

Voting Rules for Expressing Conditional Preferences in Multiwinner Elections

Papasotiropoulos Giorgos

A.0020

Examination Committee:

Fotakis Dimitris,

School of Electrical and Computer Engineering,
National Technical University of Athens.

Leonardos Nikos,

Department of Informatics and Telecommunications,
National and Kapodistrian University of Athens.

Markakis Evangelos,
Department of Informatics,

Athens University of Economics and Business.

Pagourtzis Aris,

School of Electrical and Computer Engineering,
National Technical University of Athens.

co-Supervisors:

Markakis Evangelos, asst prof

Pagourtzis Aris, assoc prof

April, '19

Λογική και Διακριτά
Μαθηματικά - 2016
ΑΠΘ
Μεταπτυχιακό Πρόγραμμα
«Αλγόριθμοι και
Προβλήματα»

ABSTRACT

Computational Social Choice studies the aggregation of individual preferences toward a collective decision from an algorithmic point of view. Various problems in multiagent systems, decision making technologies, network design, policy making, recommendation systems and so on, require the design and theoretical evaluation of a wide range of voting rules.

In the first chapter we present the origins, possible applications, some of the subtopics of Computational Social Choice as well as a historical overview of the field. In the second chapter we introduce the reader to election scenarios with more than a single winner by describing some commonly desired properties of multi-winner voting rules and defining the most widely used rules together with a glance at algorithmic and computational aspects. Since in many voting settings, voters wish to be allowed to express preferential dependencies, in the third chapter we focus on elections on combinatorial domains by presenting some specific applications along with some solutions which have been proposed in order to deal with combinatorial votes. Ultimately, in the fourth chapter we describe the recently proposed model for handling conditional approval preferences on multiple binary issues followed by new contributions which mainly concerns optimum and approximate results for minisum and minimax conditional approval voting rule.

ΠΕΡΙΛΗΨΗ

Ο τομέας της Υπολογιστικής Θεωρίας Κοινωνικής Επιλογής μελετά, από αλγοριθμική σκοπιά, την αποτίμηση των προσωπικών προτιμήσεων προς μια συλλογική απόφαση. Πληθώρα προβλημάτων σε πολυπρακτορικά συστήματα, τεχνολογίες λήψης αποφάσεων, σχεδιασμό δικτύων, πολιτικό σχεδιασμό, συστήματα συστάσεων και άλλα, απαιτούν το σχεδιασμό και τη θεωρητική αξιολόγηση κανόνων ψηφοφορίας.

Στο πρώτο κεφάλαιο παρουσιάζουμε την προέλευση, ορισμένες εφαρμογές και υποπεριοχές μαζί με μία ιστορική επισκόπηση του αντικειμένου. Στο δεύτερο κεφάλαιο, εισάγουμε τον αναγνώστη σε εκλογικά σενάρια με περισσότερους από έναν νικητές, περιγράφοντας κάποιες επιθυμητές ιδιότητες των σχετικών κανόνων ψηφοφοριών και ορίζοντας τους πιο συχνά χρησιμοποιούμενους κανόνες μαζί με μία ματιά στα γνωστά αλγοριθμικά και υπολογιστικά τους αποτελέσματα. Μιας και σε πολλές περιπτώσεις, οι ψηφοφόροι επιθυμούν να τους επιτραπεί να εκφράσουν εξαρτήσεις μεταξύ των θεμάτων, όταν καλούνται να αποφασίσουν για περισσότερα από ένα θέματα, στο τρίτο κεφάλαιο εστιάζουμε σε εκλογές συνδυαστικής φύσεως, παρουσιάζοντας ορισμένες σχετικές εφαρμογές μαζί με λύσεις που έχουν προταθεί για την αντιμετώπιση αυτών των περιστάσεων. Τέλος, στο τέταρτο κεφάλαιο, περιγράφουμε ένα μοντέλο για χειρισμό ψήφων αποδοχής υπό συνθήκες σε πολλαπλά δυαδικά ζητήματα, ακολουθούμενο από ορισμένα νέα αποτελέσματα που αφορούν κυρίως βέλτιστους και προσεγγιστικούς αλγορίθμους για τον *minisum* και τον *minimax* κανόνα.

CONTENTS

	Page
1 Introduction	1
1.1 Social Choice Theory	1
1.2 Computational Social Choice	3
1.3 Historical Overview	5
2 Multiwinner Voting	9
2.1 Types of Multiwinner Elections	9
2.2 Some Multiwinner Voting Rules	12
2.3 Computational Aspects	20
3 Combinatorial Voting	27
3.1 Types of Combinatorial Elections	27
3.2 Dealing with Combinatorial Votes	29
4 Conditional Approval Voting	43
4.1 Basic Notation	43
4.2 Contribution	51
4.3 Negative Results	53
4.4 Positive Results	54
Bibliography	i

CHAPTER 1

INTRODUCTION

We frequently across situations that a collective decision has to be made taking into account the preferences of some individuals. Computational Social Choice is the field which tries to give both "efficient" (computable in polynomial time, in terms of input) and "fair" (obey to given axioms) algorithms. This introductory¹ chapter is a short discussion of definitions, applications and topics included in Social Choice Theory and Computational Social Choice together with a few remarkable results from the history of both and it is mainly based on the introductory chapters of [ASS10], [Bra&al.16] and [End17].

Contents

1.1 Social Choice Theory	1
1.2 Computational Social Choice	3
1.3 Historical Overview	5

1.1 Social Choice Theory

Social Choice is an old field which lies at the intersection of political science, philosophy, mathematics and economics. It studies the aggregation of individual preferences towards a collective decision by principally designing and evaluating theoretically a wide range of voting rules. We can consider it as a study of decision procedures which map collective outputs to individual inputs. Social Choice theory is primarily motivated by the democratic premise that social policy and group choice should be based on preferences

¹I couldn't find out what the best way to compose current thesis is, until I came up with the advise of the social choice theorist Charles Dodgson (also known by his pen name Lewis Carroll): "Begin at the beginning and go on till you come to the end; then stop".

1.1. Social Choice Theory

of the society [Fis15]. Dealing with the problem "given a large but finite set of criteria, and a large but finite number of alternatives, how can the criteria be ranked in priority order, and how should the alternatives be ranked from best to worst consistent with the ordering of criteria that may be conflicting or incommensurable?", in [RA86] is stated that the surest path to a solution starts with an axiomatic framework. Not long after, we will more extensively discuss that, surprisingly, no aggregation method exists for aggregating preferences of two or more individuals over three or more alternatives, such that the method satisfies five seemingly plausible axioms. In short, in [Lis13] is reported that Social Choice Theory is the field which by examples, general models and theorems is trying to cope with questions such as:

- How can a group of individuals choose a winning outcome (e.g., policy, electoral candidate) from a given set of options?
- What are the (either desirable or unwanted) properties of different voting systems?
- When can a voting system be consider as fair?
- How can a collective (e.g., electorate, legislature, court, expert panel, or committee) arrive at coherent collective preferences or judgments on some issues, on the basis of its members' individual preferences or judgments?
- How can we rank different social alternatives in an order of social welfare?

In Social Choice, a problem can be considered as solved if a procedure which meets some desired criteria exists. Interestingly, not much work has been done on the computational complexity of this procedure. We may imagine that in some settings it is preferable having a non-optimum but computable algorithm instead of an optimum but computationally intractable one. And here is where Theoretical Computer Science come into play.

1.2 Computational Social Choice

In order to solve efficiently problems that everyday arise in AI or in other fields an interdisciplinary research area was propounded. Computational Social Choice is a field closely related to theoretical computer science and game theory which contemplates the difficulty of determining the output of a known voting rule. It is mainly focused on designing and analyzing -from the perspective of computational complexity, approximation and parameterized theory- algorithms for social choice problems as well as applying mechanisms for collective decision in real-world problems such as in multi-agent systems, decision making technologies, network design, policy making, recommendation systems, distributed computing, information retrieval and so on. While we already briefly mentioned some general problems on which the researchers of the field were interested over the last years, we will next list some other wide categories of topics of major interest as mentioned in [Che&al.07]:

Preference Aggregation and Voting: A preference relation from every individual agent is given and we look for a function that maps that profile into a single socially preferred alternative, a set of alternatives or a ranking over the alternatives. Voting can be viewed as a subtopic of Preference Aggregation. Albeit it is widely considered as the main interest of social choice theory, this paragraph is giving as few information as possible since the rest chapters of the current thesis will exclusively and extensively refer to it.

Resource Allocation and Fair Division: A set of agents are asked for a valuation function over a set of objects that are going to be allocated to the agents as per a pre-selected mechanism. Those objects are either infinitely divisible or discrete. For instance, in the continuous model we may have to cut a cake into pieces in order to satisfy some people with different preferences: some prefer a piece covered with chocolate and biscuits, some could prefer a piece with as less chocolate as possible² and some just want a big piece of cake irrespective of its constitution. A less intuitive example of that case is an allocation of percentage of time to agents over a shared good. On the other

²Beware of those people!

hand some items must be entirely allocated to one and only agent. Let's consider an example where some runways of an airport have to be allocated to some aircrafts. An allocation that cedes an airway to more than a plane at the same time will -probably- not be favored by passengers. The allocation mechanism must satisfy the voters given some specific desired requirements and additionally must be a fair one. A common requirement could be to allocate the resources in a way that the total value is maximized (minisum solution) or in a way that the least satisfied agent is as much satisfied as possible (minimax solution). A lot of fairness notions for an allocation have been proposed over the years and we refer the reader to [Ama&al.17].

Coalition Formation: A field which focuses on the procedures appearing when agents cooperate instead of compete each other. Let's think of an example where the winning committee must have at least 50 votes and there are five parties A, B, C, D, E where A has been approved by 48 voters and every other by just 13. All four could make a coalition with A in order to govern but in that way (and if the power is divided according to strength) they will have the least power. On the other hand if they form a coalition they will all be winners with equal power, leaving the stronger party away from power³. In coalition formation the main issues are how the coalitions will be formed and how the surplus should be divided. For more information we refer the reader to a survey which primarily includes computational aspects of Coalition Formation theory along with some of its applications [CEW11].

Judgment Aggregation and Belief Merging: Some individuals express their beliefs on an issue and a common collective decision is going to be made. The goal of judgment aggregation is outputting a decision based on agents' votes over a set of complex propositions that may have some interdependency constraints between them. An interesting (and the most discussed in the related literature) introductory example follows: Three judges agreed that in order to imprison a suspect, there must be evidences that he committed a crime and there must be at least a reliable witness that ensures that the suspect is a criminal.

³Despite the discussed analysis, we must have always in mind that since our personal benefits hardly coincides with the benefits of the most powerful one, it's better not to cooperate with him in any case.

A paradox arises if we observe the following table of their beliefs:

	<i>evidences</i>	<i>witness</i>	<i>evidences AND witness</i>
j_1	yes	yes	yes
j_2	yes	no	no
j_3	no	yes	no
majority	yes	yes	no

Despite the facts that on the one hand they all agree about the "AND" rule in order to establish a suspect as guilty and on the other hand, the majority outcome of the rule is "no", we can notice that the conjunction of majority outcomes of each rule separable is "yes"⁴ which concludes the paradox.

1.3 Historical Overview

As previously seen, one of the very many issues included in the space of interest of computational social choice is voting and this thesis is, from now on, exclusively focused on it. Voting engages with the aggregation of individual preferences of several agents (voters) over a given set of alternatives (e.g. candidates) so as to be able to output the most desirable outcome for the group. A historical overview of some representative examples which encompass the interest on voting scenarios follows.

Going back to ancient Greece and India, Aristotle and Kautilya respectively, explored the possibilities of collective decision making and gave attention to many different forms of government. It is worth mentioning that all three principal ancient Greek writers (Aristotle, Plato and Thucydides) mainly described elections in binary domains [HM15], which is -many years later- proven as the only acceptable alternative in order some desirable properties to be fulfilled.

At Roman times, Pliny the Younger, a Roman senator described in one of his letters the following problem: It was a case where some verdicts had the following options about a prisoner's fate: acquittal (A), banishment

⁴Another paradox comes from everyday life: we often come across situations that neither evidences nor reliable witnesses exist and nonetheless the suspect is being imprisoned.

1.3. Historical Overview

(B), condemnation to death (C)⁵. Option A was favored by Pliny and the largest number of supporters although not from the majority of the verdicts, option B had less supporters and option C was favored by only a few. One of the proponents of harsh punishment suggested the withdrawal of the death option. Then, all upholders of C preferred to B, thus option A (which would have been the winning using a plurality rule in all three options) was rejected. Issues of taking control by deleting candidates, election manipulation and undesirable properties of voting rules are all illustrated in the above example.

In the period of Enlightenment, it was a French engineer Jean-Charles de Borda who proposed a method of voting (today known as Borda rule) under which every voter ranks all candidates in order of preference and assign to each candidate c a score equal to the number of candidates that she ranks below c . Using a similar to the previous mentioned example, he argued that his rule did not have the above deficiency (as plurality did).

It was just a bit later when the French philosopher and mathematician Marquis de Condorcet argued against Borda rule using an example similar to the following:

- 4 voters prefer A to B to C,
- 3 voters prefer B to C to A,
- 2 voters prefer B to A to C and
- 2 voters prefer C to A to B.

Under both plurality and Borda rule, B is the election-winner, however a majority of voters prefer A to B and also prefer A to C in direct majority contest. We can think that A (who we call a Condorcet winner) is a good choice as an outcome. Now suppose that two additional voters are added to the election and the preferences become as follows:

- 4 voters prefer A to B to C,

⁵A remark obtained from A. Berkman is a proper respond to the supporters of B and C: "Full freedom is the very breath of the existence of social revolution; and be it never forgotten that the cure for evil and disorder is more liberty not suppression. Suppression leads only to violence and destruction."

- 3 voters prefer B to C to A,
- 2 voters prefer B to A to C and
- 4 voters prefer C to A to B.

We now have to deal with a situation that a majority prefers A to B, a majority prefers B to C and a majority prefers C to A forming a cyclic preference relation. That is known as the Condorcet paradox and shows that there is not always the case that a Condorcet winner exists⁶.

Some years later, in the nineteenth century, a British mathematician and story teller Charles Dodgson designed a rule (today known as Dodgson rule) that circumvent the above difficulty: In cases where a Condorcet winner exist, namely c , he agreed that c must be the winner of the election whereas in other cases he proposed a count on the number of swaps, between two candidates who are adjacent in a preference order of the voters, which are required before each candidate becomes a Condorcet winner. The candidate with the minimum such count is the winner.

Another, more recent, crucial result was due to Kenneth Arrow [Arr12] who proved that there is no reasonable preference aggregation rule that satisfy all three requirements:

1. If every voter prefers A to B then the final choice must not be B.
2. If every voter do not change preferences order between A and B (and may change preferences between every other pair of candidates), the order between A and B in the final outcome must remain unchanged.
3. There is no a single voter (called "dictator") who always determines the rule's outcome.

A generalization of that result is the Gibbard-Satterthwaite theorem (published independently by the philosopher Allan Gibbard in 1973 [Gib73] and the economist Mark Satterthwaite in 1975 [Sat75]) which states that for every voting rule exactly one of the following is always true:

⁶(I wanted the following quote of Marquis de Condorcet to be part of my thesis and here is the place where it seems as less irrelevant.) "Rejecting theory as useless, in order to work only on everyday things, is like proposing to cut the roots of a tree, because they do not carry fruit".

1.3. Historical Overview

1. There is a single voter who always determines the rule's outcome.
2. There are only two possible alternatives for the outcome despite the total number of candidates.
3. Some voters may have incentives to misreport their preferences in order to be more satisfied with the final outcome.

The connection between the theorems of Arrow and Gibbard-Satterthwaite may not be at first obvious but the simple common/parallel proof available in [Ren01] will easily convince the reader. Until those theorems, a concern of Social Choice theorists was to determine a mechanism so that every voter is unable to cheat in any way in order to improve his payoff. Essentially, it is because of those that computational social choice started. They raised the question of how easy it is a manipulation from a computational point of view and a research area that tries to find a mechanism that making it as hard as possible for any voter to cheat. Needless to say that nowadays, manipulation is just one of the two main concerns of Computational Social Choice, along with a study of existing voting rules from the viewpoint of algorithms and computational complexity.

CHAPTER 2

MULTIWINNER VOTING

As we have previously seen, there is a plethora of reasons why societies run elections⁷ some of which aiming at the election of a single candidate, whereas there exist a lot of situations that the goal is the election of a group of candidates. In the current chapter we will consider elections with multiple winners, we will mention some variances of those elections as well as the most widely used voting rules for electing a winning committee and finally we are going to concisely consider computational aspects of those rules by discussing optimum, approximate and parameterized results.

