, 122, 123]

D. Baumeister, S. Dennisen, and L. Rey. Winner determination and manipulation in minisum and minimax committee elections. In *Proceedings of the 3rd International Conference on Algorithmic Decision Theory (ADT)*, pages 469–485. Springer, 2015. [→ p. 87]

D. Baumeister, S. Bouveret, J. Lang, N.-T. Nguyen, T. T. Nguyen, J. Rothe, and A. Saffidine. Positional scoring-based allocation of indivisible goods. *Autonomous Agents and Multi-Agent Systems*, 31(3):628–655, 2017. [→ p. 63]

J. Behrens, A. Kistner, A. Nitsche, and B. Swierczek. *The Principles of LiquidFeedback*. 2014. [→ p. 81]

X. Bei, N. Chen, X. Hua, B. Tao, and E. Yang. Optimal proportional cake cutting with connected pieces. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI)*, pages 1263–1269, 2012. [→ p. 157]

X. Bei, N. Chen, G. Huzhang, B. Tao, and J. Wu. Cake cutting: envy and truth. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3625–3631, 2017. [→ p. 168]

X. Bei, G. Huzhang, and W. Suksompong. Truthful fair division without free disposal. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 63–69, 2018. [→ p. 168]

A. Belov and J. Marques-Silva. MUSer2: An efficient MUS extractor. *Journal on Satisfiability, Boolean Modeling and Computation*, 8:123–128, 2012. [→ p. 18]

G. Benade, S. Nath, A. D. Procaccia, and N. Shah. Preference elicitation for participatory budgeting. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 376–382, 2017. [→ p. 39, 77]

N. Betzler, A. Slinko, and J. Uhlmann. On the computation of fully proportional representation. *Journal of Artificial Intelligence Research*, 47(1):475–519, 2013. [→ p. 89, 90, 93, 95, 97, 101, 115, 117, 125]

A. Biere. Lingeling, Plingeling and Treengeling entering the SAT competition 2013. In *Proceedings of the SAT Competition 2013*, pages 51–52, 2013. [→ p. 18]

V. Bilò, I. Caragiannis, M. Flammini, A. Igarashi, G. Monaco, D. Peters, C. Vinci, and W. S. Zwicker. Almost envy-free allocations with connected bundles. In *Proceedings of the 10th Innovations in Theoretical Computer Science Conference (ITCS 2019)*, pages 14:1–14:21, 2018. Extended version: arXiv:1808.09406. [→ p. xiv, 154]

A. Biswas and S. Barman. Fair division under cardinality constraints. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 91–97, 2018. [→ p. 153]

D. Black. On the rationale of group decision-making. *The Journal of Political Economy*, 56(1): 23–34, 1948. [→ p. 90, 117]

H. L. Bodlaender. A tourist guide through treewidth. *Acta Cybernetica*, 11(1–2):1–21, 1994. [→ p. 94]

A. Bogomolnaia, H. Moulin, and R. Stong. Collective choice under dichotomous preferences. Extended working paper version, 2002. [→ p. 55]

A. Bogomolnaia, H. Moulin, and R. Stong. Collective choice under dichotomous preferences. *Journal of Economic Theory*, 122(2):165–184, 2005. [→ p. 5, 39, 55, 57, 58, 68]

A. Bogomolnaia, H. Moulin, F. Sandomirskyi, and E. Yanovskaya. Dividing goods and bads under additive utilities. 2016. arXiv:1610.03745. [→ p. 155]

A. Bogomolnaia, H. Moulin, F. Sandomirskyi, and E. Yanovskaya. Competitive division of a mixed manna. *Econometrica*, 85(6):1847–1871, 2017. [→ p. 155]

J. A. Bondy and U. S. R. Murty. *Graph Theory*. Springer, 1st edition, 2008. [→ p. 140, 141]

K. S. Booth and G. S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. *Journal of Computer and System Sciences*, 13(3): 335–379, 1976. [→ p. 120]

K. C. Border and J. S. Jordan. Straightforward elections, unanimity and phantom voters. *Review of Economic Studies*, 50(1):153–170, 1983. [→ p. 38, 39]

S. Bouveret and J. Lang. Efficiency and envy-freeness in fair division of indivisible goods: logical representation and complexity. *Journal of Artificial Intelligence Research*, 32(1):525–564, 2008. [→ p. 163]

S. Bouveret and M. Lemaître. Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Autonomous Agents and Multi-Agent Systems*, 30(2):259–290, 2016. [→ p. 134]

S. Bouveret, K. Cechlárová, E. Elkind, A. Igarashi, and D. Peters. Fair division of a graph. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 135–141, 2017. [→ p. xiv, 138]

S. J. Brams and P. C. Fishburn. *Approval Voting*. Birkhäuser, 1983. [→ p. 56]

S. J. Brams and D. L. King. Efficient fair division: Help the worst off or avoid envy? *Rationality and Society*, 17(4):387–421, 2005. [→ p. 63]