Contents

2.1	Types of Multiwinner Elections	9
2.2	Some Multiwinner Voting Rules	12
2.2.1	Ordinal Multiwinner Voting Rules	13
2.2.2	Approval Multiwinner Voting Rules based on Scoring Functions	16
2.2.3	Approval Multiwinner Voting Rules Based on Distances	18
2.3	Computational Aspects	20

2.1 Types of Multiwinner Elections

In some elections there exist structural sophisticated restrictions like "any member of the winning committee must be accepted by at least a given percentage of the voters". Not taking them into account, we will now discuss some families of elections that comprise and characterize a lot of different voting scenarios. We will from now on focus on elections with more than a

⁷And, in my opinion, there is also a plethora of reasons why citizenry should avoid participating in them nowadays...

2.1. Types of Multiwinner Elections

single winner. In parliamentary elections, it is desirable the winning committee to be the one with highest appreciated candidates although there will be totally dissatisfied minorities. On the other hand there are elections in which we do not mind if the outcome is not the optimum for any voter provided that everyone is -at least poorly- satisfied or provided that some conditions are satisfied (eg if voters are divided into groups, we may want that at least one member from each team is satisfied). Three main types of voting rules, as distinguished in [Fal&al.17], are the following:

Excellence Based. A simple generalization of single winner elections is multiwinner elections that focus on the excellence of each candidate. The winning committee is formed from a set of candidates that are the most appreciated by the voters. A property of that kind of elections is that if there exists a small set of totally similar candidates and one of them is (resp. is not) in the winning committee, all of them must be in it (resp. must not be in it). "The goal is to independently pick the individuals of the highest quality without any regard to any interaction between them". Those elections are often used as a preliminary stage for a single winner determination when some judges, referees or interviewers aim to select a small fraction of the candidates which meets some desired criteria. From the aforementioned group, the judges will eventually in some scenaria select the final winner. Another property that must be satisfied is that a candidate who is taking place in a committee of size k is good enough to participate in a committee of size greater than k .

Diverse Committee. A different set of elections requires that if there exists a small set of totally similar candidates, one and only one must be in the winning committee. Those elections are defined by different principles than Excellence Based elections since interactions between candidates must not be disregarded. For instance, some friends are planning their trip in a new country and they have to make a decision about where to spend their time. They want to visit historical places, traditional villages, commercial centers and finally spend time in nature. The final choice must include at least one option from every group. Another intriguing example is an airline that due to financial and other reasons wishes to select only a few films to provide to passengers who before the flight gave their preference order on all

available films. The goal is to satisfy every traveler even if some of them will not watch their top-ranked movie⁸.

Proportional Representation. The first coming in mind example when thinking of elections are the parliamentary ones. There we wish a committee to be elected so that the opinions of the society are represented proportionally. Thus we seek for a committee of a given size k in a way that each member is approved by approximately $\frac{1}{k}$ of the total number of voters. A definition of proportionality, given in [Bl&al.58], is the ability of a voting rule to reflect within the committee, all shades of political opinion of a society. A desired property of rules for that kind of elections is that if there exist a group of $l \leq k$ candidates and there are $\frac{ln}{k}$ voters who all rank those candidates as their preferred ones, then they should all belong to every winning committee.

Example 1. A simple but motivating example is given in [LS18] for an approval based election. Lets consider 100 voters and candidates $\{a, b, c, d, e\}$ where 66 voters approve $\{a, b, c\}$, 33 voters approve $\{d\}$ and a voter approve $\{e\}$, seeking for a committee with 3 members. A possible proportional committee is $\{a, b, d\}$, a diverse committee is $\{a, d, e\}$ and an excellence-based committee is $\{a, b, c\}$.

In either single or multiwinner elections there are numerous ways for voters to express their opinion. We mainly focus on ordinal and approval preferences. In a voting scenario with ordinal preferences, each voter poses her opinion using a linear order of all the candidates, for instance, if a voter's opinion is $(a \succ b \succ c)$, she prefers a over b and she prefers both a and b over c . There are also cases when ties between outcomes are allowed in the expressed preferences. In an approval voting scenario, each voter approves (votes for) as many candidates as she likes and "best" (the notion of "best" depends on the selected voting rule) candidates are taking place in the winning committee.

Definition 1. A multiwinner election is the quadruple (C, V, P, k) where

⁸Other constraints should also be taken into account such as movies like "Flight", "Cast Away" (and even a part of "Madagascar 2") it is advisable to be avoided.

2.2. Some Multiwinner Voting Rules

- C is the set of candidates with $|C| = m$,
- V is the set of voters with $|V| = n$,
- $P = (P_1, \dots, P_n)$ is the voters' profile where P_i denotes the preferences of i -th voter,
- an integer $k \in [m]$ which denotes the desired size of the output. Obviously if $k = 1$ we are dealing with the special case of a single winner election.

In the ordinal model P consists of a linear order for each voter over all candidates. In the approval model $P_i \in \{0, 1\}^m, \forall i \in [n]$ and

$$P_i(j) = \begin{cases} 1, & \text{if voter } i \text{ approves } j\text{-th candidate,} \\ 0, & \text{otherwise.} \end{cases}$$

In the rest of the chapter we assume that there are n voters and m candidates ($n, m > 1$) and we wish to elect a committee in the sense that every member of it is a winner of the election (allowing ties, there may be more than a single winning committee). We can easily observe that there can be 2^m possible winning sets but in practice there exist restrictions on such possible sets. A restriction that is going to be used throughout, is a cardinality restriction: we only care about committees of a given size k . We rarely care about committees of size m or empty winning sets so unless otherwise stated we suppose that we seek for committees of size k where $1 < k < m$. In some elections there exist representativeness restrictions, such as diversity constraints as those taken into account in [Ben&al.18]: in a national public housing program there are imposed ethnic quotas like every ethnic group must not own more than a certain percentage of houses in every neighbourhood. Other similar examples can be found in [BKP19].

2.2 Some Multiwinner Voting Rules

Definition 2. A multiwinner voting rule is a function that given a positive integer k , where $1 \leq k \leq m$, and an election (as a pair of a set of m candidates and the profile of voters' preferences) returns a nonempty family of subsets of candidates of size k referred to as the winning committees.

A single winning committee may then occur using a tie-breaking mechanism which will not be discussed. In the next subsections we are going to provide the definitions of some common multiwinner voting rules as presented in [Elk&al.17b]. The first subsection is about rules for ordinal preferences and the rest are about rules for approval preferences, based on scoring vectors and distances respectively.

2.2.1 Ordinal Multiwinner Voting Rules

Single Transferable Vote (STV).

STV proceeds in rounds where in every round we check for the existence of a candidate c who is ranked first by at least $q = \lfloor \frac{n}{k+1} \rfloor + 1$ voters. If so, c is included in the winning committee (and excluded from every preference order in the given profile) and the q aforementioned voters are excluded from the process. Otherwise the candidate that is ranked first by the least number of voters is excluded from the process (and from all the preference orders). We continue with the next round until k candidates are selected. For instance let us meditate on the example given in [Fal&al.17].	Algorithm 1 STV(C, V, P, κ) <ol style="list-style-type: none"> 1: $q = \lfloor \frac{n}{k+1} \rfloor + 1$, committee = \emptyset 2: while size of committee $< k$ do <li style="padding-left: 20px;">3: if $\exists c$ ranked first by $A \subseteq V : A \geq q$ then <li style="padding-left: 40px;">4: include c in the committee <li style="padding-left: 40px;">5: exclude c and A from the process <li style="padding-left: 20px;">6: else <li style="padding-left: 40px;">7: exclude candidate ranked the fewest times first <li style="padding-left: 20px;">8: end if 9: end while 10: return committee
---	--

Example 2. An electorate compiled from 6 voters must make a decision about a committee of size 2 from the candidates $\{a, b, c, d, e\}$. Their preferences are explicitly given below:

- Voter 1: $a \succ b \succ c \succ d \succ e$
 Voter 2: $e \succ a \succ b \succ d \succ c$
 Voter 3: $d \succ a \succ b \succ c \succ e$

2.2. Some Multiwinner Voting Rules

Voter 4: $c \succ b \succ d \succ e \succ a$

Voter 5: $c \succ b \succ e \succ a \succ d$

Voter 6: $b \succ c \succ d \succ e \succ a$

At the first round $q = 3$ and there does not exist a candidate who is ranked first by 3 voters whereas all candidates except c is ranked first by only one voter. Using an alphabetical tie-breaking mechanism we exclude a from the election process. At the second round, still no candidates are ranked first by more than 3 voters. Excluding b and proceeding to the third round, we can see that c is the only that is ranked in the first position by more than q voters (the last q of them and the first one) and thus she is included in the committee. Continuing similarly, if we continue breaking ties in alphabetic order the output of STV will be $\{c, e\}$ but if we break ties in reverse alphabetical order the output will be $\{c, d\}$. Hence, STV is strongly dependent from the tie breaking scheme.

Single Non-Transferable Vote (SNTV).

The SNTV voting rule selects the k candidates with maximum plurality score, where the plurality score of a candidate is the number of voters who rank her as their top-preferred.

Bloc.

The Bloc voting rule selects the k candidates with highest k -approval score, where the k -approval score of a candidate is the number of voters who rank her in their top k positions.

k -Borda.

The k -Borda voting rule selects the k candidates with maximum individual Borda score, where the Borda score of a candidate is defined as follows: Let v be a vote over a set of m candidates and c be a specific candidate for which if she is ranked in the i -th place of v 's ranking then $pos_v(c) = i$ and Borda score of c is $\sum_v m - pos_v(c)$.

Example 3. Using the ordinal votes from example 2, the winning committees of size 2 under the above mentioned rules follows. SNTV elects

the candidate c and one other candidate using a tie-breaking mechanism. Bloc rule elects candidate b and either c or a . Borda score of candidates is respectively $30 - (19, 13, 16, 20, 22) = (11, 17, 14, 10, 8)$ and thus k -Borda elects b and c as the winners.

Chamberlin-Courant.

Under Chamberlin-Courant voting rule, the score that a committee receives from a voter v is $m - pos_v(c)$ where c is the committee member that v ranks highest, among all the members of the committee. The

Algorithm 2 CHAMBERLIN-COURANT (C, V, P, k)

```

1:  $S$  : set of all possible committees of size  $k$ 
2:  $\forall s \in S, score(s) = 0$ 
3: for every  $s \in S$  do
4:   for every  $v \in V$  do
5:      $score(s) += max\{m - pos_v(c) : c \in s\}$ 
6:   end for
7: end for
8: sort  $S$  in decreasing order of  $score(s)$ 
9: return first committee of  $S$ 

```

committee with the maximum total score from every voter is the winning. We can think of the rule as follows: each voter choosing as her representative in the committee the member that she prefers the most and contributes to the committee's score the ranking that she gives to her representative. It is worth mentioning here that it is also possible that another scoring vector rather than the Borda is used.

Example 4. A decision about a sized 2 committee from the candidates $\{a, b, c, d\}$ must be taken from 6 voters. Their preferences are explicitly given below:

Voter 1: $a \succ b \succ c \succ d$

Voter 2: $a \succ b \succ d \succ c$

Voter 3: $d \succ a \succ b \succ c$

Voter 4: $c \succ b \succ d \succ a$

Voter 5: $c \succ b \succ a \succ d$

Voter 6: $b \succ c \succ d \succ a$

2.2. Some Multiwinner Voting Rules

All possible committees of size 2 are $\{a, b\}$, $\{a, c\}$, $\{a, d\}$, $\{b, c\}$, $\{b, d\}$, $\{c, d\}$ and their scores from each one of six voters is given as a sum of six Borda scores as presented below:

committee	score
$\{a, b\}$	$3 + 3 + 2 + 2 + 2 + 3 = 15$
$\{a, c\}$	$3 + 3 + 2 + 3 + 3 + 2 = 16$
$\{a, d\}$	$3 + 3 + 3 + 1 + 1 + 1 = 12$
$\{b, c\}$	$2 + 2 + 1 + 3 + 3 + 3 = 14$
$\{b, d\}$	$2 + 2 + 3 + 2 + 2 + 3 = 14$
$\{c, d\}$	$1 + 1 + 3 + 3 + 3 + 2 = 13$

Thus the winning pair of candidates is $\{a, c\}$ where a represents the first three voters and c the rest.

2.2.2 Approval Multiwinner Voting Rules based on Scoring Functions

Approval Voting Rule (AV).

AV outputs committees of size k those k candidates that are approved the most by the voters. Equivalently each candidate c gains a single point from every voter that approves c and belongs to the winning committee if there are no $k - 1$ other candidates with more points than c .

Approval Based Chamberlin-Courant Rule.

Under Chamberlin-Courant voting rule, the score that a committee receives from a voter v is 1 if there is a candidate approved by v in the committee, or 0 otherwise.

Proportional Approval Voting (PAV).

PAV is a rule that satisfies many axioms related with the proportional representation of the voters' views in the winning committee. Under this rule each voter that approves i members of the committee assigns to the committee $(1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{i})$ where $i \leq k$. The committee with the maximum such score is the winning one.

Satisfaction Approval Voting (SAV).

It is defined as a voter's v satisfaction score $s(v)$, the fraction of her ap-

proved candidates who are elected. The committee that maximizes voters' satisfaction scores is the winning.

Example 5. Lets consider the following approval ballots from the example given in [Fal&al.17]:

voter	approval ballot
Voter 1	$\{a, b, c\}$
Voter 2	$\{a\}$
Voter 3	$\{d\}$
Voter 4	$\{b, c, d\}$
Voter 5	$\{b, c\}$
Voter 6	$\{b\}$

The most approved candidates are b and c since they are approved 4 and 3 times respectively whereas all other candidates are approved less than twice. Approval Based Chamberlin-Courant rule elects as winning committees either $\{a, b\}$ or $\{b, d\}$ as we can confirm using the following table.

committee	Approval CC score
$\{a, b\}$	5
$\{a, c\}$	4
$\{a, d\}$	4
$\{b, c\}$	4
$\{b, d\}$	5
$\{c, d\}$	4

PAV outputs the committees $\{a, b\}$, $\{b, c\}$, $\{b, d\}$ as we can confirm from the following table.

committee	PAV score
$\{a, b\}$	$1.5 + 1 + 0 + 1 + 1 + 1 = 5.5$
$\{a, c\}$	$1.5 + 1 + 0 + 1 + 1 = 4.5$
$\{a, d\}$	$1 + 1 + 1 + 1 + 0 + 0 = 4$
$\{b, c\}$	$1.5 + 0 + 0 + 1.5 + 1.5 + 1 = 5.5$
$\{b, d\}$	$1 + 0 + 1 + 1.5 + 1 + 1 = 5.5$
$\{c, d\}$	$1 + 0 + 1 + 1.5 + 1 + 0 = 4.5$

2.2. Some Multiwinner Voting Rules

Finally SAV outputs the committee $\{a, d\}$ as we can confirm from the following table.

committee	s(1)	s(2)	s(3)	s(4)	s(5)	s(6)	total sat.
$\{a, b\}$	2/3	1	0	1/3	1/2	1	3.4
$\{a, c\}$	2/3	1	0	1/3	1/2	0	2.4
$\{a, d\}$	1/3	1	1	1/3	0	0	3.6
$\{b, c\}$	2/3	0	0	2/3	1	1	3.2
$\{b, d\}$	1/3	0	1	2/3	1/2	1	3.4
$\{c, d\}$	1/3	0	1	2/3	1/2	0	2.4

2.2.3 Approval Multiwinner Voting Rules Based on Distances

In the previous subsection we discussed some voting rules based on scoring functions. In current subsection we will focus on rules based on distances. Despite the fact that there are plenty of distance functions which can be used, we will only study Hamming and Euclidean distances. Since the most frequently used distance function in voting scenaria is Hamming distance, we will firstly define it formally.

Definition 3. Hamming distance is a metric for comparing two binary strings of equal length and is equal to the number of bit positions in which the two strings are different.

We define the Hamming distance between a voter's ballot and a committee as the number of candidates in the committee that are not approved by the voter increased by the number of candidates that are approved by the voter and are not taking part in the committee.

Two of the most debated solutions on electing a k -member committee are minisum (or candidate-wise majority) and minimax. In fact minisum is same as av rule, which has previously been mentioned.

Minisum Solution outputs the committee which (when seen as a 0/1-vector) minimizes the sum of the Hamming distances to the ballots. Equivalently, it simply elects a set, of size k , containing the most approved candidates.

Minimax Solution outputs the committee which (when seen as a 0/1-vector) minimizes the maximum Hamming distance to all the ballots.

Example 6. There are 4 voters that have to make a decision about a two-members committee between candidates $\{a, b, c, d, e\}$. Their approval ballots are noted with 1 in the following table.

	a	b	c	d	e
v_1	1	1	0	0	1
v_2	1	1	0	0	0
v_3	1	1	1	1	0
v_4	0	0	1	1	1

The minisum solution is $\{a, b\}$ with distances $\{1, 0, 2, 5\}$, but $\{a, c\}$ (the minimax solution) has distances $\{3, 2, 2, 3\}$ thus is more widely acceptable.