S. J. Brams and A. D. Taylor. *Fair division: from cake-cutting to dispute resolution*. Cambridge University Press, 1996. [→ p. 133, 138]

S. J. Brams, A. D. Taylor, and W. S. Zwicker. A moving-knife solution to the four-person envy-free cake-division problem. *Proceedings of the American Mathematical Society*, 125(2): 547–554, 1997. [→ p. 134]

F. Brandl and D. Peters. An axiomatic characterization of the Borda mean rule. *Social Choice and Welfare*, 52(4):685–707, 2019. [→ p. xvi]

F. Brandl, F. Brandt, C. Geist, and J. Hofbauer. Strategic abstention based on preference extensions: Positive results and computer-generated impossibilities. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 18–24, 2015a. [→ p. 13, 14]

F. Brandl, F. Brandt, and J. Hofbauer. Incentives for participation and abstention in probabilistic social choice. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1411–1419, 2015b. [→ p. 14]

F. Brandl, F. Brandt, and H. G. Seedig. Consistent probabilistic social choice. *Econometrica*, 84 (5):1839–1880, 2016. [→ p. 63]

F. Brandl, F. Brandt, M. Eberl, and C. Geist. Proving the incompatibility of efficiency and strategyproofness via SMT solving. *Journal of the ACM*, 65(2), 2018. [→ p. 57]

F. Brandl, F. Brandt, and J. Hofbauer. Welfare maximization entices participation. *Games and Economic Behavior*, 14:308–314, 2019a. [→ p. 14]

F. Brandl, F. Brandt, D. Peters, C. Stricker, and W. Suksompong. Donor coordination: Collective distribution of individual contributions. 2019b. Working paper. [→ p. xiii, 56]

F. Brandt. Set-monotonicity implies Kelly-strategyproofness. *Social Choice and Welfare*, 45(4): 793–804, 2015. [→ p. 14]

F. Brandt. Rolling the dice: Recent results in probabilistic social choice. In U. Endriss, editor, *Trends in Computational Social Choice*, chapter 1, pages 3–26. AI Access, 2017. [→ p. 78]

F. Brandt. Collective choice lotteries: Dealing with randomization in economic design. In J.-F. Laslier, H. Moulin, R. Sanver, and W. S. Zwicker, editors, *The Future of Economic Design*. Springer-Verlag, 2019. Forthcoming. [→ p. 63]

F. Brandt and C. Geist. Finding strategyproof social choice functions via SAT solving. *Journal of Artificial Intelligence Research*, 55:565–602, 2016. [→ p. 13, 14]

F. Brandt, M. Brill, E. Hemaspaandra, and L. A. Hemaspaandra. Bypassing combinatorial protections: Polynomial-time algorithms for single-peaked electorates. *Journal of Artificial Intelligence Research*, 53:439–496, 2015. [→ p. 90, 117, 123]

F. Brandt, M. Brill, and P. Harrenstein. Tournament solutions. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 3. Cambridge University Press, 2016a. [→ p. 123]

F. Brandt, C. Geist, and D. Peters. Optimal bounds for the no-show paradox via SAT solving. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 314–322, 2016b. [→ p. xiii]

F. Brandt, C. Geist, and D. Peters. Replication data for: Optimal bounds for the no-show paradox via SAT solving. Harvard Dataverse. http://dx.doi.org/10.7910/DVN/TGIQB7, 2016c. [→ p. 21]

F. Brandt, C. Geist, and D. Peters. Optimal bounds for the no-show paradox via SAT solving. *Mathematical Social Sciences*, 90:18–27, 2017. Special Issue in Honor of Hervé Moulin. [→ p. xiii, 22]

F. Brandt, J. Hofbauer, and M. Strobel. Exploring the no-show paradox for Condorcet extensions using Ehrhart theory and computer simulations. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 520–528, 2019. [→ p. 13]

R. Bredereck, J. Chen, and G. J. Woeginger. Are there any nicely structured preference profiles nearby? *Mathematical Social Sciences*, 79:61–73, 2016. [→ p. 116]

M. Brill, R. Freeman, S. Janson, and M. Lackner. Phragmén's voting methods and justified representation. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 406–413, 2017. [→ p. 76, 81]

M. Brill, J.-F. Laslier, and P. Skowron. Multiwinner approval rules as apportionment methods. *Journal of Theoretical Politics*, 30(3):358–382, 2018. [→ p. 80, 81]

E. Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6):1061–1103, 2011. [→ p. 7, 134, 136]

D. E. Campbell and J. S. Kelly. A strategy-proofness characterization of majority rule. *Economic Theory*, 22(3):557–568, 2003. [→ p. 30, 33]

D. E. Campbell and J. S. Kelly. Strategy-proofness and weighted voting. *Mathematical Social Sciences*, 60(1):15–23, 2010. [→ p. 14]