It is true that although minisum is a computable solution it does not always output a widely acceptable committee. In the case of minimax, majority tyranny is avoided, but the problem becomes NP-complete. Additionally, every optimum solution for the minimax solution, is manipulable, which means that there exist at least one voter which has incentives to misexpress its opinion in order to end up with a more preferable result.

We will in short refer to a prominent generalization of the minisum and minimax rule. There is a family of rules resulting from Ordered Weighted Averaging (OWA) operators that minisum and minimax are just two extreme rules for it. Each rule is defined by a weight vector (which in the i -th position shows the importance of the i -th largest Hamming distance of all the voters). Since minisum perceives each distance as of equal weights, the vector $(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n})$ corresponds to the minisum solution. On the other hand, minimax only taking into account the maximum distance thus the vector $(1, 0, \dots, 0)$ corresponds to the minimax solution. Examples, negative and positive results (showing that there exist exact and approximately computable families of vectors other than the minisum but there is also a wide non-computable spectrum of vectors) can be found in [Ama&al.15].

We close the current section by mentioning a model of preferences (also studied recently in [Elk&al.17a] where an experimental work over multiwinner voting rules concerning their best possible applications between

2.3. Computational Aspects

those mentioned in the Introduction of the current chapter) called the Two-Dimensional Euclidean Model of preferences. In this model every voter and candidate is represented by a point in the two-dimensional space (in the world of politics the point can be easily derived from the opinions on two different issues). Each voter tends to prefer the closest (using the euclidean distance) candidates to her.

2.3 Computational Aspects

The current section contains a state of art from a computational point of view about the previously discussed voting rules, mainly focusing on mini-max solution.

It is not hard to observe that STV, SNTV, Bloc and k -Borda can be computed in polynomial time (unless of course a non-computable tie-breaking rule is used). In short, STV proceeds in rounds where in every round at least one candidate is being excluded from the process until the desired number of candidates is reached. In every round we only need to compute the number of voters who ranked each candidate first. Taking into account that each round of STV is polynomially computable we come to a conclusion that SNTV (which seems like a single round of STV) is also polynomially computable. At what it concerns Bloc and k -Borda we just need to mention that those rules only require some simple computations in order to assign a score to every candidate and some comparisons in order to elect those with the highest such score.

On the other hand, clearly Chamberlin-Courant computation requires exponential time but an approximation solution is given in [LB11]. For a wide range of parameterized results we refer the reader to [YW18]. Approval Chamberlin-Courant rule is also NP-hard to compute and an interesting equivalence with MAX-COVER occurs. MAX-COVER is defined by a set N of n elements, a family S of m subsets of N , and an integer k and we seek for a subcollection of S of size at most k that covers the maximum possible number of elements from N . The intuitive definition from the Chamberlin-Courant rule can lead to a one-to-one correspondence between candidates and sets in S , voters and elements and finally, approval ballots and inclusion

of element in a set. Lastly, FPT approximation schemes (both randomized and deterministic) was given in [SF17], for the case when each voter approves at most p candidates, with respect to the parameter $k + p$.

The AV rule is polynomial time computable while PAV is not (which is proved in [Azi&al.15] using a reduction from INDEPENDENT SET). A randomized algorithm for PAV that gives a 2.3589-approximation in expectation is suggested in [BSS18].

Lastly, needless to say that (like AV rule) the minisum solution for approval based elections based in Hamming distances is also polynomially solvable.

Concerning the state of the art for minimax approval voting, it was introduced in [BKS04] and some exponential algorithms for its computation was given in [BKS07] and [KBS06]. As it has been already mentioned, it is an NP-complete problem, as proven in [LeG04] by a reduction from VERTEX COVER. The work done on approximation algorithms contains a 3-approximation algorithm given in [LMM06], a 2-approximation algorithm in [CKM10] and two PTAS given in [BS14] and (a faster one) in [Cyg&al.18]. A parameterized analysis both on complexity and algorithms also exists.

At what follows we will discuss in details the minimax problem from the approximation and parameterized point of view, but first we will give a formal definition of the problem:

MINIMAX APPROVAL VOTING (MAV)

Input: a set A of m candidates, $k \in [m]$, $P = (P_1, \dots, P_n)$ where P_i denotes the preferences of the i -th voter, ballots $v_i \equiv P_i \in \{0, 1\}^m$, $i \in [n]$, $wt(v_i) = \sum_j v_i(j)$, $maxscore(v) = \max_i H(v, v_i)$

Output: $v^* : wt(v^*) = k$, $maxscore(v^*) = \min_i \{maxscore(v_i)\}$

For the 3-approximation algorithm we shall define the k -completion of a vector v as a new vector v' with $wt(v') = k$ for which $H(v, v')$ is the minimum possible Hamming distance between v and every v' with $wt(v') = k$. The algorithm picks arbitrarily one of the m ballots and it outputs a k -completion of it. Formally, the 3-approximation can be achieved using the algorithm 4 together with the procedure given in algorithm 8.

2.3. Computational Aspects

Algorithm 3 COMPLETION(v, k)

```

1: if wt( $v$ ) <  $k$  then
2:   pick  $k - \text{wt}(v)$  coordinates of  $v$ 
   that are 0 and set them to 1
3: else if wt( $v$ ) >  $k$  then
4:   pick wt( $v$ ) -  $k$  coordinates of  $v$ 
   that are 1 and set them to 0
5: end if
6: return  $v$ 

```

Algorithm 4 MAV(A, P, k) [LMM06]

```

1:  $v \leftarrow$  an arbitrary ballot
2:  $\bar{v} \leftarrow$  COMPLETION( $v, k$ )
3: return  $\bar{v}$ 

```

The computation of a k -completion can be easily done in polynomial time and the analysis of the algorithm consists basically in two applications of the triangle inequality:

Proof. Let v^* be the optimal solution, $\text{OPT} = \text{maxscore}(v^*)$ and \bar{v} be the solution produced by algorithm 4, then for every voter $i \in [n]$:

$$H(\bar{v}, v_i) \leq H(\bar{v}, v) + H(v, v_i), \quad (1)$$

$$\text{and also } H(v, v_i) \leq H(v, v^*) + H(v^*, v_i). \quad (2)$$

$$\begin{aligned} \text{Combining (1) and (2): } H(\bar{v}, v_i) &\leq H(\bar{v}, v) + H(v, v^*) + H(v^*, v_i) \\ &\leq H(v^*, v) + H(v, v^*) + H(v^*, v_i), \end{aligned}$$

where all three are less or equal to $\text{maxscore}(v^*)$, concluding that $H(\bar{v}, v_i) \leq 3\text{OPT}$, which proves the desired. □

It is worth mentioning that algorithm 4 achieves approximation ratio of 2 if there exists a voter v who approves exactly k candidates, since by picking v 's ballot, a completion is not needed and thus only a single use of triangle inequality suffices.

The next two algorithms (2-approximation and PTAS) make use of the idea that minimax approval voting can be formulated as ILP. The formulation used for the 2-approximation solution can be seen nearby. The

$$\begin{aligned} \min \quad & d, \\ \text{s.t.} \quad & d \geq k + |P_i| - 2 \sum_{a \in P_i} x_a, \forall i \in [n] \\ & \sum_{a \in A} x_a = k \\ & x_a \in \{0, 1\}, \forall a \in A \\ & d \geq 0 \end{aligned}$$

2-approximation algorithm is based on rounding the fractional solution of a linear programming relaxation for minimax approval voting. In this way a fractional solution is obtained by including in the committee the k largest variables (by breaking ties arbitrarily) using the algorithm 5:

Algorithm 5 MAV(A,P,k) [CKM10]

- 1: solve the LP-relaxation for $x_a \in [0, 1]$ and obtain a solution (d^*, x^*)
 - 2: choose the set K of the k elements with highest values in x^* (b.t.a)
 - 3: **return** K
-

The proof of approximation ratio (which is proven as tight) consists of a few combinatorial arguments which demonstrate that the Hamming distance of the most disappointed voter from the solution obtained by the LP-rounding is no more than two times the optimum:

Proof. For every voter $i : Y_i = P_i \cap K$ and let j be the voter whose $H(P_j, K) \geq H(P_i, K)$.

For the sake of contradiction, let's assume that $H(P_j, K) > 2d^*$.

$$k + |P_j| - 2|Y_j| \geq H(P_j, K) > 2d^* \geq 2(k + |P_j| - 2 \sum_{a \in P_j} x_a^*) \Rightarrow$$

$$k + |P_j| - 4 \sum_{a \in P_j} x_a^* < 0 \quad (1)$$

$$\forall a \in (K \setminus Y_j) : x_a^* \geq \max_{a' \in P_j \setminus Y_j} x_{a'}^* \Rightarrow$$

$$\sum_{a \in K \setminus Y_j} x_a^* \geq (k - |Y_j|) \max_{a' \in P_j \setminus Y_j} x_{a'}^* \geq \frac{(k - |Y_j|) \sum_{a' \in P_j \setminus Y_j} x_{a'}^*}{|P_j| - |Y_j|} \quad (2)$$

$$\text{if } a \in (K \setminus Y_j) \text{ then } a \in (A \setminus P_j) \Rightarrow \sum_{a \in A \setminus P_j} x_a^* \geq \sum_{a \in K \setminus Y_j} x_a^* \quad (3)$$

$$\sum_{a \in P_j \setminus Y_j} x_a^* = \sum_{a \in P_j} x_a^* - \sum_{a \in Y_j} x_a^* \geq \sum_{a \in P_j} x_a^* - |Y_j| \quad (4)$$

$$\sum_{a \in A \setminus P_j} x_a^* = k - \sum_{a \in P_j} x_a^* \quad (5)$$

Putting (3), (2), (4) together with (5), yield to a contradiction with (1).

□

2.3. Computational Aspects

Lastly, an equivalent formulation to the previous mentioned is presented nearby and is the one used for obtaining a PTAS for the problem. The (fastest known) PTAS consists of a randomized round-

$$\begin{array}{ll}
 \min & d, \\
 \text{s.t.} & d \geq \sum_{s_i[j]=1}^{j \in [m]} (1 - x_j) + \sum_{s_i[j]=0}^{j \in [m]} (x_j), \forall i \in [n] \\
 & \sum_{a \in A} x_a = k \\
 & x_a \in \{0, 1\}, \forall a \in A \\
 & d \geq 0
 \end{array}$$

ing of the LP optimal solution x^* . To this end, the algorithm creates a vector x which in j -th position is 1 with probability $x^*[j]$ and 0 with probability $1 - x^*[j]$. Then it outputs a k -completion of x . The algorithm is presented using pseudocode in algorithm 6.

Algorithm 6 MAV(A,P,k) [Cyg&al.18]

- 1: solve the LP-relaxation for $x_j \in [0, 1]$ and obtain a solution (d^*, x^*)
 - 2: **for** $j \in [m]$ **do**
 - 3: $x[j] \leftarrow 1$ w.p. x_j^* and $x[j] \leftarrow 0$ w.p. $1 - x_j^*$
 - 4: **end for**
 - 5: $y \leftarrow \text{COMPLETION}(x, k)$
 - 6: **return** y
-

The ratio analysis is based on two lemmata which give upper bounds on the probability that x is far from OPT and the probability that the number of '1s' in x is far from k and we omit their proofs due to their technicalities and we refer the reader to the original paper.

Lemma 1. Let $d(x, P) = \max_{y \in P} H(x, y)$, for any $\epsilon \in (0, 1)$: $\text{OPT} \geq \frac{122 \ln n}{\epsilon^2}$ then $\Pr[d(x, P) > (1 + \frac{\epsilon}{2})\text{OPT}] \leq \frac{1}{4}$.

Lemma 2. Let $d(x, P) = \max_{y \in P} H(x, y)$, for any $\epsilon \in (0, 1)$: $\text{OPT} \geq \frac{122 \ln n}{\epsilon^2}$ then $\Pr[|wt(x) - k| > (\frac{\epsilon}{2}\text{OPT})] < \frac{1}{4}$.

Theorem 1. With probability at least 1/2 the algorithm 6 produces a solution of cost at most $(1 + \epsilon)\text{OPT}$.

Proof.

$$\text{Using lemmata 1 and 2: } \begin{cases} \Pr[d(x, P) > (1 + \frac{\epsilon}{2}\text{OPT})] \leq \frac{1}{4}, \\ \Pr[|wt(x) - k| > (\frac{\epsilon}{2}\text{OPT})] < \frac{1}{4}, \end{cases}$$

$$\text{then with probability at least } \frac{1}{2} \begin{cases} d(x, P) \leq (1 + \frac{\epsilon}{2})\text{OPT}, \\ d(y, x) = |wt(x) - k| \leq \frac{\epsilon}{2}\text{OPT} \end{cases}$$

and using the triangle inequality, with sufficient probability, we have that

$$d(y, P) \leq (1 + \epsilon)\text{OPT},$$

which concludes the proof. □

From the parameterized point of view, a detailed study in [MNS15] proved that MAV is W[2]-hard when parameterized by the size of the committee using a reduction from

parameter	complexity (technique)
m	FPT (Trivial)
n	FPT (ILP)
d	FPT (Search Tree)
k	W[2]-hard (reduction to Hitting Set)
n,k	FPT (Color Coding)
s,d,n	W[1]-hard (reduction to Clique)

HITTING SET. Also, it is still (W[1]-)hard when it suffices to satisfy only a set of size s (a constant given as input) of the voters (authors refer to the problem as MAV OUTLIERS), when parameterized by s, d, n together (using a reduction from CLIQUE). On the contrary, MAV admits FPT algorithms when parameterized by the number of voters (since then only a small modification in the LP can bring a formulation with only a constant number of constraints) or by the number of candidates (trivially). Lastly, an FPT algorithm exists, for a parameterization on d , using the "bounded search tree" technique.

2.3. Computational Aspects

CHAPTER 3

COMBINATORIAL VOTING

Multiwinner voting, which was discussed in the previous chapter, is a precise example of combinatorial voting. In the current chapter we will firstly describe the basic contexts of elections on multiple issues which are either interdependent or not. Our will is finding approaches in order to deal with that kind of elections so that to achieve both low-cost and high-expressivity. In that way, we will present solutions which have been proposed in addition with each drawbacks and some of the paradoxes that might arise.

Contents

3.1	Types of Combinatorial Elections	27
3.2	Dealing with Combinatorial Votes	29
3.2.1	Proposition-wise Aggregation	30
3.2.2	Sequential Voting	33
3.2.3	Explicit Votes	35
3.2.4	Limited Alternatives	36
3.2.5	Partial Report of Preferences	36
3.2.6	Completion Principles	37
3.2.7	Representation Languages	39
3.2.8	Iterative Voting	41

3.1 Types of Combinatorial Elections

Following the categorization given in [LX16] we will introduce the following examples in order to illustrate the amplitude of applications of combinatorial elections. The first category includes scenarios where the voters have to give

3.1. Types of Combinatorial Elections

their approval preferences for some issues. The second category includes scenarios where every issue offers different alternatives to the voters who must come to a collective decision. The last category includes scenarios where the agents must pick out some of the suggested alternatives the order of which does not matter. The set of alternatives in every category is a Cartesian product of finite value domains that concerns each issue. It is obvious that for m binary issues the set of alternatives is $\{0, 1\}^m$.

Multiple Referenda. Consider a scenario when citizens' opinions matter on how the government would spend the money gained from tax revenues. Citizens are asked to vote for or against a set of possible alternatives: Science & Education, Health, Military, Culture. Some may prefer the money to be spent for any of the alternatives but Military⁹, while some others may wish money to be spent for Science & Education, Health and Culture only if money are going to be spent for Military¹⁰ and so on. Another commonly used example is a two-issues election where n voters have to express their opinion on whether they want a swimming pool to be built in their neighbourhood or not and whether they want a tennis court to be built in their neighbourhood or not.

Group Planning. A voting scenario which frequently arises when we and our friends have to make a collective decision on a specific topic. As an example imagine some friends that must decide for a common menu composed of first and main courses, dessert and drink given some prespecified alternatives. It is realistic to expect that there are agents who decide for each issue despite the value of other issues but it is reasonable that many of the agents have preferences for the drink that depends on the collective choice of the main course.

Committee Elections. An obvious example arises in the last category, a category which was extensively studied in the previous chapter. A set of agents have to choose from a set of candidates for a group of representatives in parliamentary elections. Another real-life situation example is created when a group of friends has to choose some sights

⁹Those are the rational ones.

¹⁰It is reasonable to suggest to brush aside the ballots of those, isn't it?

to visit in their trip from a given set or to choose a set of different pizzas for their lunch¹¹.

Voting in combinatorial domains is also related to other issues of social choice theory despite voting such as:

Judgment Aggregation. It is the field that tries to solve the question "How can a group of individuals make consistent collective judgments on a set of logically connected propositions?" ([LP10]). It is notable that the difficulties that arise in judgment aggregation are because logical relations instead of preferential dependencies which are responsible for the difficulties in voting in combinatorial domains.