D. E. Campbell and J. S. Kelly. Anonymous, neutral, and strategy-proof rules on the Condorcet domain. *Economics Letters*, 128:79–82, 2015. [→ p. 32, 33]

D. E. Campbell and J. S. Kelly. Correction to "A strategy-proofness characterization of majority rule". *Economic Theory Bulletin*, 4(1):121–124, 2016. [→ p. 30]

D. E. Campbell, J. Graver, and J. S. Kelly. There are more strategy-proof procedures than you think. *Mathematical Social Sciences*, 64(3):263–265, 2012. [→ p. 14]

I. Caragiannis, E. Elkind, M. Szegedy, and L. Yu. Mechanism design: from partial to probabilistic verification. In *Proceedings of the 13th ACM Conference on Electronic Commerce (ACM EC)*, pages 266–283, 2012. [→ p. 34]

I. Caragiannis, D. Kurokawa, H. Moulin, A. D. Procaccia, N. Shah, and J. Wang. The Unreasonable Fairness of Maximum Nash Welfare. In *Proceedings of the 17th ACM Conference on Economics and Computation (ACM-EC)*, pages 305–322, 2016a. [→ p. 135, 136, 162]

I. Caragiannis, A. D. Procaccia, and N. Shah. Truthful univariate estimators. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pages 127–135, 2016b. [→ p. 41]

J. R. Chamberlin and P. N. Courant. Representative deliberations and representative decisions: Proportional representation and the Borda rule. *The American Political Science Review*, 77 (3):718–733, 1983. [→ p. 81, 89, 125]

R. T. Clemen. Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5:559–583, 1989. [→ p. 39]

V. Conitzer and T. Sandholm. Complexity of mechanism design. In *Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 103–110, 2002. [→ p. 14]

V. Conitzer and T. Sandholm. Applications of automated mechanism design. In *Proceedings of the UAI workshop on Bayesian Modeling Applications*, 2003. [→ p. 14]

V. Conitzer and T. Sandholm. Expressive markets for donating to charities. *Artificial Intelligence*, 175(7-8):1251–1271, 2011. [→ p. 14]

V. Conitzer, J. Derryberry, and T. Sandholm. Combinatorial auctions with structured item graphs. In *Proceedings of the 19th AAAI Conference on Artificial Intelligence (AAAI)*, pages 212–218, 2004. [→ p. 157]

V. Conitzer, M. Brill, and R. Freeman. Crowdsourcing societal tradeoffs. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS) Blue Sky Ideas track*, pages 1213–1217, 2015. [→ p. 39]

V. Conitzer, R. Freeman, M. Brill, and Y. Li. Rules for choosing societal tradeoffs. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pages 460–467, 2016. [→ p. 39]

D. Cornaz, L. Galand, and O. Spanjaard. Bounded single-peaked width and proportional representation. In *Proceedings of the 20th European Conference on Artificial Intelligence (ECAI)*, pages 270–275, 2012. [→ p. 93, 116]

A. Darmann and J. Schauer. Maximizing Nash product social welfare in allocating indivisible goods. *European Journal of Operational Research*, 247(2):548–559, 2015. [→ p. 63]

B. de Keijzer, S. Bouveret, T. Klos, and Y. Zhang. On the complexity of efficiency and envy-freeness in fair division of indivisible goods with additive preferences. In *Proceedings of the 1st International Conference on Algorithmic Decision Theory (ADT)*, pages 98–110, 2009. [→ p. 163]

G. Demange. Single-peaked orders on a tree. *Mathematical Social Sciences*, 3(4):389–396, 1982. [→ p. 91, 94, 117]

X. Deng, Q. Qi, and A. Saberi. Algorithmic solutions for envy-free cake cutting. *Operations Research*, 60(6):1461–1476, 2012. [→ p. 149, 150, 154]

J.-P. Doignon and J.-C. Falmagne. A polynomial time algorithm for unidimensional unfolding representations. *Journal of Algorithms*, 16(2):218–233, 1994. [→ p. 91, 113]

M. Dom. Algorithmic aspects of the consecutive-ones property. *Bulletin of the European Association for Theoretical Computer Science*, 98:27–59, 2009. [→ p. 129]

L. E. Dubins and E. H. Spanier. How to cut a cake fairly. *The American Mathematical Monthly*, 68(1):1–17, 1961. [→ p. 133]

C. Duddy. Condorcet's principle and the strong no-show paradoxes. *Theory and Decision*, 77(2): 275–285, 2014a. [→ p. 13, 14]

C. Duddy. Electing a representative committee by approval ballot: An impossibility result. *Economics Letters*, 124(1):14–16, 2014b. [→ p. 77, 86]

C. Duddy. Fair sharing under dichotomous preferences. *Mathematical Social Sciences*, 73:1–5, 2015. [→ p. 56, 57]

E. Eisenberg and D. Gale. Consensus of subjective probabilities: The pari-mutuel method. *The Annals of Mathematical Statistics*, 30(1):165–168, 1959. [→ p. 46, 47]