Fair Division. Fair division is the problem of dividing one or several goods (divisible or indivisible) amongst two or more agents in a way that satisfies a suitable fairness criterion ([End18]). We can think of the set of all allocations as a combinatorial space of alternatives out of which we have to make a collective decision that satisfies the agents the most as we try to do in voting settings. An interesting difference is that in a voting scenario (say a multiple referenda scenario) it is reasonable to make the assumption that all voters are equally concerned with every issue which is not true at all for fair division.

3.2 Dealing with Combinatorial Votes

Many solutions have been proposed in order to map a given profile to an outcome that satisfies the voters. A major issue for all the solutions is a guarantee of trade-off between cost and expressivity. Let's address an intuitive example in the context of multiple referenda given in [LN00]:

Example 7. An electorate compiled from 3 groups of voters of equal size (in every group all voters share exactly the same preference) must make a decision on three interrelated issues. Their preferences are explicitly

¹¹Calling elections for such a situation is (sadly) never in the interests of a pizza contains pineapple.

3.2. Dealing with Combinatorial Votes

given below:

Group 1: $110 \succ 101 \succ 011 \succ 001 \succ 100 \succ 010 \succ 000 \succ 111$

Group 2: $101 \succ 011 \succ 110 \succ 010 \succ 100 \succ 001 \succ 000 \succ 111$

Group 3: $011 \succ 110 \succ 101 \succ 100 \succ 010 \succ 001 \succ 000 \succ 111$

It is easy to see that in a referendum that consists of m issues and n voters, the cost of construction and transmission of preferences is exponentially large. Additionally the voters may be wishless to express a full linear preference on all their alternatives since occasionally they are only able to make a decision for the first (and possibly also for the last) few positions of their preferences.

On the other hand, it could be a solution to express only their top-ranked alternatives. Let's discuss the outcome of the plurality voting rule (in which the winning alternative is the one that is placed in the first place the most times) in the current example. Unfortunately we cannot come to a solution since the three alternatives 110, 101, 011 are tied in the first place. Furthermore if we try to apply the majority rule (in which the winning alternative for each issue is the one which gains the most votes) separately for each of the issues we will output as the winning alternative, the worst for every voter 111.

Suchlike paradoxes will be discussed in the following subsections in addition to the benefits that each solution may have.

3.2.1 Proposition-wise Aggregation

In order to escape some paradoxes a solution could be an issue-by-issue organization of the election process. An order over all issues is at first defined and all voters have to cast their ballots independently on every issue before a common decision for any issue is made.

Definition 4. We say that a (transitive, irreflexive, complete preference relation) linear order \succ on a set of alternatives $A = D_1 \times D_2 \cdots \times D_m$ concerning m variables (issues) is separable if and only if we do not need to know the value of neither issue to find out whether a particular change in the value of an issue will make the voter better or worse off ([XCL11]).

Definition 5. Given a voting profile of n voters with their linear orders on A , the profile is called separable when each linear order is separable.

Given a separable profile over a domain the outcome of proposition-wise aggregation (simultaneous ([LN00]), standard ([BKZ97]) or seat-by-seat ([BK10]) voting) is the one composed by applying a voting rule to every issue separately in a pre-defined order.

The communication cost in a proposition-wise aggregation for binary variables is $\mathcal{O}(np)$ since each voter has only to report one ballot for every variable. For instance if a voter prefers for the first issue 1 over 0 and for the rest two issues prefers 0 over 1 (that is $1_1 \succ 0_1, 0_2 \succ 1_2, 0_3 \succ 1_3$) it suffices to vote for the ballot 100. We see that the communication cost is low in that case (as well as the computation cost, a fact that it is easily observable) but is it a good solution for every multiwinner voting scenario in combinatorial domain?

Some times it is natural to define an order on issues (in the first example of group planning which is described above, it seems a good idea to vote for first course, main course, drink, dessert in that order) but that is not always the case. Consider the following example (which was also previously briefly mentioned as an example of multiple referenda) focuses on an electorate compiled from 3 residents which must make a decision on whether it should be built a swimming pool and/or a tennis court in their neighborhood.

Example 8. Their real preferences are explicitly given below where s and t means "to build a swimming pool" and "to build a tennis court" respectively and \bar{s} and \bar{t} means "not to build a swimming pool" and "not to build a tennis court" respectively:

Voter 1: $st \succ \bar{s}t \succ \bar{s}\bar{t} \succ s\bar{t}$

Voter 2: $s\bar{t} \succ \bar{s}t \succ \bar{s}\bar{t} \succ st$

Voter 3: $\bar{s}t \succ \bar{s}\bar{t} \succ st \succ s\bar{t}$

Let's see what is happening if the voting rule to be used is the majority rule and the voters are voting first for the swimming pool-issue and then

3.2. Dealing with Combinatorial Votes

for the tennis-court-issue. A reasonable thought for the first voter is the following: "If the outcome for the swimming pool-issue is s then the rest of the voters will vote for \bar{t} since both prefer $s\bar{t}$ to st and the final outcome will be $s\bar{t}$ which is my last preferred option. On the other hand if the outcome for the swimming pool-issue is \bar{s} , using the same reasoning, the rest of the voters will vote for t and the final outcome will be $s\bar{t}$ ". From the above, it is clear that voter 1 has incentives to misreport his preferences, which in general is an unwanted property.

Another problem arises if we take a closer look at voter 1. He must fill in at ease reporting a preference only on the swimming pool-issue since it is clear that he prefers the building of the swimming pool if and only if the tennis-court will also be built.

Finally, let the election process begin despite the above reservations supposing that all voters will vote for their true preferences. Using issue-wise majority rule the final outcome would be st which is the worst alternative for all but the first voter.

We observed that separability does not always occur and when it is not, the result of the proposition-wise aggregation might be paradoxical. In fact, under the separability assumption of a profile we avoid some paradoxes described above and also we can identify some desirable properties (such as the election of a Condorcet winner when there exists one ([Kad72])). Consequently, can we be sure that a separable profile is a perfect setting for implementing proposition-wise solution in an election with combinatorial structure? The following example given in [ÖS06] may convince us that it is not.

Example 9. An electorate compiled from 3 voters must make a decision on three interrelated issues. Their preferences are explicitly given below and it is easy to verify that are separable:

Voter 1: $111 \succ 011 \succ 101 \succ 001 \succ 110 \succ 010 \succ 100 \succ 000$

Voter 2: $100 \succ 000 \succ 101 \succ 001 \succ 110 \succ 010 \succ 111 \succ 011$

Voter 3: $010 \succ 011 \succ 000 \succ 001 \succ 110 \succ 111 \succ 100 \succ 101$

proposition-wise voting using majority rule will have as an outcome 110 which is Pareto-dominated by 001 (which means that every voter ranked 110 below 001).

One could now think that the majority rule is the one that causes those paradoxes. We refer the reader to the proof of the following proposition, given in the corresponding paper:

Proposition 1. [BK10] As soon as there are at least three issues or when there are exactly two issues one of which has at least three possible values, then simultaneous voting is efficient if and only if is dictatorial.

3.2.2 Sequential Voting

Sequential voting is a voting scheme which consists of phases. In each phase every single issue in a predefined order is elicited and a decision about it is made and communicated to the voters before they vote on the next variable. Formally, sequential voting is a linear order over the set of m issues \mathcal{O} (let \mathcal{O} be X_1, X_2, \dots, X_m and for each $i \in [m]$ and a resolute (or, in a more general setting which will not be discussed, an irresolute) voting rule r_i for every issue. As we have already seen, in proposition-wise aggregation, it is not always feasible for a voter to express an opinion on a single issue because her preferences on it depends on issues that may not be yet fixed. We call \mathcal{O} -legal a preference relation given a linear order of the issues \mathcal{O} if for every $i \leq m$, X_k is preferentially independent of X_{k+1}, \dots, X_m given fixed values for X_1, \dots, X_{k-1} .

Example 10. Let two binary issues and the 3-voter profile be:

Voter 1: $1_11_2 \succ 0_11_2 \succ 1_10_2 \succ 0_10_2$

Voter 2: $1_10_2 \succ 1_11_2 \succ 0_11_2 \succ 0_10_2$

Voter 3: $0_11_2 \succ 0_10_2 \succ 1_10_2 \succ 1_11_2$

The above profile is not (X_2X_1) -legal because the second voter's prefer-

3.2. Dealing with Combinatorial Votes

ences for issue 2 are depend on the outcome of issue 1. On the other hand, the above profile is $(X_1 X_2)$ -legal since the first voters prefer 1_1 to 0_1 no-matter what the outcome of X_2 be and the third voter prefers 0_1 to 1_1 independently of the value of X_2 .

When \mathcal{O} -legality occurs in a given profile, sequential voting is safe. Although in the absence of it, sequential voting suffers from exactly the same problems as simultaneous voting suffers when there exist non-separable issues. A solution is proposed in [Air&al.11] in which is suggested a design of a voting procedure in order to select the order of the issues.

In the absence of \mathcal{O} -legality in [XCL11] was proposed the so-called strategic sequential voting process for m binary variables with $\mathcal{O} = \{X_1, X_2, \dots, X_n\}$ and the preferences are common knowledge, local voting rules and the order of issues. The above procedure can be solved by backward induction where in each phase all voters perform a dominant strategy, which can be made clearer in the following example obtained from [LX16]:

Example 11. Let two binary issues and the 3-voter profile be:

Voter 1: $1_1 1_2 \succ 0_1 1_2 \succ 1_1 0_2 \succ 0_1 0_2$

Voter 2: $1_1 0_2 \succ 1_1 1_2 \succ 0_1 1_2 \succ 0_1 0_2$

Voter 3: $0_1 1_2 \succ 0_1 0_2 \succ 1_1 0_2 \succ 1_1 1_2$

If the outcome of the first phase is 1_1 then voter's 1 dominant strategy is voting for 1_2 and voters 2 and 3 dominant strategies are to vote for 0_2 and then the winning alternative will be $1_1 0_2$. In the same way, if the outcome of first phase is 0_1 then the winning alternative will be $0_1 1_2$.

Thus, for the first issue each voter will consider to vote after comparing her preferences between $1_1 0_2$ and $0_1 1_2$ which are the only possible outcomes. The first and last voters prefer the outcome $0_1 1_2$ and therefore they will vote for 0_1 .

Results on how well the winner under sequential voting approximates the

winners under some common voting rules, both when the profiles are or are not \mathcal{O} -legal and separable, are given in [CX12]. The undesirable properties of the above mentioned solution are not to be overtaken. The outcome of the above procedure, despite being unique, may be positioned in the lower places of every voter's preferences and may be Pareto-dominated by (or equivalently, ranked by every voter below) a sufficiently big amount of alternatives.

We can conclude that sequential voting is a cheap in communication and computation solution for an \mathcal{O} -legal profile given a predefined commonly accepted and of vital importance ranking \mathcal{O} . On the other hand more difficulties than many of the previous solutions can occur if we try to implement it in non- \mathcal{O} -legal profiles.

3.2.3 Explicit Votes

An alternative to the above discussion is every voter to explicitly give the ordering over all alternatives. In this subsection we will make an attempt to convince the reader that there are a few (and only a few) situations that asking voters to explicitly specify their preference relation by ranking all alternatives explicitly is realistic.

At first we might question whether a voter is able to make such a ranking. Every voter can easily express her first few and her last few preferred outcomes but is it valuable for her to express a linear order in all the different outcomes? Despite the above observation, the computational cost and the paradoxes that could occur in such situations we are only going to deal with the communication cost of ranking all the alternatives.

Suppose that we are given an election scenario where the variables are $\{X_1, X_2, \dots, X_m\}$ each X_i of them can take values from a set $D_i, i \in [m]$. Since the domain of our profile is the Cartesian product $D_1 \times D_2 \times \dots \times D_m$, there are $\prod_{i \in [m]} |D_i|$ possible alternatives. Thus, the complexity is exponentially large which makes the explicit preference elicitation unrealistic unless we consider an election scenario with only a few variables where each of them having only a few alternatives, in which case the currently mentioned technique is clearly the most legitimate.

3.2.4 Limited Alternatives

A proposed solution derived from the limitations that explicit votes have, is to give permission to voters to vote only about specific alternatives.

It is to question about who (and with what kind of knowledge) would have the opportunity to decide which those alternatives would be, since the outcome definitely depends on that choice.

Example 12. n voters are voting for m issues and their linear orders are different in all but the first position where all rank the $1_1 1_2 \dots 1_m$ alternative. It is easy to observe that the best and most fair outcome is the above mentioned alternative for which we have no insights that will be in the available options of the voters in the election process.

It is clear that the preselection phase is able to make the election process very biased, strengthening a lot the authority which is responsive from that phase which of course is not desirable.

Secondly, in order that this solution to be realistic the number of alternatives the voters can vote for has to be as low as possible. Considering the fraction of that number over all the possible alternatives we can observe that the voters will only express their preferences on a tiny fraction of the spectrum of their real alternatives.

3.2.5 Partial Report of Preferences

In order to surmount the difficulties of the limited alternatives-method which are described in the previous subsection we could definitely ask voters to report only a small part of their preferences and apply a voting rule that only needs this information. Plurality is such a rule since it counts for every alternative the total number that occurs in the first position. We cannot slide over a vital drawback of this solution: the outcome of every voting rule that will accompany such a report of preferences will for sure be non-significant as soon as alternatives are much more than the voters. For

instance, let's consider a voting scenario given in [LX16]:

Example 13. When the election process is composed by 10 binary variables and 50 voters, the number of alternatives is 2^{10} which is twenty times larger than the number of voters and it is plausible that each alternative will get no more than a single vote, which of-course cannot help in the decision-making.

3.2.6 Completion Principles

A family of approaches for dealing with combinatorial structure of voting is based on completion principles. In such a way the communication cost is kept low while paradoxes and undesirable properties mentioned in previous subsections are avoided. Voters are asked to report a small part of their preferences (as in partial report of the previous subsection) and then a completion principle is used in order to complete them into full (or partially full) preference relation using a fixed completion principle. Subsequently a voting rule can be used to obtain the result.

The most intuitive way of completion is the distance-based completion given the top-preferred outcome of every voter which makes use of the idea: "the closer an outcome is to the top-preferred of a voter, with respect to a predefined distance metric, the more preferred". An obvious choice (but definitely not the only one) of such a distance is the Hamming and a simple voting rule that can be used (still not the only one) is the minisum voting rule. All other rules that were suggested in the previous chapter for multiwinner elections can further be used while the same occurs for other distances. An example of such distance is Dirac distance, defined by $d(x, y) = 0$ if $x = y$ and $d(x, y) = 1$ otherwise, for vectors of size m in an election with m issues.

Example 14. Let there be three voters that have to vote over two propositions. They only cast their preferred outcome which is respectively 11, 01, 10. The distance from each possible outcome is completed using the Hamming metric and it is available in the following table:

3.2. Dealing with Combinatorial Votes

outcome/vote	voter 1: {11}	voter 2: {01}	voter 3: {01}
{00}	2	1	1
{01}	1	0	0
{10}	1	2	2
{11}	0	1	1

The minimum outcome is {01} and it is easy to verify that the communication cost as long as the computation cost in that case are low.

Unfortunately, the following example can convince the reader that negative results are also present in the current mentioning solution.

Example 15. Let there be five voters that have to vote over three propositions. They only cast their preferred outcome which is respectively 110, 101, 011, 000, 000. The distance from each possible outcome is completed using the Hamming metric and it is available in the following table:

outcome/vote	v_1 : {110}	v_2 : {101}	v_3 : {011}	v_4, v_5 : {000}
{000}	2	2	2	0
{010}	1	3	3	1
{100}	1	1	3	1
{110}	0	2	2	2
{001}	3	1	1	1
{011}	2	2	0	2
{101}	2	0	2	2
{111}	1	1	1	3

We can observe that the minimum voting rule would output $r = \{000\}$ although the majority of voters (voter 1, voter 2 and voter 3) strictly prefers the outcome $\bar{r} = \{111\}$.

Paradoxes like the above can occur not only using Hamming distance and minimum voting rule. For further results we refer the reader to [LL09] and [ÇL12].

Another option is the singleton ranking-based input which is defined by

a completion principle which demands a specific ranking from every voter about every single issue (and not on combinations of them). Afterwards it extends those preferences to preferences over sets of items and finally selects a set of items using a common voting rule for multiwinner voting. In the work [KL05] one can find definitions about necessary (possible) winners, given partial preferences, as the candidates being the winners in all (resp. some) of the complete extensions of the profile, together with computational aspects.

It is easy to observe that the aforementioned solutions have low communication cost, their computation cost derives from the hardness of computing the selected voting rule and the expressivity of the ballots collected are fairly low since no relations between issues are taken into account. Both previous attempts lack applicability because of their low expressivity potential.

3.2.7 Representation Languages

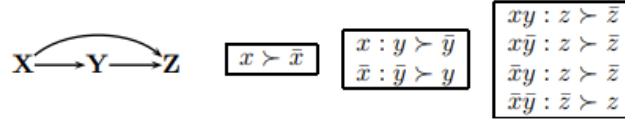
A more expressive way than any of the previous ones, for reporting preferences, is using a representation language. In that case all voters are obliged to vote in an descriptive formal language. Two characteristic examples of such languages are CP-nets (which is going to be described in the current subsection) and Conditional Voting (which is going to be extensively analyzed in chapter 4).

In [Bou&al.04] it was defined as a conditional preference network (or just CP-net) the pair (G, A) where G is a directed graph with one vertex for each variable and A is a set of conditional preference tables, one for each vertex. A voting profile consists of a CP-net for every voter. Things may become clearer considering the following example obtained from [Che&al.08]:

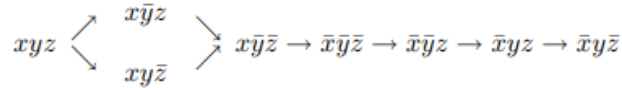
Example 16. A single voter expresses her opinion about 3 issues. She has preferences for the first issue individually, the value of the first issue is affecting the second one and collective values of the first two issues are creating her preferences for the last one. Therefore, the graph-component of her CP-net consists of 3 vertices where the second vertex has an incoming edge from the first (since it is affected by the first issue) and

3.2. Dealing with Combinatorial Votes

an outgoing edge to the third (since it affects the third issue) and the third vertex has two incoming edges one from every vertex (since both previous issues affect the last one). A visualization of the above is given below together with the preferences of the voter concerning each issue:

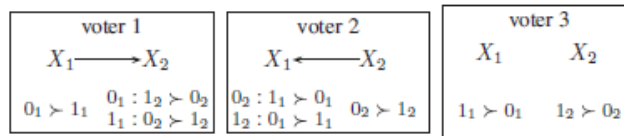


It is easy enough to convert the above CP-net to a preference network. The result of that conversion is the following:



One of the proposed solutions for obtaining an outcome given a CP-net (for others we refer the reader to the methods studied in [LVK10]) consists of creating a collective CP-net from the given CP-nets of every voter and then outputting the non-dominated alternatives exported from the new CP-net, where for every adjacent alternatives a voting rule is used to decide the common preferences. For instance, if we use the majority voting rule for the aggregation of CP-nets, the following is a featuring example:

Example 17. The example is obtained from [LX16].



Since there both exists edges from X_1 to X_2 and vice versa, the created CP-net that combines all three votes is consisted of a graph with 2 vertices and all two edges. The majority rule is then applied in order to obtain the relations that comes with every vertex in the resulting CP-net.

$$\begin{array}{c}
 X_1 \longleftrightarrow X_2 \\
 0_2 : 1_1 \succ 0_1 \quad 0_1 : 1_2 \succ 0_2 \\
 1_2 : 0_1 \succ 1_1 \quad 1_1 : 0_2 \succ 1_2
 \end{array}$$

Thus the voters prefers $1_1 0_2$ and $0_1 1_2$ over both other two alternatives but there is no comparison between them.

Other options that trying a trade-off between cost and expressivity with not-known undesirable properties are conditionally lexicographic preferences ([Boo&al.10]), GAI-nets ([GPQ08]) and conditional approval voting. We will extensively refer to the ultimate in the next chapter.

3.2.8 Iterative Voting

As we will shortly see, Iterative Voting is a topic with many open fields for research, thus we consider appropriate to mention it, but in fact the current subsection is more an alternative way of voting rather than a solution to combinatorial voting.

Let us consider what role do polls play in every typical political voting. In view of that perspective, every voter may re-vote after examining the outcome given her initial ballot. Let's consider a voting game proceeds in turns where each voter report her preferences initially, the votes are aggregated using a predefined voting rule and in each turn each voter may change her initial ballot until everyone become satisfied (or at least has no objections) with the last announcement (a reader may now think correctly that current profile is a pure Nash equilibrium from a game theoretic point of view). An assumption to be made is that every voter acts in every step as she is the last who is able to change her submission since none of them is aware of others preferences and thus none of them can make reliable prediction for future rounds. Some topics concerning iterative voting are

- Outcome-Equilibrium of a given profile
- Guarantee of Convergence to an Equilibrium
- Rounds until Convergence to an Equilibrium
- Price of Anarchy & Social Cost

3.2. Dealing with Combinatorial Votes

As someone can imagine, iterative voting is a Computational Social Choice topic, as close to Game Theory as possible, thus it exceeds the purviews of the present thesis. The reader can find heuristic mechanisms and characterizations of profile-classes together with axiomatic results in the detailed survey [Mei17].

CHAPTER 4

CONDITIONAL APPROVAL VOTING

At what concerns combinatorial domains, as we have seen in the previous chapter, the solution of Representation Languages seems to fulfill our needs for a trade-off between both low communication cost and expressivity. A newly and not much debated idea is the one going to be presented in the current chapter together with some innovative negative and positive results.

Contents

4.1 Basic Notation	43
4.2 Contribution	51
4.3 Negative Results	53
4.4 Positive Results	54
4.4.1 Positive Results for Minisum Solution	54
4.4.2 Positive Results for Minimax Solution	61

4.1 Basic Notation

Let $I = \{I_1, \dots, I_m\}$ be a set of m issues (in predefined order) each of them associated with the domain $D_i = \{I_i, \bar{I}_i\}$, $i \in [m]$. Let $D = D_1 \times D_2 \times \dots \times D_m$ be the set of possible outcomes. In a generalized version we could also assume that the cardinality of each D_i is arbitrary, although we will only focus on elections with binary issues, so called referendums. Then, for simplicity, we will denote the outcome of i -th issue I_i as P_i (or 1_i in some cases) and the outcome \bar{I}_i as N_i (or 0_i in some cases). A group $V = \{V_1, \dots, V_n\}$ of n voters have to decide of a common outcome using a voting procedure and a voting rule.

Voting Procedure: Each voter selects some edges in order to form a graph whose vertex set contains one vertex for each issue. Then, the voter is allowed to cast an approval ballot for every possible outcome of the issues which have no predecessors. Additionally the voter should specify a value for the issue $j \in \{1, \dots, m\}$ for every outcome of each predecessor of j . Thus the voter is allowed for every issue $j \in \{1, \dots, m\}$ to approve none, some or all the possible combinations of $\{t : d_j\}$ where t is a set of possible outcomes for issues that are predecessors of j , which stands for satisfaction with the outcome d_j of the j -th issue if the outcome of any issue that is a predecessor of j -th issue has the value assigned in t .

We will use the notation B_i^j for the ballot (which is mainly a set of approval or conditional statements as we will shortly see) of i -th voter concerning issue I_j , B_i for the ballot of i -th voter concerning all issues and we will say that B is a voting profile if $B = \{B_i^j : i \in [n], j \in [m]\}$. We are finally able to conclude that the discussed model is a generalization of simple approval voting procedure.

Voting Rules: Firstly, we are going to study two commonly used rules for obtaining a winning committee in multi-issue domain from the generalized point of view of conditional approval voting (CAV) as proposed at [BL16]. Conditional minisum rule (CMS) outputs the outcome that minimizes the total number of disagreements over all voters. On the other hand, conditional minimax rule (CMM) outputs the outcome that minimizes the maximum number of disagreements over all voters. Both minisum and minimax rules are belonging to a family of rules defined by a vector w , called w-AV rules which we are going to briefly present soon and they have been studied extensively in [Ama&al.15]. We then proceed to adapt to our setting the w-AV family of rules by defining w-CAV. As we will extensively discuss later, the computational difficulty of the problem increases significantly in our generalized approval model or equivalently when voters are also allowed to cast conditional approval preferences.

Definition 6. A voter's conditional approval ballot over the issues $I = \{I_1, \dots, I_m\}$ with domains D_1, \dots, D_m is a pair $\{G, \{A_i : i \in [m]\}\}$ where:

- G is a directed graph called dependency graph, with $V(G) = I$ and the set $E(G)$ denotes the allowed conditional statements between issues.

When needed, we will refer to the dependency graph of i -th voter as G_i .

- for $i, j \in [m]$, A_i is a set of approval votes $\{d_i\}$ where $d_i \in D_i$ if $\deg_{i_n}^G(I_i) = 0$ or A_i is a set of conditional approval statements $\{d_i : d_j\}$ where $d_i \in D_i$ and $d_j \in D_j$ if $(I_i, I_j) \in E(G)$.

Alternatively, conditional statements could be expressed succinctly, using propositional formulae instead of a table. As in [BL16], for simplicity reasons, we will from now on only mention the $\bigcup_{i \in [m]} A_i$ and omit the graph G , when possible, bypassing the above notation.

Definition 7. Given a voting profile B and an outcome $d = (d_1, d_2, \dots, d_m) \in D = D_1 \times D_2 \times \dots \times D_m$, we say that a voter i is dissatisfied with the j -th issue if $d_j \notin B_i^j$ and there does not exist a set of alternatives t for some issues such that $(t : d_j) \in B_i^j$ and all elements of t are coordinates of d .

From now on, we will call $\delta_v(d)$ the function which measures the dissatisfaction of voter v over the outcome d , as the total number of issues that dissatisfy her. Our δ function generalizes the concept of Hamming distance since $\delta_u(d) = H(v, d)$ for every outcome d when v refers to a set of unconditional ballots of voter u .

The formal definitions of the problems that current work deals with, follows:

CONDITIONAL MINISUM (CMS)

Input: A voting profile B with m binary issues and n voters where each voter casts a conditional approval ballot.

Output: A boolean assignment to every issue which equals

$$CondMinisum(B) = \arg \min_{d \in D} \sum_{i=1}^n \delta(d, B_i)$$

CONDITIONAL MINIMAX (CMM)

Input: A voting profile B with m binary issues and n voters where each voter casts a conditional approval ballot.

Output: A boolean assignment to every issue which equals

$$CondMinimax(B) = \arg \min_{d \in D} \max_{i=1}^n \delta(d, B_i)$$

WEIGHTED CONDITIONAL APPROVAL VOTING (W-CAV)

Input: A voting profile B with m binary issues, n voters where each voter casts a conditional approval ballot and a vector w of size n and coordinates which sum up to 1. $H(d, B)$ stands for the n -dimensional vector that contains the distances of the conditional ballots of B from the outcome d in non-ascending order.

Output: A boolean assignment to every issue which equals

$$CondW(B) = \arg \min_{d \in D} w \cdot H(d, B)$$

Example 18. Consider the following introductory example of a voting profile P . We must note that for the shake of simplicity we do not give a general-case example. Although, unless otherwise stated, in the rest of the chapter we also consider profiles containing edges between non-consecutive issues (so-called "jump edges") and edges from an issue to a previous one (so-called "back edges").

Let there be 4 voters with $G_i = (V, E_i)$ such that $V = \{I_1, I_2, I_3\}$ and

$$E_i = \begin{cases} \{(I_1, I_2), (I_2, I_3)\}, & \text{if } i \in \{1, 2, 3\}, \\ \{(I_1, I_3), (I_2, I_3)\}, & \text{if } i = 4. \end{cases}$$

The voters' preferences are shown on the following table:

voter 1	voter 2	voter 3	voter 4
$\{0_1\}$	$\{0_1, 1_1\}$	$\{1_1\}$	$\{1_1, 0_2\}$
$0_1 : \{0_2, 1_2\}$	$0_1 : 0_2$	$0_1 : \{0_2, 1_2\}$	$0_1 0_2 : 0_3$
$1_1 : 1_2$	$1_1 : 1_2$	$1_1 : 0_2$	$1_1 0_2 : 1_3$
$0_2 : 1_3$	$0_2 : 1_3$	$0_2 : \{0_3, 1_3\}$	$0_1 1_2 : 0_3$
$1_2 : 0_3$	$1_2 : 1_3$	$1_2 : \{0_3, 1_3\}$	$1_1 1_2 : 1_3$

Describing in words some of the preferences we could say that the second voter does not matter about the outcome of the first issue and also prefers 1_3 than 0_3 what ever the value of I_2 is. At first we will examine the dissatisfaction of each voter for some outcomes listing the values of δ function.

$$\delta_1(0_1 0_2 0_3) = 1, \delta_1(1_1 0_2 0_3) = 3, \delta_1(1_1 1_2 1_3) = 2, \delta_1(0_1 0_2 1_3) = 0.$$

$$\delta_2(0_1 0_2 0_3) = 1, \delta_2(1_1 0_2 0_3) = 2, \delta_2(1_1 1_2 1_3) = 0, \delta_2(0_1 0_2 1_3) = 0.$$

$$\delta_3(0_1 0_2 0_3) = 1, \delta_3(1_1 0_2 0_3) = 0, \delta_3(1_1 1_2 1_3) = 1, \delta_3(0_1 0_2 1_3) = 1.$$

$$\delta_4(0_1 0_2 0_3) = 1, \delta_4(1_1 0_2 0_3) = 1, \delta_4(1_1 1_2 1_3) = 1, \delta_4(0_1 0_2 1_3) = 2.$$

Let w be the $(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$ vector (which in fact describes the minisum voting rule), then

$$\begin{aligned} w \cdot H(0_1 0_2 0_3, P) &= w \cdot (1, 1, 1, 1) = \frac{4}{4} \\ w \cdot H(1_1 0_2 0_3, P) &= w \cdot (3, 2, 1, 0) = \frac{6}{4} \\ w \cdot H(1_1 1_2 1_3, P) &= w \cdot (2, 1, 1, 0) = \frac{4}{4} \\ w \cdot H(0_1 0_2 1_3, P) &= w \cdot (2, 1, 0, 0) = \frac{3}{4} \end{aligned}$$

Let w be the $(1, 0, 0, 0)$ vector (which in fact describes the minimax voting rule), then

$$\begin{aligned} w \cdot H(0_1 0_2 0_3, P) &= w \cdot (1, 1, 1, 1) = 1 \\ w \cdot H(1_1 0_2 0_3, P) &= w \cdot (3, 2, 1, 0) = 3 \\ w \cdot H(1_1 1_2 1_3, P) &= w \cdot (2, 1, 1, 0) = 2 \\ w \cdot H(0_1 0_2 1_3, P) &= w \cdot (2, 1, 0, 0) = 2 \end{aligned}$$

So, we can observe that, taking into account only the above mentioned choices, the best outcome using the minisum rule is $\{0_1 0_2 1_3\}$ and the best outcome using the minimax rule is $\{0_1 0_2 0_3\}$ for the given profile P .

Definition 8. Given an instance of conditional approval voting we refer as total graph of dependencies to the graph $\mathcal{G} = (V, E)$ where V is formed from one vertex for each issue and E consists of the union of the edges in every voter's dependency graph. In fact, $\mathcal{G} = \bigcup_{i \in [n]} G_i$.

4.1. Basic Notation

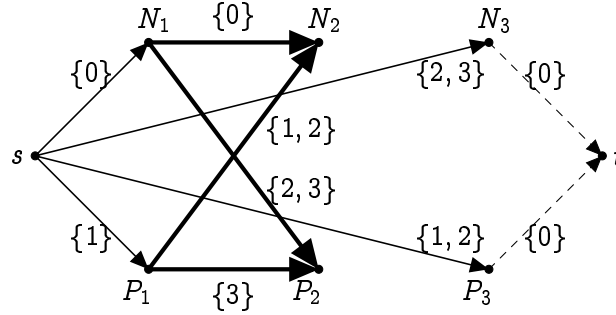
Definition 9. Given an instance of conditional approval voting where for each voter's graph of dependencies G is true that $\max\{deg_{in}^G(u) : u \in V(G)\} \leq 1$, we denote as dissatisfaction graph, the $(m + 2)$ -partite graph $G_{ds} = (V, E, c)$ where each part $i \in \{2, \dots, m, m + 1\}$ has D_i non-adjacent vertices (2 vertices in the binary case). The set of vertices $V(G)$ partitioned in $(m + 2)$ independent sets is exactly the set $\{s\} \cup \{x : x \in D_i, i \in [m]\} \cup \{t\}$. Also, c is a relation that maps every edge to a set of available colors from the power set $P(S)$ where $S = \{c_0, c_1, \dots, c_n\}$ derived from a voting profile B . In fact, c indicates which voters are dissatisfied with every option. We create G_{ds} from a voting profile B with binary issues using the following rules:

- Firstly, we examine the vertices of the graph, excluding s and t , in pairs (issue by issue) in order to add edges that corresponds to ballots from B . For each pair, we add all possible incoming edges from every issue. In every edge (d_k, d_j) we map a set containing all colors (including c_0) except those corresponding to voters who have expressed the opinion $\{d_k : d_j\}$ where $1 \leq k, j < m$ and $d_k \in \{P_k, N_k\}, d_j \in \{P_j, N_j\}$.
- We add every edge from s to $v : v \in V(G_{ds}) \setminus \{s, t\}$, if v is an outcome of the j -th issue, there is at least one voter i for which $deg_{in}^{G_i}(j) = 0$. We assign in that edge a set containing every color (including c_0) except the colors correspond to voters that express the opinion $\{v\}, v \in \{P_j, N_j\}$.
- We also add an edge between t and every possible outcome of the final issue with only c_0 in its set.
- If there is a color set s_e of an edge e , where $c_0 \in s_e$ but $s_e \setminus c_0 \neq \emptyset$ we remove c_0 from s_e , for every such edge.