E. Elkind and M. Lackner. Structure in dichotomous preferences. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2019–2025, 2015. [→ p. 87]

E. Elkind, M. Lackner, and D. Peters. Preference restrictions in computational social choice: Recent progress. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4062–4065, 2016. [→ p. xvi, 117]

E. Elkind, P. Faliszewski, P. Skowron, and A. Slinko. Properties of multiwinner voting rules. *Social Choice and Welfare*, 48(3):599–632, 2017a. [→ p. 81, 86]

E. Elkind, M. Lackner, and D. Peters. Structured preferences. In U. Endriss, editor, *Trends in Computational Social Choice*, chapter 10. 2017b. [→ p. xvi, 87, 117]

G. Erdelyi, M. Lackner, and A. Pfandler. Computational aspects of nearly single-peaked electorates. *Journal of Artificial Intelligence Research*, 58:297–337, 2017. [→ p. 116]

B. Escoffier, J. Lang, and M. Öztürk. Single-peaked consistency and its complexity. In *Proceedings of the 18th European Conference on Artificial Intelligence (ECAI)*, pages 366–370, 2008. [→ p. 91, 113]

S. Even and R. E. Tarjan. Computing an *st*-numbering. *Theoretical Computer Science*, 2(3): 339–344, 1976. [→ p. 141]

B. Fain, A. Goel, and K. Munagala. The core of the participatory budgeting problem. In *Proceedings of the 12th International Workshop on Internet and Network Economics (WINE)*, pages 384–399, 2016. [→ p. 39, 63, 69]

B. Fain, K. Munagala, and N. Shah. Fair allocation of indivisible public goods. In *Proceedings of the 19th ACM Conference on Economics and Computation (ACM-EC)*, pages 575–592, 2018. [→ p. 77]

P. Faliszewski and N. Talmon. A framework for approval-based budgeting methods. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pages 2181–2188, 2019. [→ p. 77]

P. Faliszewski, E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. The shield that never was: Societies with single-peaked preferences are more open to manipulation and control. *Information and Computation*, 209(2):89–107, 2011. [→ p. 87]

P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra. The complexity of manipulative attacks in nearly single-peaked electorates. *Artificial Intelligence*, 207:69–99, 2014. [→ p. 116]

P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. Committee scoring rules: Axiomatic classification and hierarchy. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 250–256, 2016. [→ p. 86]

P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. Multiwinner voting: A new challenge for social choice theory. In U. Endriss, editor, *Trends in Computational Social Choice*, chapter 2. 2017a. [→ p. 75, 78, 81, 125]

P. Faliszewski, P. Skowron, and N. Talmon. Bribery as a measure of candidate success: Complexity results for approval-based multiwinner rules. In *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 6–14, 2017b. [→ p. 87]

P. Faliszewski, M. Lackner, D. Peters, and N. Talmon. Effective heuristics for committee scoring rules. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, pages 1023–1030, 2018a. [→ p. xv]

P. Faliszewski, S. Szufa, and N. Talmon. Optimization-based voting rule design: The closer to utopia the better. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 32–40, 2018b. [→ p. 130]

F. Fischer, O. Hudry, and R. Niedermeier. Weighted tournament solutions. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 4. Cambridge University Press, 2016. [→ p. 123]

P. C. Fishburn. Condorcet social choice functions. *SIAM Journal on Applied Mathematics*, 33 (3):469–489, 1977. [→ p. 21]

P. C. Fishburn and S. J. Brams. Paradoxes of preferential voting. *Mathematics Magazine*, 56(4): 207–214, 1983. [→ p. 11, 12, 13]

T. Fluschnik, P. Skowron, M. Triphaus, and K. Wilker. Fair knapsack. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pages 1941–1948, 2019. [→ p. 77, 130]

D. K. Foley. Lindahl's solution and the core of an economy with public goods. *Econometrica*, 38 (1):66–72, 1970. [→ p. 69]

A. Fréchette, N. Newman, and K. Leyton-Brown. Solving the station repacking problem. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pages 702–709, 2016. [→ p. 14]

R. Freeman, D. M. Pennock, D. Peters, and J. Wortman Vaughan. Truthful aggregation of budget proposals. In *Proceedings of the 20th ACM Conference on Economics and Computation (ACM-EC)*, pages 751–752, 2019. arXiv:1905.00457. [→ p. xiii]

S. French. Group consensus probability distributions: A critical survey. *Bayesian Statistics*, 2: 183–202, 1985. [→ p. 39]

P. Gärdenfors. On definitions of manipulation of social choice functions. In J. J. Laffont, editor, *Aggregation and Revelation of Preferences*. North-Holland, 1979. [→ p. 78]

M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979. [→ p. 95]