Example 19. Given that for every voter $i \in \{1, 2, 3\} : G_i = \{\{I_1, I_2, I_3\}, \{(I_1, I_2)\}\}$ and voters' preferences are shown on the following table:

voter 1	voter 2	voter 3
$\{0_1, 0_3\}$	$\{0_1, 1_1\}$	$\{0_1, 1_1, 1_3\}$
$0_1 : \{0_2, 1_2\}$	$0_1 : 0_2$	$0_1 : 0_2$
$1_1 : 1_2$	$1_1 : 1_2$	$1_1 : 0_2$

The resulting dissatisfaction graph that visualizes the above profile where color i indicates the dissatisfaction of i -th voter for $1 \leq i \leq n$ and dissatisfaction of all voters for $i = 0$, follows:



Bold edges correspond to conditional votes, thick edges correspond to approval votes and dashed edges correspond to edges to t .

Definition 10. A layered graph is a connected graph with partitioned vertices in "layers" and each edge connects only vertices in successive layers. We will say that a (directed) graph is formed by contiguous paths if any of its connected components is a layered path graph.

Definition 11. We refer as L_m to the (directed) graph $G = (V, E) : V = I, E = \{\{I_0, I_1\}, \{I_1, I_2\}, \dots, \{I_{m-1}, I_m\}\}$. By referring to a path graph, we may even refer to paths of length 0 or equivalently to isolated vertices, unless otherwise stated.

We are now going to present the definitions of some problems which are going to be used in order to deal with CMS, CMM and W-CAV.

MIN K-SAT

Input: A SAT instance in conjunctive normal form with n variables, m clauses and at most k literals in each clause.

Output: A boolean assignment to every variable so that the total number of the satisfied clauses is minimized.

EXACT MATCHING

Input: Given a graph $G = (V, E)$, a set $E' \subseteq E$ and an integer k .

Output: A perfect matching that consists of exactly k edges of E' .

4.1. Basic Notation

K-BOUNDED COLOR PATH (K-BCP)

Input: A graph $G = (V, E)$ for which the edge set is partitioned into k disjoint color-sets $E_i \subseteq E, i \in k$, two vertices s, t of G and a vector w of size k where $w_i \in \mathbb{N}$.

Output: A path P between s and t such that $|P \cap E_i| \leq w_i, \forall i \in k$.

K-BOUNDED COLOR PERFECT MATCHING (K-BCPM)

Input: A graph $G = (V, E)$ for which the edge set is partitioned into k disjoint color-sets $E_i \subseteq E, i \in k$ and a vector w of size k where $w_i \in \mathbb{N}$.

Output: A perfect matching M such that $|M \cap E_i| \leq w_i, \forall i \in k$.

Finally, we are going to eliminate the worry of those who observed similarities between the procedure of conditional approval voting and the one characterized by CP-nets.

Let us first, remind the reader that in [Bou&al.04] it was defined as a conditional preference network (or just CP-net) the pair of a directed graph with all the variables as its vertices and conditional preference tables one for each vertex. A voting profile consists of a CP-net for every voter. Further discussion can be found in the corresponding section 3.2.7.

It is to be stated that conditional approval voting can be seen as an approval version of CP-nets. Furthermore, we will discuss a major difference that conclude on the incomparability of them even for the binary case (which is also mentioned in [BL16]). The following example will convince the reader that the two methods differ in the semantics.

Example 20. We are going to re-examine the example 16 in order to compare the outcomes $x\bar{y}z$ and $\bar{x}yz$ using CP-nets and conditional approval balloting. As we have already seen, under CP-nets $x\bar{y}z \succ \bar{x}yz$. Using conditional balloting, the voter in the example has the following preferences:

voter v
$\{x\}$
$x : y$
$\bar{x} : \bar{y}$
$xy : z$
$x\bar{y} : \bar{z}$
$\bar{x}y : \bar{z}$
$\bar{x}\bar{y} : z$

Using the previously defined notion of dissatisfaction, $\delta_v(x\bar{y}z) = 2$ and $\delta_v(\bar{x}\bar{y}z) = 1$ and thus $x\bar{y}z \prec \bar{x}\bar{y}z$.

4.2 Contribution

We are aware that CMS and CMM are both NP-hard problems, despite the fact that minisum solution is polynomially computable in the classical approval voting setting. Our negative results concern the NP-hardness of the w-CAV problem for $w = (0, \dots, 0, \frac{1}{n-c}, \dots, \frac{1}{n-c})$ even though it is one of the few known cases that the problem w-AV is efficiently and optimally solvable. We now can arguably believe that conditional votes increment notably the difficulty of the voting procedure.

Therefore we were obliged to make an attempt to cope with that hardness for some special cases of the problems. Indeed we brought out some requirements that could lead to an optimum or at least a low-factor approximation solution in a voting scenario with conditional ballots. In [BL16], are given the only known positive results:

- a trivial solution for CMS when no edges exist in \mathcal{G} and
- a differential approximation ratio of $\frac{4.34}{m+4.34}$ for CMS (for the case of acyclic dependency graph for every voter), where m is the total number of statements given by all voters, using the MAXSAT algorithm given in [EP05].

Given that, we focused exclusively on multiplicative approximation ratios and non-trivial optimum solutions for special cases of dependency graphs for the minisum solution. Our positive results are summarized in the following table:

4.2. Contribution

<i>minisum</i>	
contiguous paths or common graph & paths $\max_{G=G(v), v \in [n]} \{ \max \{ \deg_{in}^G \} \} \leq 1$	OPT ¹ 2.2074-APX ²

1: pick shortest paths (theorem 5), 2: reduction to MIN2-SAT (theorem 6).

On the other hand we also provide algorithms which are able to produce optimum solutions for the minimax case given that either the number of issues or the number of voters is bounded by a constant, us can be seen in the following table:

<i>minimax</i>	
$\mathcal{O}(1)$ issues	OPT ³
$\mathcal{O}(1)$ voters (contiguous paths or common graph & paths)	OPT ⁴

3: brute force (theorem 7), 4: solving K-BCPM (theorem 8).

Example 21. For instance, our work is able to provide an optimum minisum solution for the following case of dependency graphs:

$$v_1 : \quad I_1 \quad I_2 \rightarrow I_3 \rightarrow I_4 \rightarrow I_5 \quad I_6$$

$$v_2 : \quad I_1 \rightarrow I_2 \quad I_3 \rightarrow I_4 \quad I_5 \rightarrow I_6$$

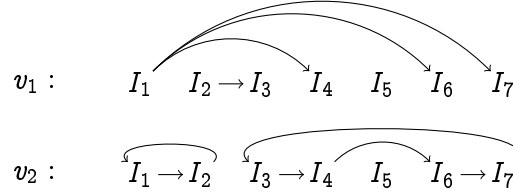
and even for the following one:

$$v_1 : \quad I_1 \quad I_2 \rightarrow I_3 \rightarrow I_4 \quad I_5 \quad I_6 \leftarrow I_7$$

$$v_2 : \quad I_1 \quad I_2 \rightarrow I_3 \rightarrow I_4 \quad I_5 \quad I_6 \leftarrow I_7$$

Also, our work is able to provide an optimum minimax solution for the above cases if either the number of issues or the number of voters is not part of the input.

Additionally, is able to provide an approximate minisum solution for the following case of dependency graphs:



4.3 Negative Results

We are aware that both CMS and CMM are NP-hard problems ([BL16]) despite the fact that only minimax is NP-hard in the non-conditional setting. It is also known that MINISUM and MINIMAX belong to a wide family of voting rules defined by Ordered Weighted Averaging operators ([Ama&al.15]). In addition, it is proven that there are cases of vectors (other than minisum) where the problem of a simple multiwinner election is efficiently solvable. In this section we will prove that even in one of those cases, the conditional version of the problem is still NP-hard.

Theorem 2. w -CAV is NP-hard for $w = (0, \dots, 0, \frac{1}{n-c}, \dots, \frac{1}{n-c})$.

Proof. We are going to present a polynomial time reduction from CMS to w -CAV when $w = (0, \dots, 0, \frac{1}{n-c}, \dots, \frac{1}{n-c})$ which suffices to prove the theorem since, as it has already been discussed, CMS is NP-hard.

Let I be an arbitrary instance of CMS consisting of n voters, m issues, a bound d on the total dissatisfaction and a voting profile B , we are going to construct an instance I' of w -CAV in the following way: let I' be formed by the same m issues, $n' = n + c$ voters where the first n voters have exactly the same preferences as in I and the newly added c voters are dissatisfied with every combination of outcomes of every issue and an additional parameter $d' = \frac{d}{n}$ as a bound on $w \cdot H$. We claim that I is a YES instance for the decision version of CMS if and only if I' is a YES instance for w -CAV.

Suppose that there exists an outcome for B with total dissatisfaction at most $d = \sum_{i=1}^n d_i$, where d_i is the dissatisfaction of the i -th voter. Then

$$w \cdot H = (0, \dots, 0, \frac{1}{n}, \dots, \frac{1}{n}) \cdot (m, \dots, m, d_1, d_2, \dots, d_n) =$$

4.4. Positive Results

$$\frac{1}{n}(d_1 + d_2 + \dots + d_n) = \frac{1}{n} \sum_{i=1}^n d_i \leq \frac{d}{n} = d'$$

so we are also dealing with a YES instance for W-CAV.

For the other direction suppose that B' has a solution with total dissatisfaction at most $d' = \frac{d}{n}$. Then:

$$w \cdot H \leq \frac{d}{n} \Rightarrow \frac{1}{n} \sum_{i=1}^n d_i \leq \frac{d}{n} \Rightarrow \sum_{i=1}^n d_i \leq d$$

so we are also dealing with a YES instance for CMS.

□

4.4 Positive Results

Despite that both CMS and CMM are NP-hard problems ([BL16]), not much work has been done on exact and low-factor approximation algorithms even for special cases of those problems. In current chapter we present the state of the art followed by our ideas for enriching the related literature.

4.4.1 Positive Results for Minisum Solution

We start the current subsection by mentioning and briefly proving the only two already known positive results for CMS.

Theorem 3. [BL16] When the graph \mathcal{G} has no edges (or equivalently when each voter has no dependencies and thus the voting profile is separable) CMS can be solved in polynomial time.

Proof. When no conditional preferences exist the voting procedure almost coincides with the one in a simple multiwinner approval voting election, for which the minisum solution can be easily computed by greedily selecting the outcome issue-by-issue according majority. The only difference that has to be stated is that in our setting a voter may approve none or all the outcomes of an issue which also can easily be handled by a small modification in the algorithm that computes a minisum solution.

□

Theorem 4. [BL16] If \mathcal{G} is acyclic then there is a $4.34/(m+4.34)$ -approximation algorithm for CMS, where m is the total number of statements in all ballots.

Proof. The proof is based on an approximation preserving reduction to MAXSAT problem. We will only mention an example based on the suggested reduction, as done in the short version of the paper. Suppose that a voter's i ballot is $B_i = \{x, x : y, \bar{x} : \{y, \bar{y}\}\}$. Then we create the formula $C_{B_i} = \{x, \bar{x} \vee y, x \vee y \vee \bar{y}\}$. Thus we can create a formula for any ballot and consequently, given a voting profile $B = (B_1, B_2 \dots B_n)$ we are dealing with $C_B = \bigcup_{i \in [n]} C_{B_i}$. □

Theorem 5. (a) When each voter i has a dependency graph G_i formed by collections of contiguous subpaths of L_m (not necessarily common) then CMS can be solved optimally in polynomial time.

(b) When every voter has the same dependency graph, formed by subpaths of L_m (not necessarily contiguous) then CMS can be solved optimally in polynomial time.

Proof. We remind the reader that a weakly connected component of a directed graph G is a maximal subgraph G' such that for each pair of vertices in G' there is a path between them when ignoring edges direction. For the first part of the theorem, we observe that minisum solution in the total dependency graph \mathcal{G} obtained by the aforementioned dependency graphs G_i can be found by computing the minisum solution in every weakly connected component. Suppose that the profile consists of m blocks of weakly connected components where each block is either a path of positive length or a single isolated vertex.

We decompose the profile into m profiles, one for each block and form their dissatisfaction graphs. Every weakly connected component w_i corresponds to dissatisfaction graph connected with s by two edges e_1, e_2 which have its endpoints to the first issue of w_i (and possibly some edges to other issues). We replace those with an extra vertex s_i and two edges between s_i and the endpoints of e_1, e_2 with cost equal to the cost of e_1, e_2 . Additionally, we add a t_i in every dissatisfaction graph that corresponds to w_i with zero cost and start point at every vertex of w_i 's last issue. We then replace the set of colors for each edge with its cardinality (not counting the c_0 color).

4.4. Positive Results

Finally, for every edge $e = \{s, d\}$, $d \in D_i$, $1 < i \leq m$ with weight w_e we add w_e to the weight of every edge $\{u, d\}$, $\forall u$ corresponds to issue i and then we delete e and we ignore s, t vertices. Thus we produced a graph with edges only between consecutive issues. An example of the above discussed transformation follows exactly after the end of the current proof. Some similar arguments can give the same proof by avoiding weakly connected components and forming a graph with every issue and edges only between contiguous issues.

We can observe that the vertices of a path of total weight k in the formed graph corresponds to alternatives of the voting procedure that dissatisfy exactly k voters since each voter that is dissatisfied with an option contributes a unit in the weight of the corresponding edge.

If w_i is a weakly connected component that is a path of length greater than 0, the minisum solution can be found just using an $s_i - t_i$ shortest path algorithm in the created graph and output the alternatives that are related with the vertices belonging to the selected path. It is easily seen that when no jump-edges exist, an $s_i - t_i$ shortest path in the created dissatisfaction graph includes exactly one vertex from every issue contained in the graph.

In the components that are isolated vertices, a choice using the minisum solution as in a single winner voting procedure suffices as described in the theorem 3 and its proof. The union of the selected alternatives from every block is the desired outcome. A more formal description of the algorithm follows.

Algorithm 7 CONTIGUOUS PATHS(I,V,B)

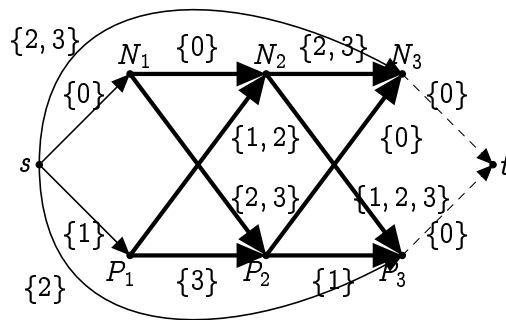
```
1: for  $w_i$  be a weakly connected component of  $\mathcal{G}$  do
2:   if  $|V(G[w_i])| = 1$  then
3:     pick minisum solution for that issue, independent of other choices
4:   else
5:     for  $e \in E(G[w_i])$  do
6:        $c'(e) = |c(e) \setminus c_0|$ 
7:     end for
8:     add a vertex  $s_i$  and an edge  $(s_i, u)$ , for every  $u$  that corresponds
```


- to the first issue of w_i , with $c'(s_i, u) = c'(s, u)$
- 9: add a vertex t_i and an edge (u, t_i) , for every u that corresponds to the last issue of w_i , with 0 cost
 - 10: **for** $e \in E(G[w_i]), e = \{s, d\}, d \in D_i, 1 < i \leq m$ **do**
 - 11: $c'(u, d) = c'(u, d) + c'(e), \forall u$ corresponds to I_i
 - 12: remove e
 - 13: **end for**
 - 14: $V(G[w_i]) = V(G[w_i]) \setminus \{s, t\}$
 - 15: compute $s_i - t_i$ shortest path and pick its vertices
 - 16: **end if**
 - 17: **end for**
 - 18: **return** the outcome correspond to picked vertices

For the second part of the theorem the same proof holds. We just need to mention that since all voters have the same dependency graph, the issues can be rearranged in order to have only contiguous paths.

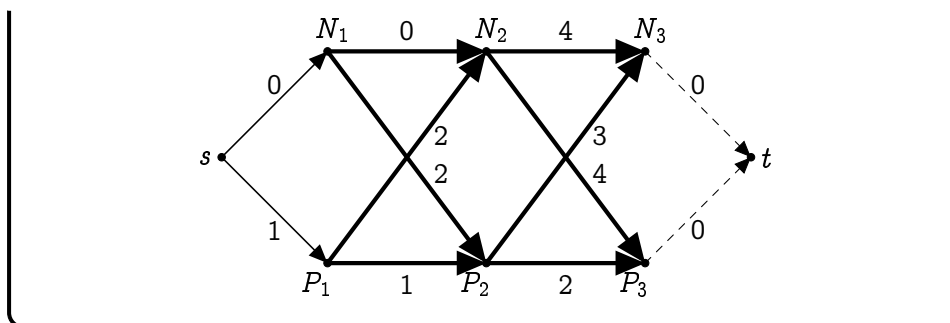
□

Example 22. Suppose that a voting profile produces the following dissatisfaction graph:



Then, the constructed graph presented in the reduction of theorem 5 is the following:

4.4. Positive Results



Theorem 6. When every voter's dependency graph G has $\max\{deg_{in}^G(u), u \in V(G)\} \leq 1$, we have a 2.2074 approximation algorithm.

Proof. Firstly we will prove that there is a polynomial time reduction from CMS when the dependency graph of every voter has maximum in-degree at most 1 to MIN2-SAT. Given the restrictions mentioned, we are dealing with the situation where every voter who votes for the issue i is allowed to give dependencies for i only according to an issue j , which can be an issue comes before or after i in the given ordering and can also be depending on i or on any other (at most one) issue.

Given B we will create a logic formula C_{ij} , for each pair (i, j) where i is a voter and j is an issue, which indicates the cases when i is not satisfied with the outcome of j . In what follows, we use (0) as the logical false value and we create a boolean variable for every issue, associating the boolean x with the outcome of j . Suppose first that j has no predecessor.

- If i voted for $\{\}$ at what concerning issue j then $C_{ij} = (x \vee \bar{x})$.
- If i voted for $\{x\}$ at what concerning issue j then $C_{ij} = (\bar{x})$.
- If i voted for $\{\bar{x}\}$ at what concerning issue j then $C_{ij} = (x)$.
- If i voted for $\{x, \bar{x}\}$ at what concerning issue j then $C_{ij} = (0)$.

On the other hand, if j is depended on the outcome of issue k and x, y are associated with j, k respectively, we set C_{ij} equal to the conjunction of all combinations (of two outcomes connected with disjunction) of issues j, k that dissatisfy voter i . For instance, if i voted for $\{\{P_k : P_j\}, \{P_k : N_j\}, \{N_k : P_j\}\}$ then $C_{ij} = (\bar{x} \wedge \bar{y})$ and if i voted for $\{\{P_k : P_j\}, \{N_k : N_j\}\}$

then $C_{ij} = (\bar{x} \wedge y) \vee (x \wedge \bar{y})$ and if i voted for all four possible outcomes, then $C_{ij} = (0)$.

We observe that in any case C_{ij} is a set containing either (0) or a set of one to four pairs of variables, with logical conjunction between the pairs and logical disjunction between the variables.

Lemma 3. Every statement of each voter will create an expression with at most 2 clauses, each of them contains at most 2 literals.

For the statements with approval votes, the lemma holds obviously. Let's pick a specific statement of voter i on issue j that is depended on issue k and the formula derived from it. Let x be associated with j and y with k . Voter i is dissatisfied with at least 0 and at most 4 options.

If i is dissatisfied with no possible outcomes or with exactly one outcome (for instance with $x : y$) then it suffices to add the expression $(x \wedge \bar{x})$ or the conjunction of the alternatives in the aforementioned outcome (in our case $(x \wedge y)$) respectively, so the lemma holds obviously.

If i is dissatisfied with two statements then:

- If the produced expression has the form $(y \wedge x) \vee (y \wedge \bar{x})$, then is equivalent with (y) (similarly for $(\bar{y} \wedge x) \vee (\bar{y} \wedge \bar{x})$).
- If the produced expression has the form $(y \wedge x) \vee (\bar{y} \wedge x)$, then is equivalent with (x) (similarly for $(y \wedge \bar{x}) \vee (\bar{y} \wedge \bar{x})$).
- If the produced expression has the form $(y \wedge \bar{x}) \vee (\bar{y} \wedge x)$, then is equivalent with $(y \vee x) \wedge (\bar{y} \vee \bar{x})$.
- If the produced expression has the form $(y \wedge x) \vee (\bar{y} \wedge \bar{x})$, then is equivalent with $(y \vee \bar{x}) \wedge (\bar{y} \vee x)$.

If i is dissatisfied with three statements, without loss of generality, we are dealing with the situation $\{x : y, \bar{x} : y, x : \bar{y}\}$. Then the constructed expression is

$$(x \wedge y) \vee (\bar{x} \wedge y) \vee (x \wedge \bar{y}) \equiv (y \vee (x \wedge \bar{y})) \equiv ((y \vee x) \wedge (y \vee \bar{y})) \equiv (y \vee x).$$

4.4. Positive Results

If i is dissatisfied with every possible outcome, it suffices to add the expression $(x \vee \bar{x})$. Hence every expression has at most 2 clauses each of them contains at most 2 literals. \square

If in the created expression we make every possible simplification presented in the above lemma, then

$$\bigwedge_{i \in [n], j \in [m]} \{C_{ij}, C_{ij} \neq (0)\}$$

is an instance of MIN2-SAT. In the voting-scenario we aim to find a boolean assignment to minimize the total number of dissatisfied voters in all issues whether in the sat-scenario we aim to find a boolean assignment to minimize the number of satisfied clauses.

Thus, a YES-instance for the MIN2-SAT corresponds to a YES-instance for the CMS and vice-versa. The objective function of CMS is related to the cost of a MIN2-SAT solution since every voter's opinion is mapped to a boolean expression consisted of one or two clauses (depending on the form of the expressed opinion). In particular, the construction given has at most $2nm$ clauses with at most 2 literals each. Additionally, we can observe that if there are k dissatisfied clauses there must be less than or equal to $k/2$ unsatisfied voters. Equivalently

$$2\text{OPT}_{\text{CMS}} \geq \text{SOL}_{\text{MIN2-SAT}} \geq \text{OPT}_{\text{MIN2-SAT}}.$$

Thus every α -approximation algorithm for the MIN2-SAT problem is a 2α -approximation algorithm for the CMS problem. Finally, to obtain the desired approximation ratio we just need to use the algorithm presented at [AZ05] which achieves an approximation factor of 1.1037 for the MIN2-SAT problem. \square

The ideas of the proof given in theorem 6 still work for a reduction from CMS to MIN3-SAT when every voter's dependency graph G has $\max\{deg_{in}^G(u), u \in V(G)\} \leq 2$. In that case it is again only a matter of boolean algebra to check a lemma similar to lemma 3 so as to write any C_{ij} in CNF form with at most 4 clauses, each of them containing at most 3 literals, for every

$i \in [n], j \in [m]$. Thus, an α -approximation algorithm for MIN-3SAT is a 4α -approximation algorithm for CMS. Taking into account that for MIN3-SAT, we are aware of an 1.2136-approximation algorithm (again due to [AZ05]) we can obtain a 4.8544 approximation algorithm.

Obviously, it is natural to suppose that the same technique works for the reduction of CMS with maximum in-degree bounded by k to MIN($k+1$)-SAT but the approximation ratio is (still constant but) excessively large when k increases using the result in [BTV99]. More formally, when every voter's dependency graph G has $\max\{\deg_{in}^G(u), u \in V(G)\} \in \mathcal{O}(1)$ then there is an $\mathcal{O}(1)$ -approximation algorithm.

4.4.2 Positive Results for Minimax Solution

Theorem 7. When there is a constant number of issues CMM can be solved optimally.

Proof. CMM admits a solution exactly similar to the classic MINIMAX problem using a brute-force algorithm. □

Lemma 4. If all voters share the same dependency graph which is formed by a path through all issues and there is no option which cause dissatisfaction to more than a single voter, then the optimum solution of CMM, given a voting profile B with k voters, can be obtained from the optimum solution of k -BCLP given the, corresponding to B , dissatisfaction graph and s, t .

Proof. Consider the dissatisfaction graph $G_{ds} = (V, E, c)$, a set of $n + 1$ available colors $C = \{c_0, c_i : i \in [n]\}$ and a function $c : E \rightarrow C$. For every $e \in E$ with $c(e) = c_i, i > 0$ it is true that i -th voter is dissatisfied with the option indicated by e . We will also use w as a vector of dimension $(n + 1)$ where w_i indicates the maximum allowed number of times that c_{i-1} can be used, so w_1 corresponds to c_0, w_2 to c_1 etc. Since every option dissatisfies at most one voter, all edges can be colored with at most $n + 1$ colors: one color for each voter and an extra color c_0 for those edges that satisfy all voters, so a dimension of $(n + 1)$ for w suffices.

Solving CMM is equivalent with solving multiple times BCLP in the above graph. Since BCLP is a decision problem whereas CMM is an optimization

4.4. Positive Results

one, taking into account that the maximum dissatisfaction of any voter is m , we ought to run an algorithm which returns the optimum solution for BCLP at most m times. Firstly, if x_n is the n -dimensional array (x, x, \dots, x) , $x \in \mathbb{N}$, $w = (m, 0_n)$, then $w = (m, 1_n)$, then $w = (m, 2_n)$ and so on and so forth until either find a YES outcome or $w = m_{n+1}$ outputs NO.

□

Lemma 5. If all voters share the same dependency graph which is formed by a path through all issues and there is no option which cause dissatisfaction to more than a single voter, then K-BCLP can be reduced to K-BCPM.

Proof. We will following present a reduction from BOUNDED COLOR LONGEST PATH in $G(V, E)$ with coloring function c to BOUNDED COLOR PERFECT MATCHING in $G'(V', E')$ with coloring function c' and we denote as I_i^0 the option N_i and as I_i^1 the option P_i . Let $G' = (V', E', c')$ where

$$\begin{aligned} V' &= V, \\ E' &= \{\{I_{j-1}^k, I_j^{l'}\}, \forall \{I_{j-1}^k, I_j^l\} \in E, j \geq 2, l' = (l+1)\%2, k, l \in \{0, 1\}\} \cup \\ &\quad \{\{s, I_1^{l'}\}, l' = (l+1)\%2, l \in \{0, 1\}\} \cup \\ &\quad \{I_m^k, t\}, \forall \{I_m^k, t\} \in E\}, \\ c'(e') &= c(e) \text{ where } e \text{ is the corresponding edge of } e' \text{ in } E, \forall e' \in E'. \end{aligned}$$

We will convince the reader that a path in G_{ds} is a bounded color longest path if and only if it corresponds to a bounded color maximum matching in G' . Consequently, in order to solve CMM we just need to solve BOUNDED COLOR PERFECT MATCHING and return those options that correspond to vertices of the matching with out-degree equal to one.

It is not hard to see that every path in G_{ds} corresponds to a matching in G' due to the construction of G' . In particular, in every path of G_{ds} , two consecutive edges e_1, e_2 are sharing a common vertex. Although in G' the end vertex of e_2 was changed when forming e_1' and thus (since the definition of CMM ensures that there are no other possible common vertices) no common vertex exist between e_1' and e_2' .

For the other direction, suppose every maximum matching in G' is composed of exactly one edge from each block (we call a block, a set of four

vertices correspond to the options of two consecutive issues). Due to the construction of G' , that matching corresponds to a longest path in G_{ds} . If there is a block with exactly two edges in the matching then it must be the case that its predecessor-block and its successor-block do not contribute anything to the matching.

Suppose there are $k \in \mathbb{N}$ discrete groups of consecutive blocks in which every block contributes 0 or 2 edges to the matching. It is easily seen that every such group of blocks starts and ends with a block contributing 0 edges to the matching. Using the notation $x_i, i \in [k]$ for the number of blocks belonging to the i -th group, we can observe that the edges of the matching coming from those groups are $\sum_{i=1}^k (x_i - 1)$. All the other $m + 1 - \sum_{i=1}^k x_i$ blocks contributes $m + 1 - \sum_{i=1}^k x_i$ to the matching so the size of that matching is

$$\sum_{i=1}^k (x_i - 1) + m + 1 - \sum_{i=1}^k x_i = m + 1 - k.$$

On the other hand, we can find a matching which is formed exactly by one edge from every block, thus has size $m + 1$. Consequently, unless $k = 0$, the above mentioned matching is not a maximum one.

Concluding the proof, the proposed procedure for solving CMM, (avoiding some details) follows:

Algorithm 8 COMMON & UNIQUELY COLORED(I,V,B)

- 1: let G be the undirected version of the dissatisfaction graph
 - 2: **for** $u \in V(G) \setminus \{N_m, P_m\}$ **do**
 - 3: form G' by swapping the colors of the outgoing edges of u
 - 4: **end for**
 - 5: $A =$ set of edges outputted by an algorithm for $|V|$ -BCPM
 - 6: **return** the vertices correspond to the outcome of the latest issue of $e, \forall e \in A$
-

□

From the above lemma we can observe an interesting relation between CMM and EXACT MATCHING since the later is a special case of K-BCPM. This relation is worth mentioning since EXACT MATCHING is one of the very few

4.4. Positive Results

problems with unknown complexity characterization for more than 35 years since neither a polynomial time deterministic algorithm is known nor an NP-completeness proof. In fact, the problem was introduced by Papadimitriou and Yannakakis ([PY82]) and admits a randomized parallel optimal algorithm ([MVV87]) which makes us to believe that it is likely to belong in \mathbf{P} . For complete graphs and complete bipartite graphs Karzanov ([Kar87]) gave an algorithm for constructing such a matching. His proof was not totally clear and was revised in [GS11] and [YMS02] and finally in [Gur&al.13]. For planar graphs, the problem can also be solved in polynomial time ([BP87]). In addition, an optimal solution also exists for $K_{3,3}$ -free graphs due to [Vaz89]. Finally, a polynomial algorithm was given in [Yus12] which either certifies that no such matching exists or it returns a matching with one less edge which satisfies the color bound.

Of course it can be easily observed that it is hardly ever the case where each option cause dissatisfaction to at most one voter so as to apply the procedures of lemmata 4 and 5 but the next theorem is a useful generalization of it.

Theorem 8. If all voters share the same dependency graph which is formed by a single path through all issues then CMM can be solved optimally for constant number of voters.

Proof. We will present a procedure that makes use of lemma 5 by executing multiple times an algorithm for K-BCPM with different input vectors w in each execution. We are firstly going to state which those vectors are going to be together with a sketch that proves that only a polynomial number of executions suffices to solve CMM and finally we will justify for the existence of an optimal algorithm for K-BCPM in our setting.

Instead of using a vector w of size $n + 1$ as input (as we did in the lemma 5) to the K-BCPM we will create a vector with one coordinate for each combination of voters which is present in the dissatisfaction graph. Thus we will use a vector of size c which is at most equal to the cardinality of the power-set of n voters, $\mathcal{O}(2^n)$. Suppose that voter i is the one with the maximum dissatisfaction d , then any other voter must have a dissatisfaction at most d . In this point we must notice that a dissatisfaction d of voter i can be caused by any possible combination of coordinates of vector w which correspond to

sets that include i . So we have to check all the ways to distribute d units of dissatisfaction to the coordinates that correspond to sets that include i which are polynomially many if n is constant. Until now we are able to assign values to all coordinates that correspond to sets which include i . A similar analysis is required in order to assign values in every other coordinate of w and we can observe that in any case (provided that n is constant) the number of executions will still be polynomially large. Furthermore, the above procedure must be executed nm times since we have to check the values of the vector for any voter that could be the most dissatisfied and for any total dissatisfaction, which is bounded by m . Thus, we conclude that if n is constant, a known algorithm for BCPM suffices to solve CMM for the case when all voters have the same dependency graph which is formed by a path through all issues and the desired number of executions is polynomially large.

The problem κ -BCPM (and thus CMM, can be solved optimally using the randomized algorithm presented in [Gra&al.14] or a generalization of the randomized algorithm given in [NPZ07]. But our will is an optimum deterministic algorithm. There is a work ([Gra&al.10]) which deals with κ -BCPM allowing a function that assigns a weight to each edge. It proves that there is a multicriteria FPTAS or FPRAS (which finds a perfect matching violating the desired total weight by $(1+\epsilon)$ and each constraint by $(1+\epsilon)$) for problems whose exact version admit pseudopolynomial algorithm or randomized algorithm respectively. In particular it is mentioned that for the case of planar graphs, using the result of [BP87], we can achieve a deterministic FPTAS. We claim that the dissatisfaction graph in our setting can be embedded in a plane and if it is not immediately seen we refer the reader to the example 23.