N. Garg, A. Goel, and B. Plaut. Markets for public decision-making. In *Proceedings of the 14th International Workshop on Internet and Network Economics (WINE)*, page 445, 2018. arXiv:1807.10836. [→ p. 47]

C. Geist and U. Endriss. Automated search for impossibility theorems in social choice theory: Ranking sets of objects. *Journal of Artificial Intelligence Research*, 40:143–174, 2011. [→ p. 13, 14]

C. Geist and D. Peters. Computer-aided methods for social choice theory. In U. Endriss, editor, *Trends in Computational Social Choice*, chapter 13, pages 249–267. AI Access, 2017. [→ p. xvi, 14]

C. Genest and J. Zidek. Combining probability distributions: A critique and an annotated bibliography. *Statistical Science*, 1(1):114–148, 1986. [→ p. 39]

A. Gibbard. Manipulation of voting schemes: A general result. *Econometrica*, 41(4):587–601, 1973. [→ p. 29, 34]

A. Gibbard. Manipulation of schemes that mix voting with chance. *Econometrica*, 45(3):665–681, 1977. [→ p. 63, 70]

A. Goel, A. K. Krishnaswamy, S. Sakshuwong, and T. Aitamurto. Knapsack voting for participatory budgeting. *ACM Transactions on Economics and Computation*, 7(2):8:1–8:27, 2019. [→ p. 37, 38, 39, 52]

A. Gulati, G. Shanmuganathan, X. Zhang, and P. Varman. Demand based hierarchical QoS using storage resource pools. In *USENIX Annual Technical Conference*, pages 1–13, 2012. [→ p. 51]

J. Guo and R. Niedermeier. Exact algorithms and applications for tree-like weighted set cover. *Journal of Discrete Algorithms*, 4(4):608–622, 2006. [→ p. 94]

E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Exact analysis of Dodgson elections: Lewis Carroll's 1876 voting system is complete for parallel access to NP. *Journal of ACM*, 44(6): 806–825, 1997. [→ p. 89]

E. Hemaspaandra, H. Spakowski, and J. Vogel. The complexity of Kemeny elections. *Theoretical Computer Science*, 349(3):382–391, 2005. [→ p. 89, 122]

M. J. H. Heule, O. Kullmann, and V. W. Marek. Solving and verifying the Boolean Pythagorean triples problem via cube-and-conquer. In *Proceedings of the 19th International Conference on Theory and Applications of Satisfiability Testing*, pages 228–245, 2016. [→ p. 13]

A. J. Hoffman and J. B. Kruskal. Integral boundary points of convex polyhedra. In H. W. Kuhn and A. W. Tucker, editors, *Linear Inequalities and Related Systems*, pages 223–246. Princeton University Press, 1956. [→ p. 125]

R. Holzman. To vote or not to vote: What is the quota? *Discrete Applied Mathematics*, 22(2): 133–141, 1988. [→ p. 13]

J. Hopcroft and R. Tarjan. Algorithm 447: efficient algorithms for graph manipulation. *Communications of the ACM*, 16(6):372–378, 1973. [→ p. 141]

A. Hylland. Proportional representation without party lists. In R. Malnes and A. Underdal, editors, *Rationality and institutions : essays in honour of Knut Midgaard on the occasion of his 60th birthday*, pages 126–153. Universitetsforlaget, 1992. [→ p. 79]

A. Igarashi. Fairness and stability in structured environments, 2018. DPhil thesis, University of Oxford. [→ p. xiv]

A. Igarashi and D. Peters. Pareto-optimal allocation of indivisible goods with connectivity constraints. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pages 2045–2052, 2019. Extended version: arXiv:1811.04872. [→ p. xiv, 158, 163, 164]

A. Igarashi, D. Peters, and E. Elkind. Group activity selection on social networks. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 565–571, 2017. [→ p. xv]

M. Intriligator. A probabilistic model of social choice. *The Review of Economic Studies*, 40(4): 553–560, 1973. [→ p. 39]

F. Jaeckle, D. Peters, and E. Elkind. On recognising nearly single-crossing preferences. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, pages 1079–1086, 2018. [→ p. xv]

S. Janson. Phragmén's and Thiele's election methods. 2016. arXiv:1611.08826. [→ p. 76]

J. L. Jimeno, J. Pérez, and E. García. An extension of the Moulin No Show Paradox for voting correspondences. *Social Choice and Welfare*, 33(3):343–459, 2009. [→ p. 13]

K. Kardel. Participation and group-participation in social choice theory. Master's thesis, Technische Universität München, 2014. [→ p. 12, 13]

D. M. Kilgour. Approval balloting for multi-winner elections. In J.-F. Laslier and R. Sanver, editors, *Handbook on Approval Voting*, pages 105–124. Springer, 2010. [→ p. 75]

K. H. Kim and F. W. Roush. Special domains and nonmanipulability. *Mathematical Social Sciences*, 1(1):85–92, 1980. [→ p. 122]

B. Konev and A. Lisitsa. A SAT attack on the Erdős discrepancy conjecture. In *Proceedings of the 17th International Conference on Theory and Applications of Satisfiability Testing*, pages 219–226. 2014. [→ p. 13]

B. Korte and J. Vygen. *Combinatorial Optimization - Theory and Algorithms*. Springer, 2006. [→ p. 160]

G. Kreweras. Aggregation of preference orderings. In *Mathematics and Social Sciences I: Proceedings of the seminars of Menthon-Saint-Bernard, France (1–27 July 1960) and of Gösing, Austria (3–27 July 1962)*, pages 73–79, 1965. [→ p. 63]

H. W. Kuhn. Some combinatorial lemmas in topology. *IBM Journal of Research and Development*, 4(5):518–524, 1960. [→ p. 150]

D. Kurokawa, A. D. Procaccia, and J. Wang. Fair enough: Guaranteeing approximate maximin shares. *Journal of the ACM*, 65(2):8, 2018. [→ p. 134]

M. Lackner. Incomplete preferences in single-peaked electorates. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI)*, pages 742–748, 2014. [→ p. 120, 121]

M. Lackner and P. Skowron. Consistent approval-based multi-winner rules. In *Proceedings of the 19th ACM Conference on Economics and Computation (ACM-EC)*, pages 47–48, 2018. [→ p. 81, 82, 86]

F. Lakhani, D. Peters, and E. Elkind. Correlating preferences and attributes: Nearly single-crossing profiles. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 414–420, 2019. [→ p. xv]

A. Lempel, S. Even, and I. Cederbaum. An algorithm for planarity testing of graphs. *Theory of Graphs: International Symposium*, pages 215–232, 1967. [→ p. 141]

D. Lepelley and V. Merlin. Scoring run-off paradoxes for variable electorates. *Economic Theory*, 17(1):53–80, 2000. [→ p. 13]

M. H. Liffiton, A. Previti, A. Malik, and J. Marques-Silva. Fast, flexible MUS enumeration. *Constraints*, 21(2):223–250, 2016. [→ p. 18]

R. J. Lipton, E. Markakis, E. Mossel, and A. Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce (ACM-EC)*, pages 125–131, 2004. [→ p. 135]

Z. Lonc and M. Truszczynski. Maximin share allocations on cycles. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 410–416, 2018. [→ p. 138]

T. Lu and C. Boutilier. Budgeted social choice: From consensus to personalized decision making. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 280–286, 2011. [→ p. 39, 90, 125]

J. Massó and I. M. de Barreda. On strategy-proofness and symmetric single-peakedness. *Games and Economic Behavior*, 72(2):467–484, 2011. [→ p. 40]

J. F. Maurras, K. Truemper, and M. Akgül. Polynomial algorithms for a class of linear programs. *Mathematical Programming*, 21(1):121–136, 1981. [→ p. 126]

R. Meir, A. D. Procaccia, J. S. Rosenschein, and A. Zohar. Complexity of strategic behavior in multi-winner elections. *Journal of Artificial Intelligence Research*, 33:149–178, 2008. [→ p. 86]

L. N. Merrill. Parity dependence of a majority rule characterization on the Condorcet domain. *Economics Letters*, 112(3):259–261, 2011. [→ p. 31]

F. Meunier and S. Zerbib. Envy-free divisions of a partially burnt cake. 2018. arXiv:1804.00449. [→ p. 155]

B. Monroe. Fully proportional representation. *American Political Science Review*, 89(4):925–940, 1995. [→ p. 90]

H. Moulin. On strategy-proofness and single-peakedness. *Public Choice*, 35:437–455, 1980. [→ p. 37, 38, 39, 40, 41, 53]

H. Moulin. *The Strategy of Social Choice*. North-Holland, 1983. [→ p. 32]

H. Moulin. *Axioms of Cooperative Decision Making*. Cambridge University Press, 1988a. [→ p. 87, 117]

H. Moulin. Condorcet's principle implies the no show paradox. *Journal of Economic Theory*, 45 (1):53–64, 1988b. [→ p. 4, 11, 12, 13, 21, 22, 122]

H. Moulin. *Fair Division and Collective Welfare*. The MIT Press, 2003. [→ p. 66]

K. Nehring and C. Puppe. The structure of strategy-proof social choice, Part I: General characterization and possibility results on median spaces. *Journal of Economic Theory*, 135 (1):269–305, 2007. [→ p. 130]

M. Núñez and M. R. Sanver. Revisiting the connection between the no-show paradox and monotonicity. *Mathematical Social Sciences*, 90:9–17, 2017. [→ p. 23, 25]

S. Obraztsova, Y. Zick, and E. Elkind. On manipulation in multi-winner elections based on scoring rules. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 359–366, 2013. [→ p. 86]

H. Oh, A. D. Procaccia, and W. Suksompong. Fairly allocating many goods with few queries. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pages 2141–2148, 2019. Extended version: arXiv:1807.11367. [→ p. 135, 154]