Until now, we have proved that we can achieve an FPTAS for CMM when the number of voters is a constant. Although, since our problem is unweighted the obtained FPTAS does not violate the cost of the objective function since it outputs a valid perfect matching and thus it only violates the constraints by $(1+\epsilon)$ each. Observing that the coordinates of w are all integers and the approximation scheme is fully polynomial-time, we can pick an appropriate ϵ (say $\epsilon = \frac{1}{m^2}$) so that no violation occurs in any color. Actually, the FPTAS

4.4. Positive Results

gives a solution with the i -th color appearing w_i^{sol} times such that:

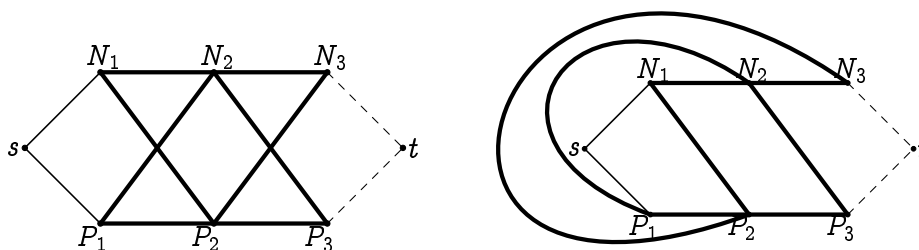
$$w_i^{sol} \leq (1 + \epsilon)w_i \leq (1 + \frac{1}{m^2})w_i \leq w_i + \frac{w_i}{m^2} < w_i + 1,$$

where the strict inequality follows from the fact that any $w_i \leq m$. Since every w_i is integer we have that

$$w_i^{sol} \leq w_i,$$

and hence we can conclude that in the case of our setting (which forms a planar graph) the optimum existing algorithm for EXACT MATCHING can be transformed to an FPTAS for κ -BCPM which in fact always produces an optimum solution for CMM. □

Example 23. Revising the example 22, we will show how the produced dissatisfaction graph can be embedded in a plane as stated in the proof of theorem 8. The following are two drawings of the same graph.



Of course we just prove the planarity claim for a specific case only, but it is obvious how the result can be generalized for arbitrary number of issues.

An analysis similar to the one of theorem 5 should convince the reader that theorem 8 can be generalized for the cases when each voter i has a dependency graph G_i formed by collections of contiguous subpaths of L_m (not necessarily common) or when every voter has the same dependency graph, formed by subpaths of L_m (not necessarily contiguous), always requiring a constant number of voters. Additionally, issues with no-predecessors for every voter are also acceptable.

BIBLIOGRAPHY

- [Air&al.11] Stéphane Airiau, Ulle Endriss, Umberto Grandi, Daniele Porello, and Joel Uckelman. “Aggregating Dependency Graphs into Voting Agendas in Multi-Issue Elections.” In: *IJCAI*. 2011, pp. 18–23.
- [Ama&al.15] Georgios Amanatidis, Nathanaël Barrot, Jérôme Lang, Evangelos Markakis, and Bernard Ries. “Multiple referenda and multiwinner elections using hamming distances: Complexity and manipulability”. In: *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*. 2015, pp. 715–723.
- [Ama&al.17] Georgios Amanatidis, Georgios Birmpas, George Christodoulou, and Evangelos Markakis. “Truthful allocation mechanisms without payments: Characterization and implications on fairness”. In: *Proceedings of the 2017 ACM Conference on Economics and Computation*. 2017, pp. 545–562.
- [Arr12] Kenneth Arrow. *Social choice and individual values*. Vol. 12. Yale university press, 2012.
- [ASS10] Kenneth Arrow, Amartya Sen, and Kotaro Suzumura. *Handbook of social choice and welfare*. Vol. 2. Elsevier, 2010.
- [AZ05] Adi Avidor and Uri Zwick. “Approximating MIN 2-SAT and MIN 3-SAT”. In: *Theory Computing Systems* 38 (2005), pp. 329–345.
- [Azi&al.15] Haris Aziz, Serge Gaspers, Joachim Gudmundsson, Simon Mackenzie, Nicholas Mattei, and Toby Walsh. “Computational aspects of multi-winner approval voting”. In: *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*. 2015, pp. 107–115.

Bibliography

- [Ben&al.18] Nawal Benabbou, Mithun Chakraborty, Xuan-Vinh Ho, Jakub Sliwinski, and Yair Zick. “Diversity constraints in public housing allocation”. In: *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. 2018, pp. 973–981.
- [BK10] Jean-Pierre Benoît and Lewis A Kornhauser. “Only a dictatorship is efficient”. In: *Games and Economic Behavior* 70.2 (2010), pp. 261–270.
- [BKP19] Steven Brams, Marc Kilgour, and Richard Potthoff. “Multiwinner approval voting: an apportionment approach”. In: *Public Choice* 178.1-2 (2019), pp. 67–93.
- [BKS04] Steven Brams, Marc Kilgour, and Remzi Sanver. “A minimax procedure for negotiating multilateral treaties”. In: *Diplomacy games*. Springer, 2004, pp. 265–282.
- [BKS07] Steven Brams, Marc Kilgour, and Remzi Sanver. “A minimax procedure for electing committees”. In: *Public Choice* 132.3-4 (2007), pp. 401–420.
- [BKZ97] Steven Brams, Marc Kilgour, and William Zwicker. “Voting on referenda: the separability problem and possible solutions”. In: *Electoral Studies* 16.3 (1997), pp. 359–377.
- [BL16] Nathanaël Barrot and Jérôme Lang. “Conditional and sequential approval voting on combinatorial domains”. In: *25th International Joint Conference on Artificial Intelligence (IJCAI’16)*. 2016, pp. 88–94.
- [Bla&al.58] Duncan Black, Robert Albert Newing, Iain McLean, Alistair McMillan, and Burt Monroe. *The theory of committees and elections*. Springer, 1958.
- [Boo&al.10] Richard Booth, Yann Chevaleyre, Jérôme Lang, Jérôme Mengin, and Chattrakul Sombatheera. “Learning conditionally lexicographic preference relations.” In: *ECAI*. 2010, pp. 269–274.
- [Bou&al.04] Craig Boutilier, Ronen Brafman, Carmel Domshlak, Holger Hoos, and David Poole. “CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements”. In: *Journal of artificial intelligence research* 21 (2004), pp. 135–191.
- [BP87] Francisco Barahona and William R Pulleyblank. “Exact arborescences, matchings and cycles”. In: *Discrete Applied Mathematics* 16.2 (1987), pp. 91–99.

-
- [Bra&al.16] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel Procaccia. *Handbook of computational social choice*. Cambridge University Press, 2016.
- [BS14] Jarosaw Byrka and Krzysztof Sornat. “PTAS for minimax approval voting”. In: *International Conference on Web and Internet Economics*. Springer. 2014, pp. 203–217.
- [BSS18] Jaroslaw Byrka, Piotr Skowron, and Krzysztof Sornat. “Proportional Approval Voting, Harmonic k-median, and Negative Association”. In: *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2018.
- [BTV99] Dimitris Bertsimas, Chungpiaw Teo, and Rakesh Vohra. “On dependent randomized rounding algorithms”. In: *Operations Research Letters* 24.3 (1999), pp. 105–114.
- [CEW11] Georgios Chalkiadakis, Edith Elkind, and Michael Wooldridge. “Computational aspects of cooperative game theory”. In: *Synthesis Lectures on Artificial Intelligence and Machine Learning* 5.6 (2011), pp. 1–168.
- [Che&al.07] Yann Chevaleyre, Ulle Endriss, Jérôme Lang, and Nicolas Maudet. “A short introduction to computational social choice”. In: *International Conference on Current Trends in Theory and Practice of Computer Science*. Springer. 2007, pp. 51–69.
- [Che&al.08] Yann Chevaleyre, Ulle Endriss, Jérôme Lang, and Nicolas Maudet. “Preference handling in combinatorial domains: From AI to social choice”. In: *AI magazine* 29.4 (2008), p. 37.
- [CKM10] Ioannis Caragiannis, Dimitris Kalaitzis, and Evangelos Markakis. “Approximation Algorithms and Mechanism Design for Minimax Approval Voting.” In: *AAAI*. 2010.
- [ÇL12] Tuçe Çuhadarolu and Jean Lainé. “Pareto efficiency in multiple referendum”. In: *Theory and decision* 72.4 (2012), pp. 525–536.
- [CX12] Vincent Conitzer and Lirong Xia. “Approximating common voting rules by sequential voting in multi-issue domains”. In: *Proceedings of KR 2012* (2012).
- [Cyg&al.18] Marek Cygan, Lukasz Kowalik, Arkadiusz Socala, and Krzysztof Sornat. “Approximation and parameterized complexity of minimax approval voting”. In: *Journal of Artificial Intelligence Research* 63 (2018), pp. 495–513.

Bibliography

- [Elk&al.17a] Edith Elkind, Piotr Faliszewski, Jean-François Laslier, Piotr Skowron, Arkadii Slinko, and Nimrod Talmon. “What Do Multiwinner Voting Rules Do? An Experiment Over the Two-Dimensional Euclidean Domain.” In: *AAAI*. 2017, pp. 494–501.
- [Elk&al.17b] Edith Elkind, Piotr Faliszewski, Piotr Skowron, and Arkadii Slinko. “Properties of multiwinner voting rules”. In: *Social Choice and Welfare* 48.3 (2017), pp. 599–632.
- [End17] Ulle Endriss. *Trends in Computational Social Choice*. Lulu.com, 2017.
- [End18] Ulle Endriss. “Lecture Notes on Fair Division”. In: *CoRR* abs/1806.04234 (2018).
- [EP05] Bruno Escoffier and Vangelis Paschos. “Differential Approximation of min sat, max sat and Related Problems”. In: *Computational Science and Its Applications – ICCSA 2005*. Springer Berlin Heidelberg, 2005, pp. 192–201.
- [Fal&al.17] Piotr Faliszewski, Piotr Skowron, Arkadii Slinko, and Nimrod Talmon. “Multiwinner voting: A new challenge for social choice theory”. In: *Trends in Computational Social Choice* 74 (2017).
- [Fis15] Peter Fishburn. *The theory of social choice*. Princeton University Press, 2015.
- [Gib73] Allan Gibbard. “Manipulation of voting schemes: a general result”. In: *Econometrica* 41.4 (1973), pp. 587–601.
- [GPQ08] Christophe Gonzales, Patrice Perny, and Sergio Queiroz. “GAI-networks: Optimization, ranking and collective choice in combinatorial domains”. In: *Foundations of computing and decision sciences* 33.1 (2008), pp. 3–24.
- [Gra&al.10] Fabrizio Grandoni, Piotr Krysta, Stefano Leonardi, and Carmine Ventre. “Utilitarian Mechanism Design for Multi-Objective Optimization”. In: vol. 43. Jan. 2010, pp. 573–584. DOI: 10.1137/130913602.
- [Gra&al.14] Fabrizio Grandoni, R Ravi, Mohit Singh, and Rico Zenklusen. “New approaches to multi-objective optimization”. In: *Mathematical Programming* 146.1-2 (2014), pp. 525–554.
- [GS11] Hans-Florian Geerdes and Jácint Szabó. *A unified proof for Karzanov’s exact matching theorem*. Tech. rep. QP-2011-02. Egerváry Research Group, Budapest, 2011.

-
- [Gur&al.13] Rohit Gurjar, Arpita Korwar, Jochen Messner, and Thomas Thierauf. “Exact Perfect Matching in Complete Graphs.” In: *Electronic Colloquium on Computational Complexity (ECCC)*. Vol. 20. 2013, p. 112.
- [HM15] Jac Heckelman and Nicholas Miller. *Handbook of social choice and voting*. Edward Elgar Publishing, 2015.
- [Kad72] Joseph Kadane. “On division of the question”. In: *Public Choice* 13.1 (1972), pp. 47–54.
- [Kar87] A.V. Karzanov. “Maximum matching of given weight in complete and complete bipartite graphs”. In: *Cybernetics and Systems Analysis* 23.1 (1987), pp. 8–13.
- [KBS06] Marc Kilgour, Steven Brams, and Remzi Sanver. “How to elect a representative committee using approval balloting”. In: *Mathematics and Democracy*. Springer, 2006, pp. 83–95.
- [KL05] Kathrin Konczak and Jérôme Lang. “Voting procedures with incomplete preferences”. In: *Proc. IJCAI-05 Multidisciplinary Workshop on Advances in Preference Handling*. Vol. 20. 2005.
- [LB11] Tyler Lu and Craig Boutilier. “Budgeted Social Choice: From Consensus to Personalized Decision Making.” In: Jan. 2011, pp. 280–286.
- [LeG04] Rob LeGrand. *Analysis of the minimax procedure*. WUCSE-2004-67. All Computer Science and Engineering Research, 2004.
- [Lis13] Christian List. “Social Choice Theory”. In: *Stanford Encyclopedia of Philosophy*. 2013.
- [LL09] Gilbert Laffond and Jean Lainé. “Condorcet choice and the Ostrogorski paradox”. In: *Social Choice and Welfare* 32.2 (2009), pp. 317–333.
- [LMM06] Rob LeGrand, Evangelos Markakis, and Aranyak Mehta. “Approval voting: local search heuristics and approximation algorithms for the minimax solution”. In: *Proceedings of the First International Workshop on Computational Social Choice (COMSOC 2006), University of Amsterdam*. 2006, pp. 234–247.
- [LN00] Dean Lacy and Emerson MS Niou. “A problem with referendums”. In: *Journal of Theoretical Politics* 12.1 (2000), pp. 5–31.

Bibliography

- [LP10] Christian List and Ben Polak. “Introduction to judgment aggregation”. In: *Journal of Economic Theory* 145.2 (2010), pp. 441–466.
- [LS18] Martin Lackner and Piotr Skowron. “Consistent approval-based multi-winner rules”. In: *Proceedings of the 2018 ACM Conference on Economics and Computation*. 2018, pp. 47–48.
- [LVK10] Minyi Li, Quoc Bao Vo, and Ryszard Kowalczyk. “An Efficient Majority-Rule-Based Approach for Collective Decision Making with CP-Nets.” In: *KR*. 2010.
- [LX16] Jérôme Lang and Lirong Xia. *Voting in Combinatorial Domains*. 2016.
- [Mei17] Reshef Meir. “Iterative voting”. In: *Trends in Computational Social Choice* (2017), p. 69.
- [MNS15] Neeldhara Misra, Arshed Nabeel, and Harman Singh. “On the parameterized complexity of minimax approval voting”. In: *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*. 2015, pp. 97–105.
- [MVV87] Ketan Mulmuley, Umesh Vazirani, and Vijay Vazirani. “Matching is as easy as matrix inversion”. In: *Combinatorica* 7.1 (1987), pp. 105–113.
- [NPZ07] Christos Nomikos, Aris Pagourtzis, and Stathis Zachos. “Randomized and Approximation Algorithms for Blue-Red Matching”. In: *Mathematical Foundations of Computer Science 2007*. Springer Berlin Heidelberg, 2007, pp. 715–725.
- [ÖS06] Ipek Özkal-Sanver and Remzi Sanver. “Ensuring pareto optimality by referendum voting”. In: *Social Choice and Welfare* 27.1 (2006), pp. 211–219.
- [PY82] Christos Papadimitriou and Mihalis Yannakakis. “The complexity of restricted spanning tree problems”. In: *Journal of the ACM (JACM)* 29.2 (1982), pp. 285–309.
- [RA86] Hervé Raynaud and Kenneth Arrow. *Social choice and multicriterion decision-making*. Cambridge, Mass.: MIT Press, 1986.
- [Ren01] Philip Reny. “Arrows theorem and the Gibbard-Satterthwaite theorem: a unified approach”. In: *Economics Letters* 70.1 (2001), pp. 99–105.

- [Sat75] Mark Allen Satterthwaite. “Strategy-proofness and Arrow’s conditions: Existence and correspondence theorems for voting procedures and social welfare functions”. In: *Journal of economic theory* 10.2 (1975), pp. 187–217.
- [SF17] Piotr Skowron and Piotr Faliszewski. “Chamberlin–Courant Rule with Approval Ballots: Approximating the MaxCover Problem with Bounded Frequencies in FPT Time”. In: *Journal of Artificial Intelligence Research* 60 (2017), pp. 687–716.
- [Vaz89] Vijay Vazirani. “NC algorithms for computing the number of perfect matchings in K_3 , 3-free graphs and related problems”. In: *Information and computation* 80.2 (1989), pp. 152–164.
- [XCL11] Lirong Xia, Vincent Conitzer, and Jérôme Lang. “Strategic sequential voting in multi-issue domains and multiple-election paradoxes”. In: *Proceedings of the 12th ACM conference on Electronic commerce*. 2011, pp. 179–188.
- [YMS02] Tongnyoul Yi, Katta Murty, and Cosimo Spera. “Matchings in colored bipartite networks”. In: *Discrete Applied Mathematics* 121.1-3 (2002), pp. 261–277.
- [Yus12] Raphael Yuster. “Almost Exact Matchings”. In: *Algorithmica* 63.1 (June 2012), pp. 39–50.
- [YW18] Yongjie Yang and Jianxin Wang. “Parameterized Complexity of Multi-winner Determination: More Effort Towards Fixed-Parameter Tractability”. In: *Proceedings of the 17th International Conference on Autonomous Agents and Multi-Agent Systems*. 2018, pp. 2142–2144.