C. Papadimitriou, M. Schapira, and Y. Singer. On the hardness of being truthful. In *Proceedings of the 49th Symposium on Foundations of Computer Science (FOCS)*, pages 250–259, 2008. [→ p. 77]

J. Pérez. The Strong No Show Paradoxes are a common flaw in Condorcet voting correspondences. *Social Choice and Welfare*, 18(3):601–616, 2001. [→ p. 13, 14]

D. Peters. Complexity of hedonic games with dichotomous preferences. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pages 579–585, 2016a. [→ p. xv]

D. Peters. Graphical hedonic games of bounded treewidth. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pages 586–593, 2016b. [→ p. xv]

D. Peters. Condorcet's principle and the preference reversal paradox. In *Proceedings of the 16th Conference on Theoretical Aspects of Rationality and Knowledge (TARK)*, pages 455–469, 2017a. [→ p. xiii, 28]

D. Peters. Precise complexity of the core in dichotomous and additive hedonic games. In *Proceedings of the 5th International Conference on Algorithmic Decision Theory (ADT)*, pages 214–227, 2017b. [→ p. xv]

D. Peters. Recognising multidimensional Euclidean preferences. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 642–648, 2017c. Extended version: arXiv:1602.08109. [→ p. xv]

D. Peters. Proportionality and strategyproofness in multiwinner elections. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1549–1557, 2018a. [→ p. xiii]

D. Peters. Single-peakedness and total unimodularity: New polynomial-time algorithms for multi-winner elections. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, pages 1169–1176, 2018b. [→ p. xiii]

D. Peters and E. Elkind. Simple causes of complexity in hedonic games. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 617–623, 2015. [→ p. xv]

D. Peters and E. Elkind. Preferences single-peaked on nice trees. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pages 594–600, 2016. [→ p. xiii]

D. Peters and M. Lackner. Preferences single-peaked on a circle. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 649–655, 2017. [→ p. xiii]

H. Peters, H. van der Stel, and T. Storcken. Pareto optimality, anonymity, and strategy-proofness in location problems. *International Journal of Game Theory*, 21(3):221–235, 1992. [→ p. 38, 39]

A. D. Procaccia. Cake cutting algorithms. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 13. Cambridge University Press, 2016. [→ p. 133]

A. D. Procaccia and M. Tennenholtz. Approximate mechanism design without money. *ACM Transactions on Economics and Computation*, 1(4):18:1–18:26, 2013. [→ p. 39]

A. D. Procaccia and J. Wang. Fair enough: Guaranteeing approximate maximin shares. In *Proceedings of the 15th ACM Conference on Economics and Computation (ACM-EC)*, pages 675–692, 2014. [→ p. 134]

A. D. Procaccia, J. S. Rosenschein, and A. Zohar. On the complexity of achieving proportional representation. *Social Choice and Welfare*, 30:353–362, 2008. [→ p. 90]

A. Proskurowski and J. A. Telle. Classes of graphs with restricted interval models. *Discrete Mathematics & Theoretical Computer Science*, 3(4):167–176, 1999. [→ p. 113]

D. Ray. On the practical possibility of a 'no show paradox' under the single transferable vote. *Mathematical Social Sciences*, 11(2):183–189, 1986. [→ p. 13]

J. Robertson and W. Webb. *Cake-cutting algorithms: Be fair if you can.* AK Peters/CRC Press, 1998. [→ p. 133]

J. Rothe, H. Spakowski, and J. Vogel. Exact complexity of the winner problem for Young elections. *Theory of Computing Systems*, 36(4):375–386, 2003. [→ p. 89, 122]

L. Sánchez-Fernández and J. A. Fisteus. Monotonicity axioms in approval-based multi-winner voting rules. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 485–493, 2019. [→ p. 87]

L. Sánchez-Fernández, E. Elkind, M. Lackner, N. Fernández, J. A. Fisteus, P. Basanta Val, and P. Skowron. Proportional justified representation. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 670–676, 2017. [→ p. 76, 80, 81]

M. R. Sanver and W. S. Zwicker. One-way monotonicity as a form of strategy-proofness. *International Journal of Game Theory*, 38(4):553–574, 2009. [→ p. 13, 23, 24, 25, 28]

M. R. Sanver and W. S. Zwicker. Monotonicity properties and their adaption to irresolute social choice rules. *Social Choice and Welfare*, 39(2–3):371–398, 2012. [→ p. 28]

S. Sato. Circular domains. *Review of Economic Design*, 14(3-4):331–342, 2010. [→ p. 122]

S. Sato. A sufficient condition for the equivalence of strategy-proofness and non-manipulability by preferences adjacent to the sincere one. *Journal of Economic Theory*, 148:259–278, 2013. [→ p. 34]

M. A. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10 (2):187–217, 1975. [→ p. 29, 34]

H. E. Scarf. The computation of equilibrium prices: an exposition. *Handbook of Mathematical Economics*, 2:1007–1061, 1982. [→ p. 150]

P. Scheffler. A linear algorithm for the pathwidth of trees. In *Topics in combinatorics and graph theory*, pages 613–620. Springer, 1990. [→ p. 112]

A. Schrijver. *Theory of linear and integer programming.* John Wiley & Sons, 1998. [→ p. 126, 127]

M. Schulze. Condorcet and Participation. Election-Methods mailing list. http://lists.electorama.com/pipermail/election-methods-electorama.com/2003-October/011042.html, 2003. (Accessed: 7 Nov 2015). [→ p. 13, 22]

M. Schulze. Free riding. *Voting Matters*, 18:2–5, 2004. [→ p. 79]

J. Schummer and R. V. Vohra. Strategy-proof location on a network. *Journal of Economic Theory*, 104(2):405–428, 2002. [→ p. 122]

E. Segal-Halevi. Fairly dividing a cake after some parts were burnt in the oven. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1276–1284, 2018. Extended version: arXiv:1704.00726. [→ p. 155]

P. Seymour. Decomposition of regular matroids. *Journal of Combinatorial Theory, Series B*, 28 (3):305–359, 1980. [→ p. 127]

P. Skowron. What do we elect committees for? A voting committee model for multi-winner rules. In *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI)*, pages 1141–1147, 2015. [→ p. 126]

P. Skowron, P. Faliszewski, and A. Slinko. Achieving fully proportional representation: Approximability results. *Artificial Intelligence*, 222:67–103, 2015. [→ p. 87, 90, 93]

P. Skowron, P. Faliszewski, and J. Lang. Finding a collective set of items: From proportional multirepresentation to group recommendation. *Artificial Intelligence*, 241:191–216, 2016. [→ p. 125]

P. Skowron, M. Lackner, M. Brill, D. Peters, and E. Elkind. Proportional rankings. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 409–415, 2017. [→ p. xv]

P. Skowron, P. Faliszewski, and A. Slinko. Axiomatic characterization of committee scoring rules. *Journal of Economic Theory*, 180:244–273, 2019. [→ p. 86]

W. D. Smith. "Participation failure" is forced in Condorcet methods with at least 4 candidates. The Center for Range Voting. http://rangevoting.org/CondPF.html, 2007. (Accessed: 7 Nov 2015). [→ p. 13, 22]

H. Steinhaus. The problem of fair division. *Econometrica*, 16:101–104, 1948. [→ p. 138]

W. Stromquist. How to cut a cake fairly. *The American Mathematical Monthly*, 87(8):640–644, 1980. [→ p. 133, 134, 143]

F. E. Su. Rental harmony: Sperner's lemma in fair division. *The American Mathematical Monthly*, 106(10):930–942, 1999. [→ p. 134, 147, 148, 149]

X. Sui, A. Francois-Nienaber, and C. Boutilier. Multi-dimensional single-peaked consistency and its approximations. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 375–382, 2013. [→ p. 130]

W. Suksompong. Fairly allocating contiguous blocks of indivisible items. In *Proceedings of the 10th International Symposium on Algorithmic Game Theory (SAGT)*, pages 333–344, 2017. Extended version: arXiv:1707.00345. [→ p. 148, 149, 154]

P. Tang and F. Lin. Computer-aided proofs of Arrow's and other impossibility theorems. *Artificial Intelligence*, 173(11):1041–1053, 2009. [→ p. 13, 14]

E. Tardos. A strongly polynomial algorithm to solve combinatorial linear programs. *Operations Research*, 34(2):250–256, 1986. [→ p. 126]

R. E. Tarjan. Two streamlined depth-first search algorithms. *Fundamenta Informaticae*, 9:85–94, 1986. [→ p. 141]

A. D. Taylor. *Social Choice and the Mathematics of Manipulation*. Cambridge University Press, 2005. [→ p. 78]

T. N. Thiele. Om flerfoldsvalg. *Oversigt over det Kongelige Danske Videnskabernes Selskabs Forhandlinger*, pages 415–441, 1895. [→ p. 76]

M. A. Trick. Recognizing single-peaked preferences on a tree. *Mathematical Social Sciences*, 17 (3):329–334, 1989. [→ p. 91, 101]

V. V. Vazirani. Combinatorial algorithms for market equilibria. In N. Nisan, T. Roughgarden, É. Tardos, and V. Vazirani, editors, *Algorithmic Game Theory*, chapter 5, pages 103–134. Cambridge University Press, 2007. [→ p. 47]

L. Yu, H. Chan, and E. Elkind. Multiwinner elections under preferences that are single-peaked on a tree. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 425–431, 2013. [→ p. xiii]

W. S. Zwicker. Introduction to the theory of voting. In F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors, *Handbook of Computational Social Choice*, chapter 2. Cambridge University Press, 2016. [→ p. 34]